OPTIMAL SENSING FOR FILTERING WITH BOUNDED ERRORS

A Dissertation

by

NILADRI DAS

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Raktim Bhattacharya
Committee Members,	Srinivas Rao Vadali
	Suman Chakravorty
	P.R. Kumar
Head of Department,	Srinivas Rao Vadali

December   2020

Major Subject: Aerospace Engineering

ABSTRACT

We address the problem of designing optimal sensing strategy for stochastic discrete-time systems. Sensors are an integral part of a system, providing knowledge about system states, through state filtering. The problem of designing optimal sensing, primarily addresses questions regarding, a) which type of sensor do we need, b) how accurate sensors do we need, and c) when and d) where do we use them. The desired performance of an optimal sensing strategy might also include minimizing energy consumption and total cost of operation or maximizing sensing accuracy or control performance, among various other metrics. Upper bounding and lower bounding the performance of a sensing strategy is tied to the notion of utility and privacy of the filtered system. The main contributions of this research are the formulations of algorithms and theorems that gives a structured way to manipulate sensing parameters to ensure either utility of the filtering system or privacy against the filtering system. To this end we show results on privacy and utility for Kalman Filter, Ensemble Kalman Filter, and Unscented Kalman Filter. The development of optimal sensing for Ensemble Kalman Filter and Unscented Kalman Filter is motivated by the space situational awareness problems regarding allocating sensing resources as well as exchanging data. The proposed contributions of this research are organized as follows. First we show preliminary results on optimal sensing for Ensemble Kalman Filter, and Unscented Kalman Filter, focusing on the utility problem. Next we address the utility and privacy problem for Kalman Filter in steady state scenario using Eigen-values analysis. We then move on to the utility problem for Kalman Filter for single-step, muti-step scenario. Finally, the utility and privacy formulations for Ensemble Kalman Filter, and Unscented Kalman Filter are further developed with

a focus on addressing space situational awareness problems.

DEDICATION

To my parents and my sister. To my friends that have always supported and

believed in me.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION TO OPTIMAL SENSING PRECISION IN ENSEMBLE AND UNSCENTED KALMAN FILTERING*

1.1 Introduction

In this work, we focus on the problem of sensor design for non-linear stochastic discrete-time systems. Sensors are an integral part of a system, providing knowledge about system states, through state estimation (filtering), which can further be utilized to control the system. The problem of sensor design for a system, primarily addresses questions regarding, a) which type of sensor do we need, b) how accurate sensors do we need, and c) when and d) where do we use them, as mentioned in [50]. The answers to the above problems explicitly depend upon, either the desired observability of the system, or the performance of the estimator and (or) the controller, or some performance metric of the system. This desired performance might also include minimizing energy consumption and total cost of operation or maximizing sensing accuracy or control performance, among various other metrics. In summary, sensor design strategy aims to strike a balance between the quality of sensing performance, sensing accuracy choice, and activation over space and (or) time.

A considerable work on sensor design for state estimation focuses on addressing sensor selection problem, such as [44, 86, 87], *when sensor precisions are known*. A typical sensor selection problem either deals with choosing a minimal subset of sensors, from a set of available sensors that guarantees the state estimate covariance to be bounded, as in [78], or in [88] where the authors minimizes the state estimate covariance when

the cardinality of the sensor set is bounded. Owing to the combinatorial complexity of the problem, current methods are heuristic and developed in the Kalman filtering framework for linear Gaussian systems. There is also limited work on nonlinear state estimation which includes [13], which focusses on choosing sparse sensor network with known sensor precisions.

The focus of this work *is not on sensor selection, but on determining the sensor precisions.* Particularly, determining the least precise sensors for which state estimation is achieved with desired accuracy. This is achieved by solving an l_1 minimization problem. Once the sensor precisions are determined, existing sensor selection algorithms such as in [44, 86, 87], can be applied to arrive at a reduced sensor set. However, due to the l_1 minimization, it is possible that the optimal solution assigns some of the precisions to zero, leading to sparsity in the sensor set. These sensors can be removed, indirectly addressing the sensor selection problem. Existing sensor selection algorithms can aid in further reducing the sensor set, possibly at the loss of estimation accuracy. In this work, we present sensor precision selection algorithm for nonlinear estimation based on ensemble Kalman filtering (EnKF) as discussed in [25] and unscented Kalman filtering (UKF) in [80].

Specifically, this work addresses the problem of determining the accuracy (or precision) of a given dictionary of sensors, for a given upper-bound on the estimation accuracy. This problem has been addressed by [50] for a linear continuous-time system, where the sensor precision and the control law were co-designed to achieve a specified closed-loop performance. The work also presents a state-estimation problem, where sensor precisions were determined to achieve a certain estimation accuracy. In this work, we look at a similar problem, but for a *nonlinear discrete-time system*, with a user specified upper-bound on the estimation error covariance.

This problem is important in many engineering applications where the choice and precision of sensors for state-estimation is not obvious. Examples of such applications include many large-scale spatio-temporal problems including space situational awareness where space objects are tracked using ground/space based sensor networks [20], structural health monitoring [50], environmental and climate monitoring [56], and distributed power-system monitoring [6], flow control applications [16], and many other practical problems mentioned in [9, 90, 39, 70, 37, 12].

However, due to the constraints on the communication bandwidth and sensor battery life, it may not be desirable to have all the sensors report their measurements at all time instants or use the highest energy settings as in active sensing scenario such as in [14]. Therefore, determining what should be the least accuracy of each sensor to achieve a given accuracy in the state estimate becomes important from a practical point-of-view. Since precision of a sensor is explicitly related to its cost, solution to the sensor precision problem has economical implication.

Contributions of the work: In this work we formulate a convex optimization problem to determine the optimal sensor precision for a given upper bound on the state estimation error covariance. This is presented for nonlinear discrete-time dynamical systems in EnKF and UKF frameworks. To the best of our knowledge, the sensor precision-selection problem for EnKF and UKF has not been addressed before. In this work, the l_1 norm of the sensor precision is minimized, subjected to a convex constraint that guarantees the desired estimation error. This allows us to start with an over parameterization of the problem and determine a sparse solution via l_1 regularization. Therefore, the system designer can specify a dictionary of sensors with unknown sensor precisions and use the algorithm presented here to determine the optimal precision (and possibly eliminate a few sensors) to achieve the require

3

estimation accuracy.

It is important to understand that the true error covariance calculated using exact Bayesian update, might be different from that predicted by EnKF and UKF. When EnKF and UKF are used to approximate the exact Bayesian update, the approximate error covariance is guaranteed to be bounded, if the sensor precision selection algorithms presented in this work are applied.

Notations: For a square matrix $\boldsymbol{M}$, let $\boldsymbol{M}^T$ denote its transpose. The variable $k \in \mathbb{Z}$ where $\mathbb{Z}$ is the set of integers, is used to index discrete time points; when used as subscript it refers to quantities taken at time k. The quantities $\boldsymbol{X}_k^{i-}$ and $\boldsymbol{X}_k^{i+}$ denotes prior and posterior random variable associated with state $\boldsymbol{X}_k^i$, where the superscript i denotes sample index. Observed value of the random variable $\boldsymbol{Y}_k$ is denoted by $\boldsymbol{Y}_k^o$. A positive definite matrix $\boldsymbol{M}$ is denoted by $\boldsymbol{M} \succ 0$. An identity and a zero matrix of dimension $n \times n$ is denoted by $\boldsymbol{I}_{n \times n}$ and $\boldsymbol{0}_{n \times n}$ respectively. Random variable $\boldsymbol{x}$ which has a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, is represented as $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. We represent the set of time indexed variable $\boldsymbol{x}_k$ as $\{\boldsymbol{x}_k\}$.

Layout of the work: The remainder of the work is organized as follows. In Section 2, we present the system model along with its corresponding augmented model. In Section 3, we describe the EnKF and UKF filter models, leading to the problem formulation in Section 4, where we present the algorithms to solve the sensor precision selection problem. In Section 5, the proposed framework is applied to the Lorenz 1996 model. The work finally concludes with Section 6.

1.2 System Model

Consider an input/output discrete-time stochastic system modeled by,

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}_k(\boldsymbol{x}_k, \boldsymbol{w}_k), \tag{1.1}$$

$$\boldsymbol{y}_k = \boldsymbol{h}_k(\boldsymbol{x}_k) + \boldsymbol{v}_k, \tag{1.2}$$

where $\boldsymbol{f}_k : \mathbb{R}^n \times \mathbb{R}^{n_w} \to \mathbb{R}^n$ represents the dynamics, $\boldsymbol{h}_k : \mathbb{R}^n \to \mathbb{R}^{n_y}$ is a measurement function, $\boldsymbol{x}_k \in \mathbb{R}^n$ and $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ are the state vector and the observation vector respectively, whereas $\boldsymbol{w}_k \in \mathbb{R}^{n_w}$ and $\boldsymbol{v}_k \in \mathbb{R}^{n_y}$ are the process noise and measurement noise respectively. We assume that both $\{\boldsymbol{w}_k\}$ and $\{\boldsymbol{v}_k\}$ are zero-mean, Gaussian, independent white random processes $[\boldsymbol{w}_k \sim \mathcal{N}(0, \boldsymbol{Q}_k), \boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R}_k), \mathbb{E}\left[\boldsymbol{w}_k \boldsymbol{w}_l^T\right] = \boldsymbol{Q}_k \delta_{kl}$, and $\mathbb{E}\left[\boldsymbol{v}_k \boldsymbol{v}_l^T\right] = \boldsymbol{R}_k \delta_{kl}]$. For sake of simplicity, the initial random variable $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ is independent of $\{\boldsymbol{w}_k\}$ and $\{\boldsymbol{v}_k\}$. We assume that $\boldsymbol{R}_k$ is a diagonal matrix, representing the measurement noise covariance. The inverse of $\boldsymbol{R}_k$ is the referred to as the *precision matrix*.

In EnKF and UKF, the measurement data $(\boldsymbol{y}_k^o)$ is used to determine the estimate of the state $\boldsymbol{x}_k$, which minimizes the estimation variance. We next introduce an *augmented model*, based on (4.3) and (4.4), which aids in formulating a multi-step precision selection problem that satisfies the specified performance criteria.

Augmented Model: We consider each of the q time steps $\{kq - q + 1, ..., kq\}$ of the system defined in (4.3) & (4.4) for $k \in \mathbb{Z}$, as a single time step
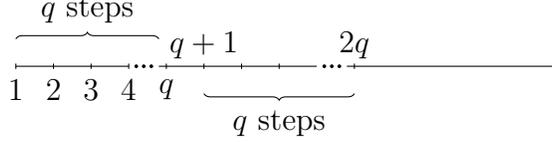
for the following augmented model:

$$\boldsymbol{X}_{k+1} = \boldsymbol{F}_k(\boldsymbol{X}_k, \boldsymbol{W}_k), \ \boldsymbol{Y}_k = \boldsymbol{H}_k(\boldsymbol{X}_k) + \boldsymbol{V}_k, \tag{1.3}$$

where,

$$\boldsymbol{X}_k := [\boldsymbol{x}_{kq-q+1}^T, ..., \boldsymbol{x}_{kq}^T]^T, \tag{1.4}$$

$$\boldsymbol{Y}_k := [\boldsymbol{y}_{kq-q+1}^T, ..., \boldsymbol{y}_{kq}^T]^T,$$

$$\boldsymbol{W}_k := [\boldsymbol{w}_{kq-q+1}^T, ..., \boldsymbol{w}_{kq+q-1}^T]^T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\mathcal{Q}}_k),$$

$$\boldsymbol{V}_k := [\boldsymbol{v}_{kq-q+1}^T, ..., \boldsymbol{v}_{kq}^T]^T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\mathcal{R}}_k),$$

$$\boldsymbol{\mathcal{Q}}_k := \text{diag}([\boldsymbol{Q}_{kq-q+1}, ..., \boldsymbol{Q}_{kq+q-1}]), \tag{1.5}$$

$$\boldsymbol{\mathcal{R}}_k := \text{diag}([\boldsymbol{R}_{kq-q+1}, ..., \boldsymbol{R}_{kq}]),$$

denotes stacked random variables. Function $\boldsymbol{F}_k(.)$ can be recursively generated using $\boldsymbol{f}_i(.)$s. It should be noted that the augmented model represents *a q-step q-shift process, rather than a q-step sliding-window process.*

Remark 1: In the rest of the work, we only use the augmented state model and consequently time step k denotes the batch of q time points $\{kq-q+1, ..., kq\}$, unless otherwise specified.

6

1.3 Filter Models: EnKF and UKF

The filtering process for the augmented model (4.5) consists of two sequential steps: dynamics update and measurement update. In EnKF, random samples are generated using Monte Carlo techniques, whereas the state distribution in UKF is represented by a Gaussian random variable (GRV) and is specified using a minimal set of carefully chosen deterministic sample points along with their associated weights, as shown in [80]. The sensor-selection problem for each these filtering frameworks are presented in the next section.

1.3.1 Dynamic Update for EnKF Model

Let $\boldsymbol{\mathcal{X}}_k^+ \in \mathbb{R}^{nq \times N}$ be the matrix with N number of *posterior* samples $\boldsymbol{X}_k^{i+}$ at time k, i.e.

$$\boldsymbol{\mathcal{X}}_k^+ = \begin{bmatrix} \boldsymbol{X}_k^{1+} & \boldsymbol{X}_k^{2+} & \cdots & \boldsymbol{X}_k^{N+} \end{bmatrix}.$$

The posterior mean from the samples is approximated as,

$$\boldsymbol{\mu}_k^+ := \mathbb{E}\left[\boldsymbol{X}_k^+\right] \approx \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{X}_k^{i+} = \frac{1}{N}\boldsymbol{\mathcal{X}}_k^+\mathbf{1}_N,$$

where $\mathbf{1}_N \in \mathbb{R}^N$ is a column vector of N ones. We define,

$$\bar{\boldsymbol{\mathcal{X}}}_k^+ := \begin{bmatrix} \boldsymbol{\mu}_k^+ & \cdots & \boldsymbol{\mu}_k^+ \end{bmatrix} = \boldsymbol{\mu}_k^+\mathbf{1}^T = \frac{1}{N}\boldsymbol{\mathcal{X}}_k^+\mathbf{1}\mathbf{1}^T,$$

then variance from the samples $\boldsymbol{\Sigma}_{xx,k}^+$ is,

$$\mathbb{E}\left[(\boldsymbol{X}_k^{i+} - \boldsymbol{\mu}_k^+)(\boldsymbol{X}_k^{i+} - \boldsymbol{\mu}_k^+)^T\right] \approx \boldsymbol{\mathcal{X}}_k^+\boldsymbol{A}\boldsymbol{\mathcal{X}}_k^{+T}. \tag{1.6}$$

where

$$\boldsymbol{A} := \left[\frac{1}{N-1} \left(\boldsymbol{I}_N - \frac{\boldsymbol{1}\boldsymbol{1}^T}{N} \right) \left(\boldsymbol{I}_N - \frac{\boldsymbol{1}\boldsymbol{1}^T}{N} \right) \right]$$

The state of each ensemble member at the next time step is estimated using the dynamics model:

$$\boldsymbol{X}_{k+1}^{i-} = \boldsymbol{F}_k(\boldsymbol{X}_k^{i+}, \boldsymbol{W}_k^i), \tag{1.7}$$

If applied to a linear system, this ensemble approach reduces the cost associated with the time propagation of the covariance matrix from $\mathcal{O}(n^3 q^3)$ (classical KF) to $\mathcal{O}(n^2 q^2 N)$ (EnKF).

1.3.2   Measurement update for EnKF Model

The ensemble members are corrected to minimize the error with respect to the measurements in the presence of noise and model uncertainties. A measurement update formulation proposed by [27] is:

$$\begin{aligned} \boldsymbol{X}_{k+1}^{i+} =& \boldsymbol{X}_{k+1}^{i-} + \boldsymbol{\Sigma}_{xy,k+1}^{-}(\boldsymbol{\Sigma}_{yy,k+1}^{-} + \boldsymbol{\mathcal{R}}_k)^{-1} \\ & \times (\boldsymbol{Y}_{k+1}^{o} - \boldsymbol{H}_{k+1}(\boldsymbol{X}_{k+1}^{i-}) + \boldsymbol{\epsilon}_k^i), \end{aligned} \tag{1.8}$$

where $\boldsymbol{\epsilon}_k^i$ is sampled from $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\mathcal{R}}_k)$. We define $\boldsymbol{\Sigma}_{xy,k+1}^{-}$ as:

$$\begin{aligned} \boldsymbol{\Sigma}_{xy,k+1}^{-} =& \frac{1}{N-1}(\boldsymbol{\mathcal{X}}_{k+1}^{-} - \bar{\boldsymbol{\mathcal{X}}}_{k+1}^{-}) \times \\ & (\boldsymbol{H}_{k+1}(\boldsymbol{\mathcal{X}}_{k+1}^{-}) - \boldsymbol{H}_{k+1}(\bar{\boldsymbol{\mathcal{X}}}_{k+1}^{-}))^T, \end{aligned} \tag{1.9}$$

and $\boldsymbol{\Sigma}_{yy,k+1}^-$ is defined as:

$$\boldsymbol{\Sigma}_{yy,k+1}^- = \frac{1}{N-1}\{\boldsymbol{H}_{k+1}(\boldsymbol{\mathcal{X}}_{k+1}^-) - \boldsymbol{H}_{k+1}(\bar{\boldsymbol{\mathcal{X}}}_{k+1}^-)) \times$$

$$(\boldsymbol{H}_{k+1}(\boldsymbol{\mathcal{X}}_{k+1}^-) - \boldsymbol{H}_{k+1}(\bar{\boldsymbol{\mathcal{X}}}_{k+1}^-))^T\}. \tag{1.10}$$

Remark 2: Equation (4.10) and (4.11) allows for direct evaluation of the nonlinear measurement function $\boldsymbol{H}_k(\boldsymbol{x})$ in calculating the Kalman gain, which is shown in [77] to hold for unbiased measurement forecasts $\{\boldsymbol{H}_k(\boldsymbol{X}_k^{i-})\}$, which we assume to be true in our work.

The covariance update equation of the augmented model is:

$$\boldsymbol{\Sigma}_{xx,k+1}^+ = \boldsymbol{\Sigma}_{xx,k+1}^- - \boldsymbol{\Sigma}_{xy,k+1}^-(\boldsymbol{\Sigma}_{yy,k+1}^- + \boldsymbol{\mathcal{R}}_{k+1})^{-1}$$

$$\times \boldsymbol{\Sigma}_{xy,k+1}^{-T}, \tag{1.11}$$

where $\boldsymbol{\Sigma}_{xx,k+1}^- = \boldsymbol{\mathcal{X}}_{k+1}^- \boldsymbol{A} \boldsymbol{\mathcal{X}}_{k+1}^{-T}$.

1.3.3 Dynamic Update for UKF Model

The dynamic update step from $k \rightarrow k+1$ starts with generating deterministic points called σ points. To capture the mean $_a\boldsymbol{\mu}_k^+$ of the augmented state vector $_a\boldsymbol{X}_k^+ := \begin{bmatrix} \boldsymbol{X}_k^+ \\ \boldsymbol{W}_k \end{bmatrix}$, where $_a\boldsymbol{X}_k^+ \in \mathbb{R}^{n_a}$ and $n_a = nq + n_w q$, as well as the augmented

error covariance $_a\boldsymbol{\Sigma}^+_{xx,k} = \begin{bmatrix} \boldsymbol{\Sigma}^+_{xx,k} & 0 \\ 0 & \boldsymbol{\mathcal{Q}}_k \end{bmatrix}$ the sigma points are chosen as

$$_a\boldsymbol{X}^{0+}_k = {}_a\boldsymbol{\mu}^+_k,$$

$$_a\boldsymbol{X}^{i+}_k = {}_a\boldsymbol{\mu}^+_k + \left(\sqrt{(n_a + \rho)_a\boldsymbol{\Sigma}^+_{xx,k}}\right)_i, i = 1, ..., n_a,$$

$$_a\boldsymbol{X}^{i+}_k = {}_a\boldsymbol{\mu}^+_k - \left(\sqrt{(n_a + \rho)_a\boldsymbol{\Sigma}^+_{xx,k}}\right)_{i-nq}, i = n_a + 1, ..., 2n_a,$$

with associated weights as

$$\boldsymbol{\omega}^{(m)}_0 = \rho/(n_a + \rho),$$

$$\boldsymbol{\omega}^{(c)}_0 = \rho/(n_a + \rho) + (1 - \alpha^2 + \beta),$$

$$\boldsymbol{\omega}^{(m)}_i = 1/\{2(n_a + \rho)\}.$$

The weight vectors are:

$$\boldsymbol{\mathcal{W}}^m = [\boldsymbol{\omega}^{(m)}_0 \boldsymbol{\omega}^{(m)}_1 ... \boldsymbol{\omega}^{(m)}_{2n_a+1}]^T,$$

$$\boldsymbol{\mathcal{W}}^c = [\boldsymbol{\omega}^{(c)}_0 \boldsymbol{\omega}^{(c)}_1 ... \boldsymbol{\omega}^{(c)}_{2n_a+1}]^T,$$

where $\rho = \alpha^2(n_a + \kappa) - n_a$ is the scaling parameter, α is set to 0.001, κ is set to 0, and β is 2 in this work. The term $\left(\sqrt{(n_a + \rho)_a\boldsymbol{\Sigma}^+_{xx,k}}\right)_i$ represents ith row of the matrix square root. The propagated state of each ensemble member at time $k + 1$ is generated exactly as EnKF by using $\boldsymbol{X}^{i-}_{k+1} = \boldsymbol{F}_k(\boldsymbol{X}^{i+}_k, \boldsymbol{W}^i_k)$, where $_a\boldsymbol{X}^{i+}_k := \begin{bmatrix} \boldsymbol{X}^{i+}_k \\ \boldsymbol{W}^i_k \end{bmatrix}$.

But unlike EnKF, the corresponding prior mean and covariance at time $k+1$ are:

$$\boldsymbol{\mu}_{k+1}^- = \boldsymbol{\mathcal{X}}_{k+1}^- \boldsymbol{\mathcal{W}}^m$$

$$\boldsymbol{\Sigma}_{xx,k+1}^- = \boldsymbol{\mathcal{X}}_{k+1}^- \boldsymbol{B}_k \boldsymbol{\mathcal{X}}_{k+1}^{-T}$$

where $\boldsymbol{B}_k := \boldsymbol{L}\boldsymbol{L}^T$, $\boldsymbol{L} := \left(\text{diag}(\boldsymbol{\mathcal{W}}^c) - \boldsymbol{\mathcal{W}}^c \mathbf{1}_{2nq+1}^T \right)$.

We define the following terms which we use in the following measurement update phase of UKF.

$$\boldsymbol{\mathcal{Y}}_{k+1}^- = \boldsymbol{H}(\boldsymbol{\mathcal{X}}_{k+1}^-), \ \bar{\boldsymbol{\mathcal{Y}}}_{k+1}^- = \boldsymbol{\mathcal{Y}}_{k+1}^- \boldsymbol{\mathcal{W}}^m \mathbf{1}_{2nq+1}^T,$$

$$\bar{\boldsymbol{\mathcal{X}}}_{k+1}^- = \boldsymbol{\mathcal{X}}_{k+1}^- \boldsymbol{\mathcal{W}}^m \mathbf{1}_{2nq+1}^T,$$

where $\boldsymbol{\mathcal{Y}}_{k+1}^- = \begin{bmatrix} \boldsymbol{Y}_{k+1}^{1-} & \boldsymbol{Y}_{k+1}^{2-} & \cdots & \boldsymbol{Y}_{k+1}^{(2nq+1)-} \end{bmatrix}$ and $\boldsymbol{\mathcal{X}}_{k+1}^- = \begin{bmatrix} \boldsymbol{X}_{k+1}^{1-} & \boldsymbol{X}_{k+1}^{2-} & \cdots & \boldsymbol{X}_{k+1}^{(2nq+1)-} \end{bmatrix}$

1.3.4 Measurement Update for UKF

We calculate $\boldsymbol{\Sigma}_{xy,k+1}^-$ and $\boldsymbol{\Sigma}_{yy,k+1}^-$ as:

$$\boldsymbol{\Sigma}_{xy,k+1}^- = (\boldsymbol{\mathcal{X}}_{k+1}^- - \bar{\boldsymbol{\mathcal{X}}}_{k+1}^-) \times \text{diag}(\boldsymbol{\mathcal{W}}^c)$$

$$\times (\boldsymbol{\mathcal{Y}}_{k+1}^- - \bar{\boldsymbol{\mathcal{Y}}}_{k+1}^-)^T \tag{1.12}$$

$$\boldsymbol{\Sigma}_{yy,k+1}^- = (\boldsymbol{\mathcal{Y}}_{k+1}^- - \bar{\boldsymbol{\mathcal{Y}}}_{k+1}^-) \times \text{diag}(\boldsymbol{\mathcal{W}}^c)$$

$$\times (\boldsymbol{\mathcal{Y}}_{k+1}^- - \bar{\boldsymbol{\mathcal{Y}}}_{k+1}^-)^T \tag{1.13}$$

The covariance update equation is exactly same as (4.12).

Remark 3: Since the covariance update equation for EnKF and UKF are identical,

this allows us to formulate a common precision selection algorithm that is presented next.

1.4 Problem Formulation

We define precision matrix of measurement ($\boldsymbol{S}_k$) as the inverse of the covariance matrix of the augmented measurement noise ($\boldsymbol{R}_k$). We assume that the precision matrix $\boldsymbol{S}_k$ is a diagonal matrix, with diagonal elements $\{\lambda_i\}$, where $i \in [1, ..., qn_y]$. The sensor precision associated with the ith sensor is λ_i. Equation (4.12) for time step k, can be written as:

$$
\begin{aligned}
\boldsymbol{\Sigma}^+_{xx,k} &= \boldsymbol{\Sigma}^-_{xx,k} - \boldsymbol{\Sigma}^-_{xy,k}(\boldsymbol{\Sigma}^-_{yy,k} + \boldsymbol{S}_k^{-1})^{-1}\boldsymbol{\Sigma}^{-T}_{xy,k} \\
&= \boldsymbol{\Sigma}^-_{xx,k} - \boldsymbol{\Sigma}^-_{xy,k}\{\boldsymbol{\Sigma}^-_{yy,k} + \text{diag}([\lambda_1, ..., \lambda_{qn_y}])^{-1}\}^{-1} \\
&\times \boldsymbol{\Sigma}^{-T}_{xy,k}
\end{aligned}
\tag{1.14}
$$

The λ_i's are the control variables, that regulate the estimation error covariance matrix $\boldsymbol{\Sigma}^+_{xx,k}$, when we have the prior ensemble $\boldsymbol{\mathcal{X}}^-_k$ which is generated from $\boldsymbol{\mathcal{X}}^+_{k-1}$ using (4.8). The augmented process noise $\boldsymbol{W}_k$ is generated by sampling from $\boldsymbol{\mathcal{Q}}_k$ in (1.5).

Our objective is to design $\{\lambda_i\}$ such that $\boldsymbol{M}_q\boldsymbol{\Sigma}^+_{xx,k}\boldsymbol{M}_q^T$ is upper-bounded by a pre-scribed positive definite matrix $\boldsymbol{P}^d_{kq}$, where the matrix $\boldsymbol{M}_q := [\boldsymbol{0}^1_{n \times n}, \boldsymbol{0}^2_{n \times n}, ..., \boldsymbol{0}^{q-1}_{n \times n}, \boldsymbol{I}_{n \times n}]$, is utilized to extract error covariance matrix of posterior estimate of $\boldsymbol{x}_{kq}$ from $\boldsymbol{\Sigma}^+_{xx,k}$. The matrix $\boldsymbol{P}^d_{kq}$ is the performance bound based on which we select sensor precisions.

Remark 4: Although we use the augmented model in (4.5), the performance bound is on the covariance of the estimate of $\boldsymbol{x}_{kq}$.

12

1.4.1 Optimal Sensor Precision

The solution to the sensor selection problem for EnKF and UKF is presented as the following theorem:

Theorem 1. *The optimal precision of each of the sensors, $\boldsymbol{\lambda}_k := [\lambda_1, ..., \lambda_{qn_y}]$ at time k, which guarantees $\boldsymbol{M}_q \Sigma_{xx,k}^+ \boldsymbol{M}_q^T \preceq \boldsymbol{P}_{kq}^d$, for given prior ensemble $\boldsymbol{\mathcal{X}}_{k-1}^+$, is obtained by solving the following semidefinite programming (SDP) problem,*

$$\boldsymbol{\lambda}_k^* = \min_{\boldsymbol{\lambda_k} := [\lambda_1, ..., \lambda_{qn_y}]^T} ||\boldsymbol{\lambda}_k||_1, \tag{1.15}$$

subject to,

$$\begin{bmatrix} \boldsymbol{P}_{kq}^d + \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{D} \end{bmatrix} \succeq 0, \quad \lambda_i \geq 0, \ \forall i \in [1, ..., qn_y], \tag{1.16}$$

where

$$\boldsymbol{A} := -\boldsymbol{M}_q \Sigma_{xx,k}^- \boldsymbol{M}_q^T + \boldsymbol{M}_q \Sigma_{xy,k}^- \boldsymbol{S}_k \Sigma_{xy,k}^{-T} \boldsymbol{M}_q^T,$$

$$\boldsymbol{B} := \boldsymbol{M}_q \Sigma_{xy,k}^- \boldsymbol{S}_k,$$

$$\boldsymbol{D} := (\Sigma_{yy,k}^-)^{-1} + \boldsymbol{S}_k,$$

$$\boldsymbol{S}_k := diag([\lambda_1, ..., \lambda_{qn_y}]).$$

The matrices $\Sigma_{xx,k}^-, \Sigma_{xy,k}^-, \Sigma_{yy,k}^-$ are calculated using the prior ensemble $\boldsymbol{\mathcal{X}}_k^-$ and the expected observations, $\boldsymbol{H}_k(\boldsymbol{\mathcal{X}}_k^-)$ using (4.10) & (4.11) for EnKF, or (4.20) & (4.21) for UKF. We calculate $\boldsymbol{\mathcal{X}}_k^-$ from $\boldsymbol{\mathcal{X}}_{k-1}^+$ using (4.8) both for EnKF and UKF.

Proof.

$$\Sigma_{xx,k}^{+} = \Sigma_{xx,k}^{-} - \Sigma_{xy,k}^{-}(\Sigma_{yy,k}^{-} + \mathcal{R}_k)^{-1}\Sigma_{xy,k}^{-T}$$

$$= \Sigma_{xx,k}^{-} - \Sigma_{xy,k}^{-}\{\mathcal{R}_k^{-1} - \mathcal{R}_k^{-1}$$

$$\times [(\Sigma_{yy,k}^{-})^{-1} + \mathcal{R}_k^{-1}]^{-1}\mathcal{R}_k^{-1}\}\Sigma_{xy,k}^{-T}$$

$$= \underbrace{\Sigma_{xx,k}^{-} - \Sigma_{xy,k}^{-}\mathcal{R}_k^{-1}\Sigma_{xy,k}^{-T}}_{-\hat{\boldsymbol{A}}}$$

$$+ \underbrace{\Sigma_{xy,k}^{-}\mathcal{R}_k^{-1}}_{\hat{\boldsymbol{B}}}\underbrace{[(\Sigma_{yy,k}^{-})^{-1} + \mathcal{R}_k^{-1}]^{-1}}_{\boldsymbol{D}^{-1}}$$

$$\times \underbrace{\mathcal{R}_k^{-1}\Sigma_{xy,k}^{-T}}_{\hat{\boldsymbol{B}}^T}$$

$$\Sigma_{xx,k}^{+} = -\hat{\boldsymbol{A}} + \hat{\boldsymbol{B}}\boldsymbol{D}^{-1}\hat{\boldsymbol{B}}^T,$$

Collection the error covariance corresponding to posterior estimate of $\boldsymbol{x}_{kq}$.

$$\boldsymbol{M}_q\Sigma_{xx,k}^{+}\boldsymbol{M}_q^T = -\boldsymbol{M}_q\hat{\boldsymbol{A}}\boldsymbol{M}_q^T + \boldsymbol{M}_q\hat{\boldsymbol{B}}\boldsymbol{D}^{-1}\hat{\boldsymbol{B}}^T\boldsymbol{M}_q^T$$

Now,

$$-\boldsymbol{A} + \boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^T \preceq \boldsymbol{P}_{kq}^d$$

$$\boldsymbol{P}_{kq}^d + \boldsymbol{A} - \boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^T \succeq 0, \tag{1.17}$$

where $\boldsymbol{M}_q\hat{\boldsymbol{A}}\boldsymbol{M}_q^T := \boldsymbol{A}$ and $\boldsymbol{M}_q\hat{\boldsymbol{B}} := \boldsymbol{B}$. Since $\boldsymbol{D} \succ 0$ and $\boldsymbol{P}_{kq}^d + \boldsymbol{A} - \boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^T \succeq 0$, using Schur's complement we get the following,

$$\begin{bmatrix} \boldsymbol{P}_{kq}^d + \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{D} \end{bmatrix} \succeq 0. \tag{1.18}$$

14

as the *necessary and sufficient condition* for (1.17) to be true.

Matrices $\boldsymbol{A}, \boldsymbol{B}$, and $\boldsymbol{D}$ are linear in $\boldsymbol{\mathcal{R}}_k^{-1}$ or $\boldsymbol{\mathcal{S}}_k$. Equation (1.18) is a linear matrix inequality (LMI) over $\boldsymbol{\mathcal{S}}_k$. The fact that the precision values are non-negative introduces the linear constraint $\lambda_i \geq 0$. The optimal precision is thus obtained by minimizing $\|\boldsymbol{\lambda}_k\|_1$, subject to the above LMI and linear constraint on $\boldsymbol{\lambda}_k$. $\qquad\square$

Algorithm 1 Precision Selection

Input: $\boldsymbol{f}_k(.), \boldsymbol{g}_k(.), \boldsymbol{h}_k(.), q, k, \boldsymbol{\mathcal{X}}_{k-1}^+, \boldsymbol{Q}_k, \boldsymbol{P}_{kq}^d$
Output: A set $\boldsymbol{\lambda} \in \{\mathbb{R}^{+(qn_y \times 1)} \cup \boldsymbol{0}_{qn_y \times 1}\}$ of sensor precisions.
  1: **procedure**
  2:     Calculate $\boldsymbol{\mathcal{Q}}_k$
  3:     $\boldsymbol{\mathcal{X}}_{k-1}^+ \to \boldsymbol{\mathcal{X}}_k^-$ using $\boldsymbol{F}_k(.), \boldsymbol{W}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\mathcal{Q}}_k)$
  4:     Calculate $\boldsymbol{\Sigma}_{xx,k}^-, \boldsymbol{\Sigma}_{xy,k}^-, \boldsymbol{\Sigma}_{yy,k}^-$
  5:     Calculate $\boldsymbol{M}_q$
  6:     Construct $\boldsymbol{\mathcal{S}}_k := \mathrm{diag}([\lambda_1, ..., \lambda_{qn_y}])$
  7:     Construct $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{D}$ matrices
  8:     Solve SDP problem in (1.15), (1.16)

Remark 5: Theorem 1 determines the optimal set of sensor precisions. The l_1 regularization induces sparseness in the solution. Therefore, if the optimization is performed on an over parameterized problem, i.e. with a large dictionary of sensors that considers all possible sensor choices, we expect the optimal solution $\boldsymbol{\lambda}^*$ to have entries that are zero. This indicates that those sensors are not needed to achieve the require state estimation accuracy and can be removed.

However, numerical solution to the sensor precision problem will result in small precision values that are not exactly zero. Those sensors can then be eliminated iteratively using theorem 1 with the reduced dictionary of sensors, discussed later in

this work. An upper bound on the precision for sensor(s) can also be incorporated in the optimization problem as a convex constraint over the argument space. For EnKF, covariance inflation [85] technique is used while calculating $\mathbf{\Sigma}_{xx,k}^{-}, \mathbf{\Sigma}_{xy,k}^{-}, \mathbf{\Sigma}_{yy,k}^{-}$ matrices.

Remark 6: Before solving the SDP problem, it is recommended to formulate the optimization problem with respect to normalized variables. This improves the numerical accuracy of the solution and make the optimal solution meaningful. For instance, in sensor precision selection for improving space-situational awareness, certain states are in the order of 10^3 km and others are in radians. Therefore, normalization with respect to dynamics, error covariance, and sensor noise is required to avoid ill conditioning of covariance matrices and improve the efficacy of the proposed optimal sensor precision algorithm.

1.4.2 Discussion: Sensor Pruning

As mentioned earlier, numerical solution of the l_1 regularization problem may assign small precision to certain $\{\lambda_i\}$s of $\boldsymbol{\lambda}$, which are not exactly zero and can possibly be removed without affecting the estimation quality. Therefore, a separate pruning process is required to reduce the set of sensors in the system. We define sensor pruning as choosing a subset of available sensors, which ensures that the covariance bound $\boldsymbol{P}_{kq}^d$ is satisfied. The rationale behind choosing a subset of the sensors is to eliminate the sensors whose precision requirement is too low compared to other sensors.

An adhoc algorithm to address this has been presented in [50], where the calculated sensor precision vector $\boldsymbol{\lambda}^*$ is first sorted in ascending order. Iteratively, the smallest

precision sensor are removed and the precisions are recalculated. This is continued till the problem becomes infeasible. The work of [50] focusses on integrated design of controller and sensing architecture, without taking observability into consideration. However, in the context of state-estimation, observability condition must be addressed while sensor pruning. Other sensor selection algorithms proposed in [78] and [88] can also be investigated. However, the problem of sensor pruning becomes challenging for nonlinear system, and is the focus of our future work.

1.5 Numerical Experiment

In this section, we provide simulation results for the sensor precision selection algorithm for single time step update ($q = 1$) and multiple time step update ($q = 3$); also including the case where sensor precisions are constrained.

1.5.1 Test Problem: The Lorenz (1996) model

The sensor precision selection scheme is applied to the Lorenz-96 (L96) model to test its validity, when an EnKF and an UKF filter are used for state estimation. The L96 model consists of N_x equally spaced variables, x_i for $i = 1, ..., N_x$, which are evolved in time using the set of differential equations:

$$\frac{dx_i}{dt} = (x_{i+1} - x_{i-2})x_{i-1} - x_i + F, \tag{1.19}$$

with cyclic boundaries: $x_{i+N} = x_i$ and $x_{i-N} = x_i$. The three terms in (1.19) are analogous to advection, damping, and forcing. The system exhibits varying degrees of chaotic behaviors depending on the choices of F and N_x. We fix N_x and F at 20 and 8 respectively, which leads to chaotic behavior in the system dynamics as shown

17

in [54], and [55].

1.5.2 Experimental Setup

In this experiment we consider no process noise, i.e. $\boldsymbol{\mathcal{Q}}_k = 0$, but with initial condition uncertainty. Forward integration of (1.19) is performed numerically using the fourth-order Runge-Kutta method with 20 internal stages for $k \rightarrow k+1$, with a time step of 0.05 time units for each stage as shown in [54]. We assume the following non-linear measurement model:

$$y_{i,k} = \frac{1}{1 + e^{-x_{i,k}}} + v_{i,k} \tag{1.20}$$

where $(.)_{i,k}$ denotes i^{th} component of a vector at time point k, with measurement noise $\boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R}_k)$. The initial ensemble is generated from a multivariate Gaussian distribution with mean vector of size qN_x, whose elements are chosen randomly from $[0, F]$ and a random positive definite matrix $(\boldsymbol{\Sigma}_{init})$ of size $qN_x \times qN_x$ as the covariance, where qN_x denotes the size of the augmented state vector. We use $2qN_x + 1$ number of samples for both EnKF and UKF, for $q = 1$ and 3.

To study the effects of state estimate covariance bounds on the optimal sensor precision values, we linearly vary the required error covariance bound from a factor of 0.9 to 0.6 of the initial covariance $\boldsymbol{\Sigma}_{init}$ as shown along the x-axis of the figures. For $q = 1$ shown in fig.(1.1) and fig.(1.3), 21 linearly varying bounds are considered within the interval of $[0.9, 0.6]$, where as for $q = 3$ shown in fig.(1.2), fig.(1.4), 7 linearly varying bounds are chosen from the same interval. Measurement model in (1.20) shows 20 different sensors, whose indices are plotted along the y-axis of each of the figures.

1.5.3   Solving for Sensor Precisions

We use CVX software of [33] to solve our SDP problem. CVX internally calls Se-DuMi solver of [74], a MATLAB implementation of the second-order interior-point methods. The l_1 norm minimization problem with LMI constraint yields desired precision values for the sensors shown in the figures as heat maps, with sensor precision ranges shown in the right y-axis. Fig.(1.1) and fig.(1.2) show optimal precisions required for EnKF for $q = 1$ and $q = 3$ respectively, satisfying prescribed covariance bounds. Fig.(1.3) and fig.(1.4), are the corresponding plots for UKF. For $q = 1$, sensor precisions are calculated to satisfy the covariance bound at the immediate next time instant. When $q = 3$, sensor precision are calculated for consecutive 3 time instants to satisfy covariance bound on the state variable at the end of the time horizon. For EnKF, the sensor precisions are restricted to be below 15, whereas for UKF precisions are bounded above by 3, while solving the SDP problem. We see that the optimal solution results in high accuracy sensing only at the end of the time interval, with poor (or no) sensing within the interval. However, this changes when upper limit on the precisions are reduced. In that case, we will see higher precision within the interval.

Note that we get different values for the optimal precisions for EnKF and UKF. These also depend on the sample size and covariance inflation parameter for EnKF, and choices of α, β, κ for UKF. It will be interesting to investigate the impact of these two frameworks on the sensor precision problem, and determine if one requires more precision than the other to arrive at the same estimation accuracy. These important questions will be addressed in our future work.
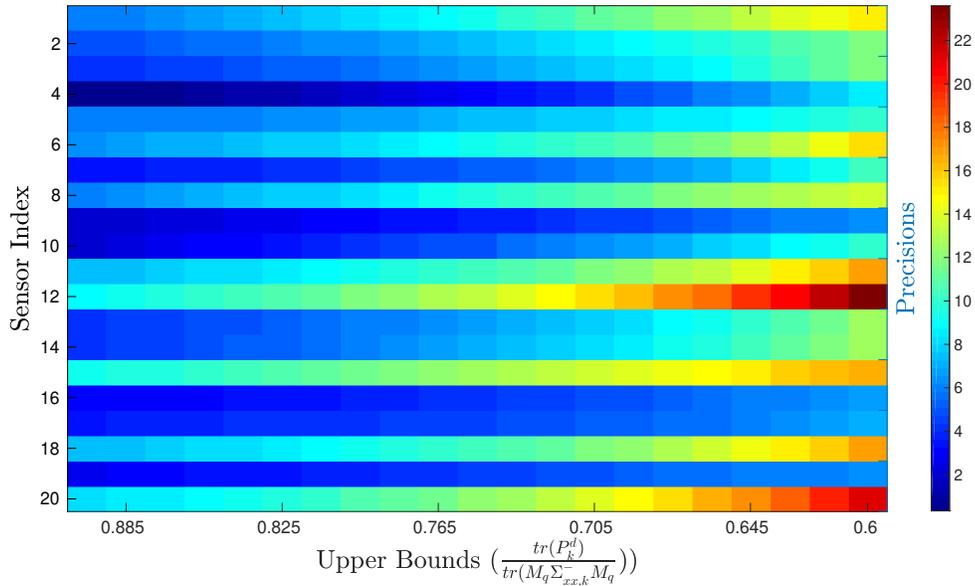
Figure 1.1: Precision of sensors updated at each time step $(q = 1)$ for EnKF without precision bounds. Reprinted with permission from [19].

1.6   Conclusion

In this work, a new sensor precision selection problem for non-linear dynamical systems is presented in the framework of EnKF and UKF. The problem is shown to be convex, which can be easily solved using standard software such as CVX. The algorithm is applied to the Lorenz 1996 model of order 20 and results from both EnKF and UKF framework are presented. Sensor pruning, in the event of very small precisions in the optimal solution, is also discussed and methods to solve them are presented. Future work involves developing new sensor pruning algorithms for non-linear systems, and also investigating impact of EnKF and UKF framework, along with other norm minimizations, on optimality and practicality.
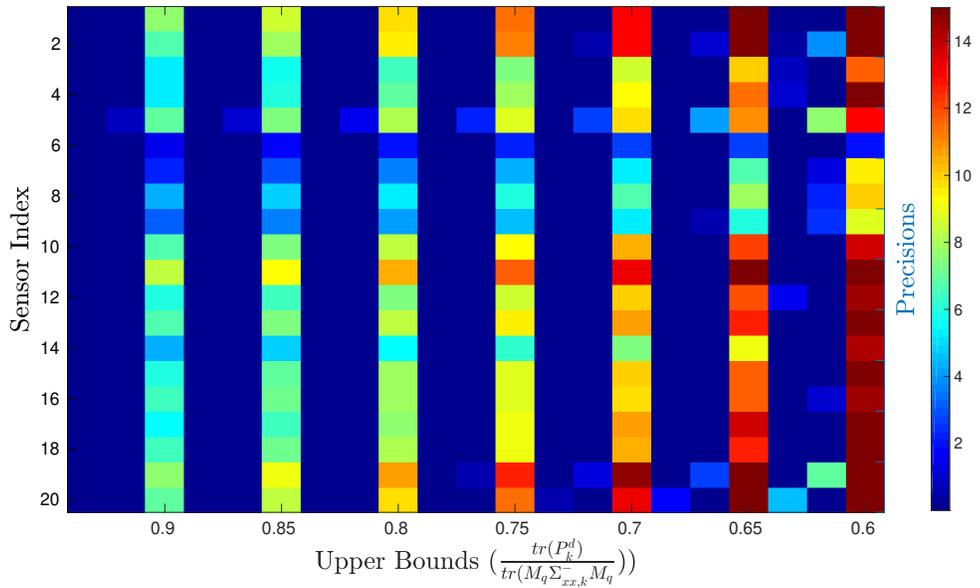
Figure 1.2: Precision of sensors updated for 3 consecutive time step ($q = 3$), with precision bounds for EnKF. Reprinted with permission from [19].
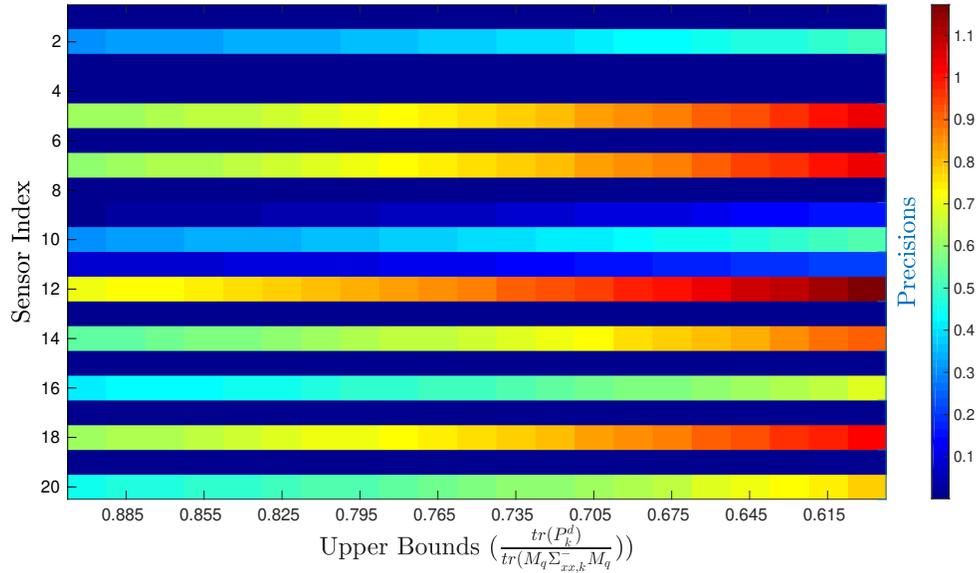
Figure 1.3: Precision of sensors updated at each time step ($q = 1$) for, for UKF without precision bounds. Reprinted with permission from [19].
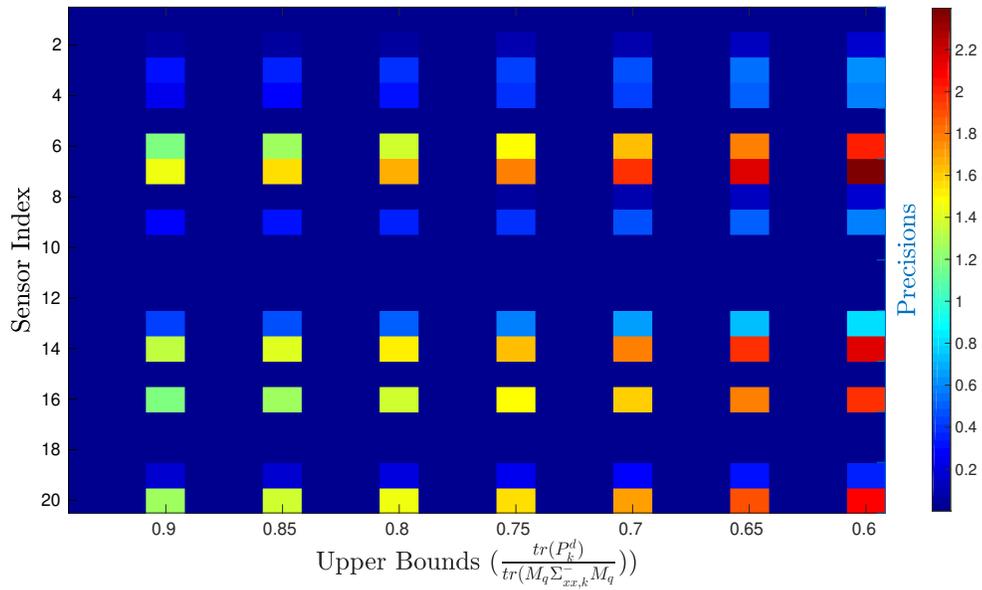
Figure 1.4: Precision of sensors updated for 3 consecutive time step ($q = 3$), for UKF with precision bounds. Reprinted with permission from [19].

2. PRIVACY-UTILITY AWARE KALMAN FILTERING FOR DISCRETE LTI SYSTEMS[*]

2.1 Introduction

The goal of privacy-utility aware output data sharing of a discrete time linear time invariant (D-LTI) stochastic system is establishment of mechanisms that enables states estimation with fidelity (utility) by a data receiver without deeply jeopardizing privacy of the data sharer. This paper deals with a D-LTI stochastic system whose states at time instant k are random variable $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$, are shared as $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ with a data receiver through an linear stochastic observation function. In this work we deal with states $\boldsymbol{x}_k$ that are partitioned into $\{\boldsymbol{x}_k^{(p)} := \text{public}, \boldsymbol{x}_k^{(q)} := \text{private}\}$ states. The objective of the paper is to design algorithms to enhance utility of the public states while ensuring privacy of the private states, when a Kalman filter in its steady state, is used by the data receiver to estimate the states. Although there are some recent progress on dynamic data privacy and Kalman filtering (KF), partitioning state data into private and public data is relatively new. Our proposed approaches is shown to be applicable to such privacy-utility problems as health monitoring, traffic management , smart meters, all of which is encompassed by the domain of Internet of Things (IOT), just to name a few. With the evolution and development of the 5th generation (5G) technology, Internet of Things (IoT) within 5G provides a foundation and opportunity for smart home and smart healthcare [81].

---

[*]Part of the data is reprinted with permission from Eigen Value Analysis in Lower Bounding Uncertainty of Kalman Filter Estimates by Niladri Das and Raktim Bhattacharya, presented at IFAC World Congress 2020, [18], Copyright 2020 by International Federation of Automatic Control.

The idea of using privacy and utility preserving mechanisms for D-LTI stochastic system being estimated by a Kalman filter in its steady state is the following. First, we need to define proper metrics for privacy and utility and combine them to construct a joint privacy-utility payoff function. We cannot treat them separately since the private and public states are coupled with each other though the D-LTI stochastic dynamics. The payoff function is based on the state error covariance matrix of the KF. The data receiver who has the perfect knowledge of the system dynamics, should be allowed to estimate the public steady-states with minimal state error covariance , while preventing it from estimating the private steady-states with maximal state error covariance. After that we need to identify the free parameters that effects the privacy-utility payoff function. We have identified three free parameters: measurement noise, the linear transformation operator in the measurement function, and sensor selection. In this paper we are interested in the first two parameters. The idea of regulating measurement noise to effect the error covariance matrix of steady state KF has been addressed before in [50] as a parameterization of steady state observer. The algorithm ensures upper bound on the error covariance which ensures utility of the measurement data. This work do not address the lower bounding the error covariance which is essential to ensuring privacy.

The idea of linearly transforming the measurements is not new. This is a standard procedure in compressed sensing [24] and has been widely studied in the context of accurate data recovery. In [73] the authors presented a linear transform to map the measurement space into a lower dimensional space. The results presented are only applicable for single step or two step time update. Their scheme although ensures optimal privacy-utility payoff, fails to recover the transformation matrix. Though their methodology do not provide the transformation matrix, their definition

of privacy-utility payoff provides a good starting point for our framework. The third free parameter which is sensor selection is loosely related to the measurement transform method. In [73] the authors have presented a masking matrix constructed using $\{0, 1\}$ that does the operation of sensor selection. It is evident that the sensor selection problem is a contained version of linear transform of the measurements.

The privacy problem has been studied widely in the context of different problems. With the emergence of the data driven information technologies, there are increasing concern over the breach of privacy of personal data collected from sensors. In general, privacy can be categorized into two classes: *data privacy* and *inference privacy*. Data privacy protected the original measurements from being inferred by the fusion center. In the context to data privacy, techniques such as generalization and bucketization have been designed to provide privacy protection. But without care this reduces the utility of the data. It is hence important to carefully study the tradeoff between privacy and utility. Other privacy metrics that have been proposed includes homomorphic encryption and local differential privacy. Inference privacy prevents fusion center from making certain statistical inferences. The privacy metric that have been proposed for inference privacy includes information privacy, differential privacy and average information leakage. The privacy that we consider in this paper belongs to inference privacy.

We need a crisp and clear definition of privacy in the context of state estimation using measurement data. Privacy-utility preserving solution provide support in different ways. But they do not have robust theoretical basis for both privacy and utility. Such a basis is important for several reasons. First, a theoretical abstraction allows us to recast the problem in a technology-independent manner. Second, a theoretical framework enables us to examine the costs of lost privacy against the benefits of

25

data dissemination, namely, the tradeoff between privacy and utility. It would be desirable to give each customer the ability to decide that tradeoff and also to give the third party (service provider) the ability to incentivize the customer to participate in such a bargain by offering interesting points of tradeoff. Finally, a theoretical framework for privacy and utility may expose points of tradeoff that are unexpected. We propose a theoretical framework for privacy-utility preserving in the context of inferencing using KF on a LTI system's measurements. We provide two independent schemes based on either manipulating the measurement noise covariance or linearly compressing the measurement space. In contrast to the works mentioned before, we deal with upper and lower bound of the error covariance matrix in the same framework, connect it to the privacy-utility payoff function, consider the steady state scenario compared to the single step, two step or over finite time horizon Kalman update of others. In the first scheme of manipulating the measurement noise covariance we present new results completely based on Eigen-value analysis. This approach significantly reduces the complexity of similar related results presented in [50] only for the upper bound of error covariance matrix. Mesurement compression technique for the steady state KF although new, has been inspired by the ideas presented in [73].

On a formulation level, if the system dynamics for states $\boldsymbol{x}$ is: $\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{w}_k$, $\forall k \in \mathbb{N}$, and the measurement equation for measurements $\boldsymbol{y}$ is: $\boldsymbol{y}_k = \boldsymbol{H}^T\boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{n}_k$, $\forall k \in \mathbb{N}$, whose Kalman filtering based covariance update equation is:

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{H} + \boldsymbol{R})^{-1}\boldsymbol{H}^T\boldsymbol{C}\boldsymbol{P}_{k|k-1}$$

for $\boldsymbol{P}_{k|k-1}$ and $\boldsymbol{P}_{k|k}$ denoting the prior and posterior covariance matrix, the question that we are interested in answering is: manipulate $\boldsymbol{R}$ or $\boldsymbol{H}^T$ matrix such that

a certain privacy-utility tradeoff is satisfied. Details on the variables used and the privacy-utility definitions are given in the succeeding sections. Regulating $\boldsymbol{R}$ is akin to adding synthetic measurement noise, scheduling sensing regimes, or deciding noise intensity for active sensors such as lidar or laser. On the other hand linear transformation of system measurement using $\boldsymbol{H}^T$ matrix falls under the purview of compressed sensing. Both of these privacy-utility preserving schemes are explored in this work. This chapter is organized as follows. In Section 2.2 we describe the system model followed by the description of Kalman Filtering in Section 2.3. Then we introduce the concept of privacy and utility in Section 2.4, followed by problems statements in Section 2.6 and Section 2.7. Next we provide one numerical example in Section 2.8 and conclude.

Notation: Let $\mathbb{N}$ and $\mathbb{R}$ ($\mathbb{R}_+$) represent the sets of natural number and real (positive real) numbers respectively. The state space of system $\mathcal{X}$ is a closed set in $\mathbb{R}^{n_x}$, where n_x is the dimension of the states. Transpose of a square matrix $\mathbf{M} \in \mathbb{R}^{n \times n}$ is denoted as $\mathbf{M}^T$. A positive definite (semi-definite) matrix $\mathbf{M}$ is denoted by $\mathbf{M} \succ 0$ ($\mathbf{M} \succeq 0$) and $\mathbf{M} \succ \boldsymbol{N}$ ($\mathbf{M} \succeq \boldsymbol{N}$) if $\mathbf{M} - \mathbf{N} \succ 0$ ($\mathbf{M} - \mathbf{N} \succeq 0$), for some matrix $\mathbf{N} \in \mathbb{R}^{n \times n}$. The set of all positive definite (semi-definite) matrices of size $n \times n$ is denoted by $\mathbb{S}_n^{++}$ ($\mathbb{S}_n^+$). Let $\lambda_i(\mathbf{M})$ denotes i^{th} eigen value of the matrix $\mathbf{M}$, when we arrange them as $\lambda_1(\mathbf{M}) \geq \lambda_2(\mathbf{M}) \geq ... \geq \lambda_n(\mathbf{M})$. Similarly, singular values $\sigma_i(\mathbf{M})$ of $\mathbf{M}$, are arranged in non-increasing order: $\sigma_1(\mathbf{M}) \geq \sigma_2(\mathbf{M}) \geq ... \geq \sigma_n(\mathbf{M})$. Let $\text{diag}(\mathbf{a})$ denotes a diagonal matrix, with $\mathbf{a}$ as its diagonal elements. We assume that $\boldsymbol{x} \in \mathcal{X}$ is continuous, $\mu(\boldsymbol{x})$ is a Lebesgue measure, and $\text{p}(\boldsymbol{x})$ is the probability density function (pdf). The expected value of the random variable $\boldsymbol{x}$ with respect to $\text{p}(\boldsymbol{x})$ is represented as $\mathbb{E}\left[\boldsymbol{x}\right]$.

2.2 System dynamics and measurement model

We focus on the class of *discrete-time linear time-invariant* stochastic systems. Let $\boldsymbol{x}_k$ represent the true states of a system at the kth time instant, where $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ for all $k \in \mathbb{N}$. The dynamics is modeled as

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{w}_k, \ \forall k \in \mathbb{N}, \tag{2.1}$$

where $\boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}$ is the state transition matrix. The process noise variable $\boldsymbol{w}_k \in \mathbb{R}^{n_w}$, is the n_w dimensional zero-mean Gaussian additive noise with $\mathbb{E}[\boldsymbol{w}_k \boldsymbol{w}_l^T] = \boldsymbol{Q}\delta_{kl}$. The discrete dynamics in (3.1a) is observed by a linear measurement model. Let $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ denote the measurement taken at the kth time instant as

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{n}_k, \ \forall k \in \mathbb{N}, \tag{2.2}$$

where $\boldsymbol{y}_k$ is corrupted by a n_y dimensional additive observation noise $\boldsymbol{n}_k \in \mathbb{R}^{n_n}$. The sensor noise at each time instant is a zero-mean Gaussian random variable with $\mathbb{E}[\boldsymbol{n}_k \boldsymbol{n}_l^T] = \boldsymbol{R}\delta_{kl}$. The matrix $\boldsymbol{C} \in \mathbb{R}^{n_y \times n_x}$ is known as the observation or the measurement or the generative matrix.

The initial state of (3.1a) is modeled by a Gaussian random variable $\boldsymbol{x}_0$ with mean $\boldsymbol{\mu}_0$ and covariance $\mathbf{P}_0$. The random variable $\boldsymbol{x}_0$ denotes the system state at the 0th time instant. The process noise $\boldsymbol{w}_k$, observation noise $\boldsymbol{n}_k$, and initial state variable $\boldsymbol{x}_0$ are all assumed to be independent, unless otherwise specified. These assumptions are strongly motivated by analytical tractability. The restriction to zero-mean noise sources is not a loss of generality. When the noise sources are not zero-mean, the $\boldsymbol{A}$ and $\boldsymbol{C}$ matrices are modified and extra states are introduced.

2.3 Kalman filtering

The discrete time system in (3.1a) and (3.1b) induces a Kalman filter, as the optimal state estimator with dynamics

Propagation –

$$\boldsymbol{\mu}_k^- = \boldsymbol{A}\boldsymbol{\mu}_{k-1}^+, \hspace{4cm} \text{(Mean Propagation)}$$

$$\mathbf{P}_k^- = \boldsymbol{A}\mathbf{P}_{k-1}^+\boldsymbol{A}^T + \boldsymbol{Q}, \hspace{3cm} \text{(Covariance Propagation)}$$

Update –

$$\boldsymbol{\mu}_k^+ = \boldsymbol{A}\boldsymbol{\mu}_{k-1}^+ + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\boldsymbol{\mu}_k^-), \hspace{2.5cm} \text{(Mean Update)}$$

$$\mathbf{P}_k^+ = (\mathbf{I} - \boldsymbol{K}_k\boldsymbol{C})\mathbf{P}_k^-, \hspace{3.5cm} \text{(Covariance Update)}$$

$$\boldsymbol{K}_k = \mathbf{P}_k^-\boldsymbol{C}^T\left[\boldsymbol{C}\mathbf{P}_k^-\boldsymbol{C}^T + \mathbf{R}\right]^{-1}, \hspace{2.5cm} \text{(Kalman Gain)}$$

$$\boldsymbol{\mu}_0^+ = \boldsymbol{\mu}_0, \hspace{4.5cm} \text{(Initial State Mean)}$$

$$\mathbf{P}_0^+ = \mathbf{P}_0, \hspace{4.5cm} \text{(Initial State Covariance)}$$

where the variables $\boldsymbol{\mu}_k^-, \boldsymbol{\mu}_k^+ \in \mathbb{R}^{n_x}$, denote the prior and posterior mean estimates of the random variable $\boldsymbol{x}_k$. $\boldsymbol{K}_k$ is the Kalman gain, at time k. The positive semi-definite matrices $\mathbf{P}_k^-, \mathbf{P}_k^+ \in \mathbb{R}^{n_x \times n_x}$ are the prior and posterior co-variance matrices at time instant k respectively. We define the matrix inverse of the observation noise co-variance $\mathbf{R}$ as the *precision matrix* $\boldsymbol{S}$, i.e. $\boldsymbol{S} := \mathbf{R}^{-1}$.

2.4 Privacy and utility problem

The primary challenge in characterizing the privacy-utility tradeoffs is creating the right abstraction – we need a principled approach that provides quantitative measures of both the amount of information leaked as well as the utility retained and provides

a basis for a negotiated level of benefit for both consumer and supplier.

We assume that the true state $\boldsymbol{x}_k$ can be partitioned into two parts as $\boldsymbol{x}_k = \{\boldsymbol{x}_k^{(p)}, \boldsymbol{x}_k^{(q)}\}$, where $\boldsymbol{x}_k^{(p)} \in \mathbb{R}^{L_{pub}}$ contains public states that are shareable with others whereas $\boldsymbol{x}_k^{(q)} \in \mathbb{R}^{L_{pri}}$ represents private states containing sensitive information that should be accessible only to authorized users. The variables are related to each through two masking matrices $\boldsymbol{M}^{(p)}$ and $\boldsymbol{M}^{(q)}$ as:

$$\boldsymbol{x}_k^{(p)} = \boldsymbol{M}^{(p)} \boldsymbol{x}_k \ ; \quad \boldsymbol{x}_k^{(q)} = \boldsymbol{M}^{(q)} \boldsymbol{x}_k$$

The corresponding steady state (prior) error covariance matrices are:

$$\boldsymbol{P}_{ss}^{(p)} = \boldsymbol{M}^{(p)} \boldsymbol{P}_{ss} \boldsymbol{M}^{(p)^T} \ ; \quad \boldsymbol{P}_{ss}^{(q)} = \boldsymbol{M}^{(q)} \boldsymbol{P}_{ss} \boldsymbol{M}^{(q)^T}$$

The public error covariance matrix $\boldsymbol{P}_{ss}^{(p)} \in \mathbb{R}^{L_p \times L_p}$ whereas $\boldsymbol{P}_{ss}^{(q)} \in \mathbb{R}^{L_q \times L_q}$ and $\mathrm{Tr}(.)$ denotes trace operator. We define the privacy metric to be $\mathrm{Tr}(\boldsymbol{P}_{ss}^{(q)})$.

2.5   Problem formulation

There are three distinct problems that we address here. These problems are explicitly dependent upon determining three tune-able parameters that effect the privacy or utility metric, which are $\mathbf{R}$, $\boldsymbol{K}_{ss}$, and $\boldsymbol{C}$, where $\boldsymbol{K}_{ss}$ is the steady state Kalman gain matrix. Changing $\mathbf{R}$ is commonly referred to as perturbing measurement noise intensity in privacy-utility literature as one of the privacy preserving mechanisms. In this case one sanitizes the available measurements with spurious noises. Measurement compression is done by manipulating the $\boldsymbol{C}$ matrix. We see the perturbation in $\boldsymbol{K}_{ss}$

matrix in cases where only the Kalman Gain matrix is made available along with the measurements to update the estimates on the consumer side ensuring desired level of privacy and (or) utility.

2.6   Problem statement I

We assume that the system matrices $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ and noise parameter $\boldsymbol{Q}$ in (3.1a) and (3.1b) are known. The matrix $\boldsymbol{R}$ which is the sensor noise covariance, is the design variable. For a prescribed upper or lower bound on the *steady state of public and private variable's prior covariance* using the Kalman filter, we are to design $\boldsymbol{R}$ or the precision matrix $\boldsymbol{S} := \boldsymbol{R}^{-1}$, that satisfies these bounds.

2.6.1   Preliminaries

In the steady state scenario of a Kalman filter, the prior state covariance $\boldsymbol{P}$ is related to our design variable $\boldsymbol{R}$ through the Discrete Algebraic Riccati Equation (DARE). The following preliminary discussions aim to connect the eigenvalues of $\boldsymbol{P}$ with that of $\boldsymbol{R}$, leading to the Linear Matrix Inequalities (LMIs), that helps in synthesizing the $\boldsymbol{R}$ matrix.

2.6.1.1   Eigen-Value Based Analysis

Middleton and Goodwin in [61] introduced the Unified Algebraic Riccati Equation :

$$\boldsymbol{PA} + \boldsymbol{A}^T\boldsymbol{P} + \Delta\boldsymbol{A}^T\boldsymbol{PA} - (\Delta\boldsymbol{A}^T + \boldsymbol{I})\boldsymbol{PB}$$
$$\times (\boldsymbol{I} + \Delta\boldsymbol{B}^T\boldsymbol{PB})^{-1}\boldsymbol{B}^T\boldsymbol{P}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \boldsymbol{Q} = 0, \tag{2.3}$$

31

where $\boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}$ and $\boldsymbol{B} \in \mathbb{R}^{n_x \times n_y}$ represent constant matrices, $\boldsymbol{Q} \in \mathbb{R}^{n_x \times n_x}$ is in $\mathbb{S}_n^+$, the matrix $\boldsymbol{P} \in \mathbb{R}^{n_x \times n_x}$ is the positive definite solution to (2.3), and Δ represents sampling period.

We introduce an extra parameter $\boldsymbol{R} \in \mathbb{R}^{n_y \times n_y}$ in UARE and call it UARE-R. This UARE-R:

$$\boldsymbol{PA} + \boldsymbol{A}^T\boldsymbol{P} + \Delta\boldsymbol{A}^T\boldsymbol{PA} - (\Delta\boldsymbol{A}^T + \boldsymbol{I})\boldsymbol{PB}$$
$$\times (\boldsymbol{R} + \Delta\boldsymbol{B}^T\boldsymbol{PB})^{-1}\boldsymbol{B}^T\boldsymbol{P}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \boldsymbol{Q} = 0, \tag{2.4}$$

is often encountered in Optimal Control [10] and Estimation [4] problems.

Remark 1. (a) Using $\Delta = 0$, replacing $\boldsymbol{A}$ by $\boldsymbol{A}^T$, and $\boldsymbol{B}$ by $\boldsymbol{C}^T$, we recover the Continuous Time Algebraic Riccati Equation (CARE), solution to which gives us the steady state covariance for a Kalman-Bucy filter. (b) Using $\Delta = 1$, replacing $\boldsymbol{A} + \boldsymbol{I}$ by $\boldsymbol{A}^T$, and $\boldsymbol{B}$ by $\boldsymbol{C}^T$ we recover the DARE associated with steady state covariance of the Kalman Filter, where $\boldsymbol{P}$ denotes the steady-state error covariance matrix.

Reiterating, our objective is to design $\boldsymbol{R}$ matrix that satisfies prescribed bounds on the steady-state estimated state error covariance matrix $\boldsymbol{P}$, using (2.4). We closely follow the calculations in [49] to relate the bounds on $\boldsymbol{R}$ with that of $\boldsymbol{P}$ in (2.4).

2.6.1.2   Preliminary results:

In the following three theorems we examine the charactertization of the lower and upper bounds on the $\boldsymbol{P}$ matrix of the UARE-R, as a function of $\boldsymbol{R}$. As our final results we will provide theorems that will connect the eigen values of $\boldsymbol{R}$ to upper or lower bounds on $\boldsymbol{P}$. This opens up a way to generate the feasible set for choosing $\boldsymbol{R}$

matrix.

Theorem 2. *Let $\boldsymbol{P}$ be the positive solution of the UARE-R in equation (2.4), then*

$$\boldsymbol{P} \preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\boldsymbol{P}_{u0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}, \qquad (2.5)$$

where $\boldsymbol{P} \in \mathbb{S}_{n_x}^+$, $\boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}$, $\boldsymbol{B} \in \mathbb{R}^{n_x \times n_y}$, $\boldsymbol{R} \in \mathbb{S}_{n_y \times n_y}^{++}$, $\boldsymbol{Q} \in \mathbb{S}_{n_x \times n_x}^+ \neq \boldsymbol{0}_{n_x \times n_x}, \Delta \geq 0$,

$$\boldsymbol{P}_{u0} \equiv (\Delta\boldsymbol{A} + \boldsymbol{I})^T \left(\frac{\boldsymbol{I}}{\eta} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\right)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I})$$

$$+ \Delta\boldsymbol{Q}, \qquad (2.6)$$

$$\eta \equiv \frac{-a + \sqrt{a^2 + bc}}{b} > 0,$$

$$a \equiv -\lambda_{max}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$

$$- \Delta\lambda_{max}(\boldsymbol{Q})\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$b \equiv 2\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T), c \equiv 2\lambda_{max}(\boldsymbol{Q}).$$

Proof. Using

$$\Delta(\boldsymbol{P}\boldsymbol{A} + \boldsymbol{A}^T\boldsymbol{P} + \Delta\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A}) = (\Delta\boldsymbol{A} + \boldsymbol{I})^T\boldsymbol{P}(\Delta\boldsymbol{A} + \boldsymbol{I}) - \boldsymbol{P}, \qquad (2.7)$$

in UARE-R

$$(\Delta\boldsymbol{A} + \boldsymbol{I})^T[\boldsymbol{P} - \Delta\boldsymbol{P}\boldsymbol{B}(\boldsymbol{R} + \Delta\boldsymbol{B}^T\boldsymbol{P}\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{P}](\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} = \boldsymbol{P}. \qquad (2.8)$$

followed by matrix inversion lemma

$$(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} = \boldsymbol{P}. \qquad (2.9)$$

Following [49]

$$\boldsymbol{P} \preceq \lambda_{max}(\boldsymbol{P})\boldsymbol{I}; \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T \succeq \lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T)\boldsymbol{I}$$

$$\boldsymbol{P} \preceq \frac{\lambda_{max}(\boldsymbol{P})}{1 + \lambda_{max}(\boldsymbol{P})\Delta\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}. \quad (2.10)$$

Using $\lambda_{max}(\boldsymbol{A}_1 + \boldsymbol{B}_1) \leq \lambda_{max}(\boldsymbol{A}_1) + \lambda_{max}(\boldsymbol{B}_1)$ from [53] where $\boldsymbol{A}_1, \boldsymbol{B}_1$ are symmetric matrices

$$\lambda_{max}(\boldsymbol{P}) \leq \frac{\lambda_{max}(\boldsymbol{P})}{1 + \lambda_{max}(\boldsymbol{P})\Delta\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T)}$$
$$\times \lambda_{max}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] + \Delta\lambda_{max}(\boldsymbol{Q}).$$

Using $\lambda_{max}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] = \Delta\lambda_{max}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + 1$ we get a quadratic in $\lambda_{max}(\boldsymbol{P})$

$$\frac{b}{2}\lambda_{max}^2(\boldsymbol{P}) + a\lambda_{max}(\boldsymbol{P}) - \frac{c}{2} \leq 0,$$

where

$$a \equiv -\lambda_{max}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) - \Delta\lambda_{max}(\boldsymbol{Q})\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$b \equiv 2\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$c \equiv 2\lambda_{max}(\boldsymbol{Q}).$$

which denotes

$$0 \leq \lambda_{max}(\boldsymbol{P}) \leq \frac{-a + \sqrt{a^2 + bc}}{b} \equiv \eta.$$

In (2.9)

$$\boldsymbol{P} \preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\lambda_{max}^{-1}(\boldsymbol{P})\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q},$$

$$\preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\eta^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{u0}. \qquad (2.11)$$

Putting it back to the modified UARE-R in equation (2.9), we get,

$$\boldsymbol{P} \preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\boldsymbol{P}_{u0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{u1}$$

$\boldsymbol{P}_{u1}$ is a more tighter bound from above compared to $\boldsymbol{P}_{u0}$. $\qquad\qquad\qquad$ □

Remark 2. If $\boldsymbol{Q} = \boldsymbol{0}$ and a positive solution of UARE-R exists, we have $\lambda_{max}(\boldsymbol{Q}) = 0$.

$$\lambda_{max}(\boldsymbol{P}) \leq \frac{\lambda_{max}(\boldsymbol{P})}{1 + \lambda_{max}(\boldsymbol{P})\Delta\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T)}$$

$$\times \lambda_{max}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})]$$

$$\lambda_{max}(\boldsymbol{P})(1 + \lambda_{max}(\boldsymbol{P})\Delta\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T))$$

$$\leq \lambda_{max}(\boldsymbol{P})\lambda_{max}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})]$$

$$\lambda_{max}(\boldsymbol{P})\Delta\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T) \leq \lambda_{max}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] - 1$$

$$\lambda_{max}(\boldsymbol{P}) \leq \frac{\lambda_{max}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] - 1}{\Delta\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T)} \equiv \eta.$$

Similarly it follows that if $\boldsymbol{R} = \boldsymbol{0}$ and a positive solution of UARE-R exists, we have $\boldsymbol{P} \preceq \Delta\boldsymbol{Q}$.

Theorem 3. *Let $\boldsymbol{P}$ be the positive solution of the UARE-R in equation (2.4), then*

$$\boldsymbol{P} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\boldsymbol{P}_{l0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \tag{2.12}$$

where $\boldsymbol{P} \in \mathbb{S}_{n_x}^+, \boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}, \boldsymbol{B} \in \mathbb{R}^{n_x \times n_y}, \boldsymbol{R} \in \mathbb{S}_{n_y \times n_y}^{++}, \boldsymbol{Q} \in \mathbb{S}_{n_x \times n_x}^+ \neq \boldsymbol{0}_{n_x \times n_x}, \Delta \geq 0,$

$$\boldsymbol{P}_{l0} \equiv (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\varphi^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I})$$

$$+ \Delta\boldsymbol{Q},$$

$$\varphi \equiv \frac{-a + \sqrt{a^2 + bc}}{b},$$

$$a \equiv -\lambda_{min}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$

$$- \Delta\lambda_{min}(\boldsymbol{Q})\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$b \equiv 2\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T), c \equiv 2\lambda_{min}(\boldsymbol{Q}).$$

Proof. Following [49]

$$\boldsymbol{P}^{-1} \preceq \lambda_{min}^{-1}(\boldsymbol{P})\boldsymbol{I}. \tag{2.13}$$

$$\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T \preceq \lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T)\boldsymbol{I}. \tag{2.14}$$

leading to

$$\boldsymbol{P} \succeq \frac{\lambda_{min}(\boldsymbol{P})}{1 + \lambda_{min}(\boldsymbol{P})\Delta\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}. \tag{2.15}$$

Using $\lambda_n(\boldsymbol{A}_1 + \boldsymbol{B}_1) \geq \lambda_n(\boldsymbol{A}_1) + \lambda_n(\boldsymbol{B}_1)$ from [53] where $\boldsymbol{A}_1, \boldsymbol{B}_1$ are symmetric

matrices

$$\lambda_{min}(\boldsymbol{P}) \geq \frac{\lambda_{min}(\boldsymbol{P})}{1 + \lambda_{min}(\boldsymbol{P})\Delta\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T)}$$
$$\times \lambda_{min}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] + \Delta\lambda_{min}(\boldsymbol{Q}).$$

Using $\lambda_{min}[(\Delta\boldsymbol{A}+\boldsymbol{I})^T(\Delta\boldsymbol{A}+\boldsymbol{I})] = \Delta\lambda_{min}(\boldsymbol{A}+\boldsymbol{A}^T+\Delta\boldsymbol{A}^T\boldsymbol{A})+1$ we get a quadratic in $\lambda_{min}(\boldsymbol{P})$ inequality

$$\frac{b}{2}\lambda_{min}^2(\boldsymbol{P}) + a\lambda_{min}(\boldsymbol{P}) - \frac{c}{2} \geq 0, \tag{2.16}$$

where

$$a \equiv -\lambda_{min}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$
$$- \Delta\lambda_{min}(\boldsymbol{Q})\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T),$$
$$b \equiv 2\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T), c \equiv 2\lambda_{min}(\boldsymbol{Q}).$$

which is of quadratic form. This denotes

$$\lambda_{min}(\boldsymbol{P}) \geq \frac{-a + \sqrt{a^2 + bc}}{b} \equiv \varphi.$$

Following from

$$(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} = \boldsymbol{P}, \tag{2.17}$$

we have

$$\boldsymbol{P} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\varphi^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{l0}. \tag{2.18}$$

Using the lower bound $\boldsymbol{P}_{l0}$ of in equation (2.18) we get,

$$\boldsymbol{P} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \tag{2.19}$$

$\square$

Remark 3. If $\boldsymbol{Q} = \boldsymbol{0}$ and a positive solution of UARE-R exists, we have $\lambda_{min}(\boldsymbol{Q}) = 0$.

$$\lambda_{min}(\boldsymbol{P}) \geq \frac{\lambda_{min}(\boldsymbol{P})}{1 + \lambda_{min}(\boldsymbol{P})\Delta\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T)}$$

$$\times \lambda_{min}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})]$$

$$\lambda_{min}(\boldsymbol{P}) \geq \frac{\lambda_{min}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] - 1}{\Delta\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T)} \equiv \varphi.$$

Similarly it follows that if $\boldsymbol{R} = \boldsymbol{0}$ and a positive solution of UARE-R exists, we have $\boldsymbol{P} \succeq \Delta\boldsymbol{Q}$.

2.6.2 Calculating Noise Intensity $\boldsymbol{R}$

We discussed how we retrieve DARE:

$$\boldsymbol{A}\boldsymbol{P}\boldsymbol{A}^T - \boldsymbol{P} - \boldsymbol{A}\boldsymbol{P}\boldsymbol{C}^T(\boldsymbol{R} + \boldsymbol{C}\boldsymbol{P}\boldsymbol{C}^T)^{-1}\boldsymbol{C}\boldsymbol{P}\boldsymbol{A}^T + \boldsymbol{Q} = 0,$$

for solving the steady-state covariance matrix $\boldsymbol{P}$ for Kalman filter, applying suitable substitution to the UARE-R. Conventionally, designing $\boldsymbol{R}$ is related to upper bounding the performance of a filter with some additional constraints. Topics like differential privacy and bounded information exchange such as in robotics has lead to the requirement of switching between different $\boldsymbol{R}$ matrices to keep the performance within bounds (upper or lower), rather than just upper bounding it which has been typically done in problems related to sensing architecture design. In this work, we utilize Thorem 2 and 3 to propose a technique to design the measurement noise covariance matrix $\boldsymbol{R}$ or the precision matrix $\boldsymbol{S}$ such that the $\boldsymbol{P}$ is upper or lower bounded. We will see in the succeeding sections that the feasible set of $\boldsymbol{S}$ is represented as a set of LMIs.

In the following results we first construct the feasible set of $\boldsymbol{S}$ that satisfies prescribed upper or lower bound on the matrix $\boldsymbol{P}$. A particular choice of $\boldsymbol{S}$ matrix results from an optimization problem over the set of feasible $\boldsymbol{S}$ for a given cost function. We use $c(\boldsymbol{S})$ to represent a generic cost function.

Remark 4. If $\boldsymbol{S}$ is a diagonal matrix, the cost function is essentially over the space of vector $\boldsymbol{\lambda}$, that constitutes the diagonal elements of $\boldsymbol{S}$.

2.6.3 Calculate $\boldsymbol{R}$ for lower bound on steady state Kalman filter state error covariance of private state $\boldsymbol{x}_k^{(p)}$

Using $\Delta = 1$, replacing $\boldsymbol{A} + \boldsymbol{I}$ by $\boldsymbol{A}^T$ and $\boldsymbol{B}$ by $\boldsymbol{C}^T$ in UARE-R, we recover the Discrete Algebraic Riccati Equation (DARE) associated with steady state covariance update equation of a linear system using Kalman Filter.

We assume complete detectability of $[\boldsymbol{A}, \boldsymbol{C}]$ and stabilizability of $[\boldsymbol{A}, \boldsymbol{Q}^{1/2}]$ ([4], pg.82) for (3.1a) and (3.1b). This ensure that the steady state prior covariance matrix $\boldsymbol{P}$ exists and is unique (for a fixed $\boldsymbol{R}$) for the corresponding DARE.

Theorem 4. *For a given scalar cost function $c(\boldsymbol{R})$ and an lower bound $(1/\lambda_u^f)$ on the spectrum of $\boldsymbol{R}$, the solution $\boldsymbol{R}^*$, whose spectrum is*

$$\lambda(\boldsymbol{R}^*) := \{\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{ny-1} \geq \lambda_{ny}\},$$

where $\lambda_{ny} \geq (1/\lambda_u^f)$, that satisfies a given lower bound $\boldsymbol{P}_l^f$ on the steady state prior covariance matrix $\boldsymbol{P}^{(p)} := \boldsymbol{M}^{(p)} \boldsymbol{P} \boldsymbol{M}^{(p)T}$ of Kalman filter, is given by the following optimization problem.

$$\boldsymbol{R}^* := \operatorname*{argmin}_{\boldsymbol{R}} \ c(\boldsymbol{R})$$

Such that,

$$\boldsymbol{R} \succeq \frac{1}{\lambda_u^f} \boldsymbol{I}, \quad \begin{bmatrix} \boldsymbol{T}_1 & \boldsymbol{T}_2 \\ \boldsymbol{T}_2^T & \boldsymbol{T}_4 \end{bmatrix} \succeq 0$$

where,

$$\boldsymbol{T}_1 = \boldsymbol{M}^{(p)} \boldsymbol{A} \boldsymbol{P}_{l0}' \boldsymbol{A}^T \boldsymbol{M}^{(p)T} - \boldsymbol{P}_l^f + \boldsymbol{M}^{(p)} \boldsymbol{Q} \boldsymbol{M}^{(p)T}$$

$$\boldsymbol{T}_2 = \boldsymbol{M}^{(p)} \boldsymbol{A} \boldsymbol{P}_{l0}' \boldsymbol{C}^T, \boldsymbol{T}_4 = \boldsymbol{R} + \boldsymbol{C} \boldsymbol{P}_{l0}' \boldsymbol{C}^T$$

$$\boldsymbol{P}_{l0}' \equiv \boldsymbol{A}(\varphi'^{-1} \boldsymbol{I} + \lambda_u^f \boldsymbol{C}^T \boldsymbol{C})^{-1} \boldsymbol{A}^T + \boldsymbol{Q}.$$

$$\varphi' \equiv f(-[\lambda_{min}(\boldsymbol{A}\boldsymbol{A}^T - \boldsymbol{I}) + \lambda_{min}(\boldsymbol{Q})\lambda_u^f \lambda_{max}(\boldsymbol{C}^T \boldsymbol{C})]$$

$$, 2\lambda_u^f \lambda_{max}(\boldsymbol{C}^T \boldsymbol{C}), 2\lambda_{min}(\boldsymbol{Q})),$$

Proof. In Theorem 3

$$\varphi \equiv \frac{-a + \sqrt{a^2 + bc}}{b},$$

$$a \equiv -\lambda_{min}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$

$$- \Delta\lambda_{min}(\boldsymbol{Q})\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$b \equiv 2\lambda_{max}(\boldsymbol{R}^{-1})\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T), c \equiv 2\lambda_{min}(\boldsymbol{Q}).$$

Notice that φ is a function of $\lambda_{max}(\boldsymbol{R}^{-1})$, where $\boldsymbol{R}$ is the design variable. We restrict the solution space of $\boldsymbol{R}$ by assuming $\lambda_{max}(\boldsymbol{R}^{-1}) \leq \lambda_l^f$. We define

$$\varphi' \equiv \frac{-a + \sqrt{a^2 + bc}}{b},$$

$$a \equiv -\lambda_{min}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$

$$- \Delta\lambda_{min}(\boldsymbol{Q})\lambda_l^f\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$b \equiv 2\lambda_l^f\lambda_{max}(\boldsymbol{B}\boldsymbol{B}^T), c \equiv 2\lambda_{min}(\boldsymbol{Q}).$$

It can be proved that $\varphi \geq \varphi'$. We have,

$$\boldsymbol{P}_{l0} \equiv (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\varphi^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I})$$

$$+ \Delta\boldsymbol{Q}$$

$$\boldsymbol{P}_{l0} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\varphi^{-1}\boldsymbol{I} + \Delta\lambda_{max}(\boldsymbol{R}^{-1})\boldsymbol{B}\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q},$$

$$\succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\varphi'^{-1}\boldsymbol{I} + \Delta\lambda_u^f\boldsymbol{B}\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}'_{l0}, \tag{2.20}$$

Using Theorem 3 we have,

$$\boldsymbol{P} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}$$

$$\succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}'^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}.$$

Let $\boldsymbol{M}^{(p)}$ be the privacy masking matrix :

$$\boldsymbol{M}^{(p)}\boldsymbol{P}\boldsymbol{M}^{(p)^T} \succeq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}'^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T} + \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)^T} \succeq \boldsymbol{P}_l^f.$$

$$\boldsymbol{P}^{(p)} \succeq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}'^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T} + \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)^T} \succeq \boldsymbol{P}_l^f.$$

Or,

$$\boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}'^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T} \succeq \boldsymbol{P}_l^f - \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)^T}.$$

Using matrix inversion lemma

$$\boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}' - \Delta\boldsymbol{P}_{l0}'\boldsymbol{B}(\boldsymbol{R} + \Delta\boldsymbol{B}^T\boldsymbol{P}_{l0}'\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{P}_{l0}')$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T} \succeq \boldsymbol{P}_l^f - \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)^T},$$

$$\boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T\boldsymbol{P}_{l0}'(\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T} - \boldsymbol{P}_l^f + \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)^T}$$

$$- \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T\Delta\boldsymbol{P}_{l0}'\boldsymbol{B}(\boldsymbol{R} + \Delta\boldsymbol{B}^T\boldsymbol{P}_{l0}'\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{P}_{l0}'$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T} \succeq 0.$$

Using Shurs complement,

$$\begin{bmatrix} \boldsymbol{T}_1 & \boldsymbol{T}_2 \\ \boldsymbol{T}_2^T & \boldsymbol{T}_4 \end{bmatrix} \succeq 0,$$

where

$$\boldsymbol{T}_1 = \boldsymbol{M}^{(p)}(\Delta \boldsymbol{A} + \boldsymbol{I})^T \boldsymbol{P}'_{l0}(\Delta \boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)T} - \boldsymbol{P}_l^f + \boldsymbol{M}^{(p)}\Delta \boldsymbol{Q}\boldsymbol{M}^{(p)T},$$

$$\boldsymbol{T}_2 = \boldsymbol{M}^{(p)}(\Delta \boldsymbol{A} + \boldsymbol{I})^T \sqrt{\Delta} \boldsymbol{P}'_{l0}\boldsymbol{B},$$

$$\boldsymbol{T}_4 = \boldsymbol{R} + \Delta \boldsymbol{B}^T \boldsymbol{P}'_{l0}\boldsymbol{B}.$$

Using $\Delta = 1$, replacing $\boldsymbol{A} + \boldsymbol{I}$ by $\boldsymbol{A}^T$ and $\boldsymbol{B}$ by $\boldsymbol{C}^T$ we get the theorem.

$\square$

2.6.4 Choosing feasible lower bound of $\boldsymbol{P}^{(p)}$ for Kalman filter

The desired covariance bounds on $\boldsymbol{P}^{(p)}$ should be chosen carefully. When system matrices $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ and noise parameter $\boldsymbol{Q}$ is already chosen or are known, there exists an lower bound on the $\boldsymbol{P}$ for any choice of the matrix $\boldsymbol{R}$. Choosing any positive definite matrices, as the desired $\mathbf{P}_l^f$, lower than this bound, will result in an in-feasible solution for the precision matrix. Hence it is important to choose the desired performance bounds accordingly. The prescribed $\mathbf{P}_l^f$ should be above $\mathbf{P}^{lb}$ satisfying the following:

$$\mathbf{P}^{lb} := \boldsymbol{A}(\mathbf{P}^{lb} - \mathbf{P}^{lb}\boldsymbol{C}^T\left[\boldsymbol{C}\mathbf{P}^{lb}\boldsymbol{C}^T\right]^{-1}\boldsymbol{C}\mathbf{P}^{lb})\boldsymbol{A}^T + \boldsymbol{B}\boldsymbol{Q}\boldsymbol{B}^T$$

The matrices $\mathbf{P}^{lb}$ is calculated using $\boldsymbol{R} = \boldsymbol{0}$ in the DARE, which is solved using generalized Shur method [71] on an extended matrix pencil. The corresponding lower bound on $\boldsymbol{P}^{(p)}$ is $\boldsymbol{M}^{(p)}\boldsymbol{P}^{lb}\boldsymbol{M}^{(p)T}$

2.6.5 Calculate $\boldsymbol{S} := \boldsymbol{R}^{-1}$ for upper bound on steady state Kalman filter state error covariance of public state $\boldsymbol{x}_k^{(q)}$

Theorem 5. *For a given scalar cost function $c(\boldsymbol{S})$ and an lower bound λ_l^f on the spectrum of $\boldsymbol{S}$, the solution $\boldsymbol{S}^*$, whose spectrum is*

$$\lambda(\boldsymbol{S}^*) := \{\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{ny-1} \geq \lambda_{ny}\},$$

that satisfies a given upper bound $\boldsymbol{P}_u^f$ on the $\boldsymbol{M}^{(q)}\boldsymbol{P}\boldsymbol{M}^{(q)T}$ matrix, is given by the following optimization problem.

$$\boldsymbol{S}^* := \underset{\boldsymbol{S}}{\operatorname{argmin}}\ c(\boldsymbol{S})$$

Such that,

$$\boldsymbol{S} \succeq \lambda_l^f \boldsymbol{I},$$

$$\begin{bmatrix} \boldsymbol{P}_u^f - \boldsymbol{M}^{(q)}\boldsymbol{Q}\boldsymbol{M}^{(q)T} & \boldsymbol{M}^{(q)}\boldsymbol{A} \\ \boldsymbol{A}^T\boldsymbol{M}^{(q)T} & \boldsymbol{P}'_{u0}{}^{-1} + \boldsymbol{C}^T\boldsymbol{S}\boldsymbol{C} \end{bmatrix} \succeq 0,$$

Proof. In Theorem 2

$$\eta \equiv \frac{-a + \sqrt{a^2 + bc}}{b} > 0,$$

$$a \equiv -\lambda_{max}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$

$$- \Delta\lambda_{max}(\boldsymbol{Q})\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$b \equiv 2\lambda_{min}(\boldsymbol{R}^{-1})\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T), c \equiv 2\lambda_{max}(\boldsymbol{Q}).$$

The η being a monotonically decreasing function with respect to $\lambda_{min}(\boldsymbol{R}^{-1})$ if $\lambda_{min}(\boldsymbol{R}^{-1}) \geq \lambda_l^f$, then $\eta \leq \eta'$, where

$$\eta' \equiv \frac{-a + \sqrt{a^2 + bc}}{b} > 0,$$

$$a \equiv -\lambda_{max}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) - \Delta\lambda_{max}(\boldsymbol{Q})\lambda_l^f\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T),$$

$$b \equiv 2\lambda_l^f\lambda_{min}(\boldsymbol{B}\boldsymbol{B}^T), c \equiv 2\lambda_{max}(\boldsymbol{Q}).$$

Now

$$\boldsymbol{P}_{u0} \equiv (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\eta^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}$$

$$\preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\eta'^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q},$$

$$\preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\eta'^{-1}\boldsymbol{I} + \lambda_{min}(\boldsymbol{R}^{-1})\Delta\boldsymbol{B}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}'_{u0},$$

$$(2.21)$$

Using Theorem 3 we have,

$$\boldsymbol{P} \preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{u0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q},$$

$$\preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}'^{-1}_{u0} + \Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}$$

Now $\boldsymbol{M}^{(q)}\boldsymbol{P}\boldsymbol{M}^{(q)^T}$ is upper bounded by $\boldsymbol{P}_u^f$ which implies

$$\boldsymbol{M}^{(q)}\boldsymbol{P}\boldsymbol{M}^{(q)^T} \preceq \boldsymbol{M}^{(q)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1}+\Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}$$
$$\times (\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(q)^T} + \boldsymbol{M}^{(q)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(q)^T} \preceq \boldsymbol{P}_u^f.$$

We have,

$$\boldsymbol{M}^{(q)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1}+\Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}$$
$$\times (\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(q)^T} + \boldsymbol{M}^{(q)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(q)^T} \preceq \boldsymbol{P}_u^f.$$
$$\boldsymbol{M}^{(q)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1}+\Delta\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}$$
$$\times (\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(q)^T} \preceq \boldsymbol{P}_u^f - \boldsymbol{M}^{(q)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(q)^T}.$$

This leads to the following LMI,

$$\begin{bmatrix} \boldsymbol{P}_u^f - \boldsymbol{M}^{(q)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(q)^T} & \boldsymbol{M}^{(q)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T \\ (\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(q)^T} & {\boldsymbol{P}'_{u0}}^{-1}+\Delta\boldsymbol{B}\boldsymbol{S}\boldsymbol{B}^T \end{bmatrix} \succeq 0,$$

Using $\Delta = 1$, replacing $\boldsymbol{A}+\boldsymbol{I}$ by $\boldsymbol{A}^T$ and $\boldsymbol{B}$ by $\boldsymbol{C}^T$ we recover the theorem. $\square$

Remark 5. An analogous result for calculating $\boldsymbol{S}$ for upper bound on steady state Kalman filter state error covariance can be recovered without using the eigen value approach.

Theorem 6. *The optimal precision of the sensors for a LTI system, which guarantees that the steady state prior covariance matrix $\boldsymbol{P} \preceq \boldsymbol{P}_u^f$ for **any initial condition** is obtained by solving the following optimization problem,*

$$S^* = \min_{S} c(S),$$

Subjected to:

$$\begin{bmatrix} T_3 & AP_u^f C^T SC \\ C^T SCP_u^f A^T & (P_u^f)^{-1} + C^T SC \end{bmatrix} \succeq 0,$$

where

$$T_3 := P_u^f - AP_u^f A^T + AP_u^f C^T SCP_u^f A^T - Q$$

and assuming uniform complete controllability regarding the process noise and uniform complete observability of the linear time varying discrete-time system.

Remark 6. Theorem 6 can be modified exactly as same as Theorem 5 to construct the result for the public variable $x^{(q)}$.

2.6.6 Choosing feasible upper bound of $P^{(q)}$ for Kalman filter

The maximum covariance on the estimated state variable x i.e. P should be less than $\mathbf{P}^{ub}$, which satisfies the following:

$$\mathbf{P}^{ub} := (AP^{ub}A^T + BQB^T)$$

The matric $\mathbf{P}^{ub}$ is calculated by using $\boldsymbol{R} = \infty$ in the DARE. The corresponding $\boldsymbol{P}^{(q)} := \boldsymbol{M}^{(q)}\mathbf{P}^{ub}\boldsymbol{M}^{(q)T}$.

2.6.7   Discussion on the cost function $c(\boldsymbol{R})$ or $c(\boldsymbol{S})$

2.7   Problem statement II

The dynamics is modeled as:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{w}_k, \ \forall k \in \mathbb{N}. \tag{2.22}$$

where $\boldsymbol{x}_k$ can be partitioned into two parts as $\boldsymbol{x}_k = \{\boldsymbol{x}_k^{(p)}, \boldsymbol{x}_k^{(q)}\}$; private and public variable respectively. The discrete dynamics is observed by a linear measurement model:

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{n}_k, \ \forall k \in \mathbb{N},$$
$$\boldsymbol{z}_k = \boldsymbol{H}^T\boldsymbol{y}_k \tag{2.23}$$

where $\boldsymbol{C}$ is the measurement matrix which is known. The measurements $\boldsymbol{y}_k$ are being compressed to $\boldsymbol{z}_k$ by $\boldsymbol{H}^T$, which is a variable before being transmitted to the end-user.

The problem that we are now interested in is: we assume that the system matrices $(\boldsymbol{A}, \boldsymbol{C})$ and noise parameters $(\boldsymbol{Q}, \boldsymbol{R})$ are known. The matrix $\boldsymbol{H}^T$ that compresses the real measurement space, is the design variable. For a prescribed upper or lower bound on the *steady state of public and private variable's prior covariance* respectively using the Kalman filter, we are to design $\boldsymbol{H}^T$, that satisfies these bounds.

The corresponding UARE-R with the masking matrix $\boldsymbol{H}^T$ is:

$$\boldsymbol{P}\boldsymbol{A} + \boldsymbol{A}^T\boldsymbol{P} + \Delta\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A} - (\Delta\boldsymbol{A}^T + \boldsymbol{I})\boldsymbol{P}\boldsymbol{B}\boldsymbol{H}$$

$$\times (\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H} + \Delta\boldsymbol{H}^T\boldsymbol{B}^T\boldsymbol{P}\boldsymbol{B}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T\boldsymbol{P}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \boldsymbol{Q} = 0. \qquad (2.24)$$

We recover the DARE for prior covariance matrix by the usual substitutions.

Theorem 1 becomes:

Theorem 7. *Let $\boldsymbol{P}$ be the positive solution of the UARE-R in equation (2.4), then*

$$\boldsymbol{P} \preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{u0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{u1} \qquad (2.25)$$

where the matrix $\boldsymbol{P}_{u0}$ is defined as,

$$\boldsymbol{P}_{u0} \equiv (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\eta^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \qquad (2.26)$$

and the positive constant η is defined as,

$$\eta \equiv f\Big(- [\lambda_1(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + \Delta\lambda_1(\boldsymbol{Q})\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})$$

$$\times \sigma_{n_y}^2(\boldsymbol{B})], 2\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{B}), 2\lambda_1(\boldsymbol{Q})\Big), \qquad (2.27)$$

where $f(a, b, c)$ is defined as,

$$f(a, b, c) \equiv \frac{-a + \sqrt{a^2 + bc}}{b}. \qquad (2.28)$$

49

Proof. We have,

$$\Delta(\boldsymbol{PA} + \boldsymbol{A}^T\boldsymbol{P} + \Delta\boldsymbol{A}^T\boldsymbol{PA})(\Delta\boldsymbol{A} + \boldsymbol{I})^T\boldsymbol{P}(\Delta\boldsymbol{A} + \boldsymbol{I}) - \boldsymbol{P}, \qquad (2.29)$$

Re-writing UARE-R as:

$$\begin{aligned}
\Delta(\boldsymbol{PA} + \boldsymbol{A}^T\boldsymbol{P} + \Delta\boldsymbol{A}^T\boldsymbol{PA}) - \Delta(\Delta\boldsymbol{A}^T + \boldsymbol{I}) \\
\times \boldsymbol{PBH}(\boldsymbol{H}^T\boldsymbol{RH} + \Delta\boldsymbol{H}^T\boldsymbol{B}^T\boldsymbol{PBH})^{-1} \\
\times \boldsymbol{H}^T\boldsymbol{B}^T\boldsymbol{P}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \boldsymbol{Q}) = 0, \qquad (2.30)
\end{aligned}$$

or,

$$\begin{aligned}
(\Delta\boldsymbol{A} + \boldsymbol{I})^T[\boldsymbol{P} - \Delta\boldsymbol{PBH}(\boldsymbol{H}^T\boldsymbol{RH} + \Delta\boldsymbol{H}^T\boldsymbol{B}^T\boldsymbol{PBH})^{-1} \\
\times \boldsymbol{H}^T\boldsymbol{B}^T\boldsymbol{P}](\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} = \boldsymbol{P}. \qquad (2.31)
\end{aligned}$$

Using Matrix Inversion lemma we get,

$$\begin{aligned}
(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}^{-1} + \Delta\boldsymbol{BH}(\boldsymbol{H}^T\boldsymbol{RH})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1} \\
(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} = \boldsymbol{P}. \qquad (2.32)
\end{aligned}$$

Following [49] we have,

$$\boldsymbol{P} \preceq \lambda_1(\boldsymbol{P})\boldsymbol{I}$$

50

and,

$$BH(H^TRH)^{-1}H^TB^T$$

$$\succeq \lambda_{n_y}((H^TRH)^{-1})\lambda_{n_y}(BHH^TB^T)I$$

$$\succeq \lambda_{n_y}((H^TRH)^{-1})\lambda_{n_y}(BB^T)\lambda_{n_y}(HH^T)I$$

$$\succeq \frac{1}{\lambda_{n_1}(H^TRH)}\lambda_{n_y}(BB^T)\lambda_{n_y}(HH^T)I$$

$$= \frac{1}{\lambda_{n_1}(H^TH)}\lambda_{n_y}(R^{-1})\lambda_{n_y}(BB^T)\lambda_{n_y}(HH^T)I$$

$$= \frac{\lambda_{n_y}(HH^T)}{\lambda_{n_1}(H^TH)}\lambda_{n_y}(R^{-1})\sigma^2_{n_y}(B)I$$

$$= \frac{\lambda_{n_y}(H^TH)}{\lambda_{n_1}(H^TH)}\lambda_{n_y}(R^{-1})\sigma^2_{n_y}(B)I$$

$$\succeq \frac{1}{\gamma^2}\lambda_{n_y}(R^{-1})\sigma^2_{n_y}(B)I$$

where condition number of H is less than or equal to γ.

Using them, we have:

$$P \preceq \frac{\lambda_1(P)}{1 + \lambda_1(P)\Delta\frac{1}{\gamma^2}\lambda_{n_y}(R^{-1})\sigma^2_{n_y}(B)}(\Delta A + I)^T$$

$$\times (\Delta A + I) + \Delta Q \tag{2.33}$$

Lemma 1 in [49] (Amir-Moez 1956) states,

$$\lambda_{i+j-1}(A_1 + B_1) \leq \lambda_j(A_1) + \lambda_i(B_1), \ i + j \leq n + 1,$$

for any symmetric matrices, $A_1, B_1 \in \mathbb{R}^{n \times n}$ and $1 \leq i, j \leq n$.

Using $i = 1$ and $j = 1$ we have,

$$\lambda_1(\boldsymbol{A}_1 + \boldsymbol{B}_1) \leq \lambda_1(\boldsymbol{A}_1) + \lambda_1(\boldsymbol{B}_1). \tag{2.34}$$

Hence,

$$\lambda_1(\boldsymbol{P}) \leq \frac{\lambda_1(\boldsymbol{P})}{1 + \lambda_1(\boldsymbol{P})\Delta\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{B})}\lambda_1[(\Delta\boldsymbol{A} + \boldsymbol{I})^T$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I})] + \Delta\lambda_1(\boldsymbol{Q})$$

Using $\lambda_1[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] = \Delta\lambda_1(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + 1$ and then rearranging

we get,

$$\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{B})\lambda_1^2(\boldsymbol{P}) - [\lambda_1(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$

$$+ \Delta\lambda_1(\boldsymbol{Q})\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{B})]\lambda_1(\boldsymbol{P}) - \lambda_1(\boldsymbol{Q}) \leq 0. \tag{2.35}$$

which is of quadratic form. Hence finally,

$$\lambda_1(\boldsymbol{P}) \leq f(-[\lambda_1(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + \Delta\lambda_1(\boldsymbol{Q})$$

$$\times \frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{B})], 2\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{B}), 2\lambda_1(\boldsymbol{Q})) \equiv \eta.$$

where,

$$f(a, b, c) \equiv \frac{-a + \sqrt{a^2 + bc}}{b}. \tag{2.36}$$

Using (2.9),

$$
\begin{aligned}
\boldsymbol{P} \preceq & (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\lambda_1^{-1}(\boldsymbol{P})\boldsymbol{I} + \\
& \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}, \\
\preceq & (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\eta^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1} \\
& \times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{u0}.
\end{aligned}
\tag{2.37}
$$

Putting it back to the modified UARE-R in equation (2.9), we get,

$$
\begin{aligned}
\boldsymbol{P} \preceq & (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\boldsymbol{P}_{u0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1} \\
& \times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{u1}
\end{aligned}
$$

$\square$

Using similar argument Theorem 3 becomes:

Theorem 8. *Let $\boldsymbol{P}$ be the positive solution of the UARE-R in equation (2.4), then*

$$
\begin{aligned}
\boldsymbol{P} \succeq & (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\boldsymbol{P}_{l0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1} \\
& \times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{l1}
\end{aligned}
\tag{2.38}
$$

where the matrix $\boldsymbol{P}_{l0}$ is defined as,

$$
\begin{aligned}
\boldsymbol{P}_{l0} \equiv & (\Delta\boldsymbol{A} + \boldsymbol{I})^T (\varphi^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1} \\
& \times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}
\end{aligned}
\tag{2.39}
$$

53

and the positive constant φ is defined as,

$$\varphi \equiv f(-[\lambda_{n_x}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta \boldsymbol{A}^T \boldsymbol{A}) + \Delta \lambda_{n_x}(\boldsymbol{Q}) \gamma^2 \lambda_1(\boldsymbol{R}^{-1})$$

$$\times \sigma_1^2(\boldsymbol{B})], 2\gamma^2 \lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B}), 2\lambda_{n_x}(\boldsymbol{Q})), \tag{2.40}$$

where $f(a, b, c)$ is defined as,

$$f(a, b, c) \equiv \frac{-a + \sqrt{a^2 + bc}}{b}. \tag{2.41}$$

Proof. Following [49] we have,

$$\boldsymbol{P}^{-1} \preceq \lambda_{n_x}^{-1}(\boldsymbol{P})\boldsymbol{I}. \tag{2.42}$$

$$\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T \boldsymbol{R} \boldsymbol{H})^{-1}\boldsymbol{H}^T \boldsymbol{B}^T$$

$$\preceq \frac{\lambda_1(\boldsymbol{H}^T \boldsymbol{H})}{\lambda_n(\boldsymbol{H}^T \boldsymbol{H})}\lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B})\boldsymbol{I}.$$

$$\preceq \gamma^2 \lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B})\boldsymbol{I}. \tag{2.43}$$

where condition number of $\boldsymbol{H}$ is less than or equal to γ. Using (2.42) and (2.43) in (2.9), we have:

$$\boldsymbol{P} \succeq \frac{\lambda_{n_x}(\boldsymbol{P})}{1 + \lambda_{n_x}(\boldsymbol{P})\Delta\gamma^2\lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B})}(\Delta \boldsymbol{A} + \boldsymbol{I})^T$$

$$\times (\Delta \boldsymbol{A} + \boldsymbol{I}) + \Delta \boldsymbol{Q} \tag{2.44}$$

Lemma 1 in [49] (Amir-Moez 1956) states,

$$\lambda_{i+j-n}(\boldsymbol{A}_1 + \boldsymbol{B}_1) \geq \lambda_j(\boldsymbol{A}_1) + \lambda_i(\boldsymbol{B}_1), \ i + j \geq n + 1,$$

for any symmetric matrices, $\boldsymbol{A}_1, \boldsymbol{B}_1 \in \mathbb{R}^{n \times n}$ and $1 \leq i, j \leq n$.

Using $i = n$ and $j = n$ we have,

$$\lambda_n(\boldsymbol{A} + \boldsymbol{B}) \geq \lambda_n(\boldsymbol{A}_1) + \lambda_n(\boldsymbol{B}_1). \tag{2.45}$$

Using equation (2.45) in equation (2.44) after applying eigen value operator on equation (2.44), we get,

$$\lambda_{n_x}(\boldsymbol{P}) \geq \frac{\lambda_{n_x}(\boldsymbol{P})}{1 + \lambda_{n_x}(\boldsymbol{P})\Delta\gamma^2\lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B})} \lambda_{n_x}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T$$
$$\times (\Delta\boldsymbol{A} + \boldsymbol{I})] + \Delta\lambda_{n_x}(\boldsymbol{Q}).$$

Using $\lambda_{n_x}[(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\Delta\boldsymbol{A} + \boldsymbol{I})] = \Delta\lambda_{n_x}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + 1$ and then rearranging we get,

$$\Delta\gamma^2\lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B})\lambda_{n_x}^2(\boldsymbol{P}) - [\Delta\lambda_{n_x}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A})$$
$$+ \Delta\lambda_{n_x}(\boldsymbol{Q})\Delta\gamma^2\lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B})]\lambda_{n_x}(\boldsymbol{P}) - \Delta\lambda_{n_x}(\boldsymbol{Q}) \geq 0, \tag{2.46}$$

which is of quadratic form. Hence finally,

$$\lambda_{n_x}(\boldsymbol{P}) \geq f(-[\lambda_{n_x}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + \Delta\lambda_{n_x}(\boldsymbol{Q})$$
$$\times \gamma^2\lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B})], 2\gamma^2\lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B}), 2\lambda_{n_x}(\boldsymbol{Q})) \equiv \varphi,$$

where,

$$f(a, b, c) \equiv \frac{-a + \sqrt{a^2 + bc}}{b}. \tag{2.47}$$

We have,

$$\lambda_n^{-1}(\boldsymbol{P}) \leq \varphi^{-1} \tag{2.48}$$

Using equation (2.48) in equation (2.44) we get,

$$\boldsymbol{P} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\varphi^{-1}\boldsymbol{I} + \boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{l0} \tag{2.49}$$

Using the lower bound $\boldsymbol{P}_{l0}$ of in equation (2.9) we get,

$$\boldsymbol{P} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}^{-1} + \boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \tag{2.50}$$

$\square$

2.7.1 Calculate $\boldsymbol{H}^T$ for lower bound on steady state of private state $\boldsymbol{x}^{(p)}$ estimates

Theorem 9. *The convex feasible set of compressive mapping $\boldsymbol{H}^T$ for a condition number less or equal to given γ, that satisfies $\boldsymbol{P}^{(p)} \succeq \boldsymbol{P}_l^f$, on the prior covariance of estimates of the private states for Kalman filtering on system described in (2.22) and (2.23), is given by the following four matrix inequalities:*

$$\begin{bmatrix} \boldsymbol{X}_1 - \boldsymbol{P'}_{l0}^{-1} & \boldsymbol{C}^T \boldsymbol{H} \\ \boldsymbol{H}^T \boldsymbol{C} & \boldsymbol{X}_2 \end{bmatrix} \succeq 0,$$

$$\begin{bmatrix} \boldsymbol{M}^{(p)} \boldsymbol{A} \boldsymbol{X}_3 \boldsymbol{A}^T \boldsymbol{M}^{(p)T} & \boldsymbol{I} \\ \boldsymbol{I} & (\boldsymbol{P}_l^f)^{-1} \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} \boldsymbol{X}_2 & \boldsymbol{H}^T \\ \boldsymbol{H} & \boldsymbol{R}^{-1} \end{bmatrix} \succeq 0, \quad \begin{bmatrix} \boldsymbol{X}_3 & \boldsymbol{I} \\ \boldsymbol{I} & \boldsymbol{X}_1 \end{bmatrix} \succeq 0,$$

where,

$$\boldsymbol{P'}_{l0} \equiv \boldsymbol{A}(\varphi^{-1} \boldsymbol{I} + \gamma^2 \boldsymbol{C}^T \boldsymbol{R}^{-1} \boldsymbol{C})^{-1} \boldsymbol{A}^T + \boldsymbol{Q} \tag{2.51}$$

$$\begin{aligned} \varphi \equiv f(&-[\lambda_{n_x}(\boldsymbol{A}\boldsymbol{A}^T - \boldsymbol{I}) + \lambda_{n_x}(\boldsymbol{Q})\gamma^2 \lambda_1(\boldsymbol{R}^{-1}) \\ &\times \sigma_1^2(\boldsymbol{C}^T)], 2\gamma^2 \lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{C}^T), 2\lambda_{n_x}(\boldsymbol{Q})), \end{aligned} \tag{2.52}$$

where $f(a, b, c)$ is defined as,

$$f(a, b, c) \equiv \frac{-a + \sqrt{a^2 + bc}}{b}. \tag{2.53}$$

Proof. We first take a look at Theorem 8, the lower bound theorem. The variable φ is defined as,

$$\begin{aligned} \varphi \equiv f(&-[\lambda_{n_x}(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + \Delta\lambda_{n_x}(\boldsymbol{Q})\gamma^2 \lambda_1(\boldsymbol{R}^{-1}) \\ &\times \sigma_1^2(\boldsymbol{B})], 2\gamma^2 \lambda_1(\boldsymbol{R}^{-1})\sigma_1^2(\boldsymbol{B}), 2\lambda_{n_x}(\boldsymbol{Q})) \end{aligned}$$

We notice that φ is a function of $\lambda_1(\boldsymbol{R}^{-1})$. In this case $\boldsymbol{R}$ is known or $\lambda_1(\boldsymbol{R}^{-1})$ is known. Hence we have,

$$\boldsymbol{P}_{l0} \equiv (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\varphi^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times\, (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}$$
$$\succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\varphi^{-1}\boldsymbol{I} + \Delta\gamma^2\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T)^{-1}$$
$$\times\, (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}'_{l0},$$

Using Theorem 8 we have,

$$\boldsymbol{P} \succeq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times\, (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}.$$

Let's assume $\boldsymbol{M}^{(p)}$ is the privacy masking matrix :

$$\boldsymbol{M}^{(p)}\boldsymbol{P}\boldsymbol{M}^{(p)T}$$
$$\succeq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{l0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times\, (\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)T} + \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)T}$$
$$\succeq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}'^{-1}_{l0} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times\, (\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)T} + \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)T} \succeq \boldsymbol{P}_l^f.$$
$$\boldsymbol{P}^{(p)}$$
$$\succeq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}'^{-1}_{l0} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times\, (\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)T} + \boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)T} \succeq \boldsymbol{P}_l^f.$$

Or,

$$\boldsymbol{M}^{(p)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T(\boldsymbol{P}_{l0}'^{-1}+\Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times(\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(p)T}\succeq\boldsymbol{P}_l^f-\boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)T}$$

Let's assume there exists $\boldsymbol{X}_1$ such that,

$$\boldsymbol{M}^{(p)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T(\boldsymbol{P}_{l0}'^{-1}+\Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$
$$\times(\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(p)T}$$
$$\succeq\boldsymbol{M}^{(p)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T\boldsymbol{X}_1^{-1}(\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(p)T}$$
$$\succeq\boldsymbol{P}_l^f-\boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)T}$$

which denotes,

$$\boldsymbol{X}_1\succeq\boldsymbol{P}_{l0}'^{-1}+\Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T \qquad (2.54)$$

and

$$\boldsymbol{M}^{(p)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T\boldsymbol{X}_1^{-1}(\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(p)T}$$
$$\succeq\boldsymbol{P}_l^f-\boldsymbol{M}^{(p)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(p)T} \qquad (2.55)$$

In (2.54) we introduce a new variable $\boldsymbol{X}_2$ such that:

$$\boldsymbol{X}_1\succeq\boldsymbol{P}_{l0}'^{-1}+\Delta\boldsymbol{B}\boldsymbol{H}\boldsymbol{X}_2^{-1}\boldsymbol{H}^T\boldsymbol{B}^T$$
$$\succeq\boldsymbol{P}_{l0}'^{-1}+\Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T$$

which gives us,

$$\begin{bmatrix} \boldsymbol{X}_1 - \boldsymbol{P}'_{l0}{}^{-1} & \sqrt{\Delta}\boldsymbol{B}\boldsymbol{H} \\ \sqrt{\Delta}\boldsymbol{H}^T\boldsymbol{B}^T & \boldsymbol{X}_2 \end{bmatrix} \succeq 0 \tag{2.56}$$

and

$$\boldsymbol{X}_2 \preceq \boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H} \tag{2.57}$$

From (2.55) we get,

$$\begin{bmatrix} \boldsymbol{X}_2 & \boldsymbol{H}^T \\ \boldsymbol{H} & \boldsymbol{R}^{-1} \end{bmatrix} \succeq 0 \tag{2.58}$$

We need to satisfy,

$$\boldsymbol{P}_l^f \preceq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T\boldsymbol{X}_1^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T} \tag{2.59}$$

We introduce another variable $\boldsymbol{X}_3$ such that,

$$\boldsymbol{P}_l^f \preceq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T\boldsymbol{X}_1^{-1}(\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T}$$
$$\preceq \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A} + \boldsymbol{I})^T\boldsymbol{X}_3(\Delta\boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(p)^T}$$

which gives us,

$$\begin{bmatrix} \boldsymbol{M}^{(p)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T\boldsymbol{X}_3(\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(p)T} & \boldsymbol{I} \\ \boldsymbol{I} & (\boldsymbol{P}_l^f)^{-1} \end{bmatrix} \succeq 0 \qquad (2.60)$$

and

$$\boldsymbol{X}_3 \succeq \boldsymbol{X}_1^{-1}$$

which leads us to,

$$\begin{bmatrix} \boldsymbol{X}_3 & \boldsymbol{I} \\ \boldsymbol{I} & \boldsymbol{X}_1 \end{bmatrix} \succeq 0 \qquad (2.61)$$

Using $\Delta = 1$, replacing $\boldsymbol{A}+\boldsymbol{I}$ by $\boldsymbol{A}^T$ and $\boldsymbol{B}$ by $\boldsymbol{C}^T$ in LMIs (**??**),(2.56),and (2.58) we get the theorem. $\qquad\square$

Remark 7. For a fixed γ, the preceding LMIs (**??**),(2.56), and (2.58) give a feasible set of $\boldsymbol{H}$. For a given problem the dimension of $\boldsymbol{H}^T$ needs to be fixed a priori. If $\boldsymbol{H}^T$ is required to be square, non-singular matrix with a condition number which is less than γ, the $\boldsymbol{P}^{(p)} \succeq \boldsymbol{P}_l^f$ is either satisfied or non satisfied irrespective of a particular choice of the $\boldsymbol{H}^T$ matrix. This is because the covariance update equation is independent of the choice of $\boldsymbol{H}^T$ when it is square non-singular. If $\boldsymbol{H}^T$ is required to be full rank matrix with row-size less than column-size (compression of real measurement space), the LMIs (**??**),(2.56), and (2.58) gives the feasible set of all $\boldsymbol{H}^T$ that has their condition number less than fixed γ. If $\boldsymbol{H}^T$ is square but non-singular, or $\boldsymbol{H}^T$ has more rows than columns. We can add an iterative algorithm if the condition number needs to be minimized. Minimizing condition number improves the numerical stability of the $\boldsymbol{H}^T$. A corollary can be established for the

case when $\boldsymbol{M}^{(p)} := \boldsymbol{I}$ akin to that of Corollary ??.

2.7.2 Calculate $\boldsymbol{H}^T$ for upper bound on steady state Kalman filter state error covariance of public state $\boldsymbol{x}_k^{(q)}$

Theorem 10. *The convex feasible set of compressive mapping $\boldsymbol{H}^T$ for a given condition number, that satisfies $\boldsymbol{P}^{(q)} \preceq \boldsymbol{P}_u^f$, on the prior covariance of estimates of the public states for Kalman filtering on system described in (2.22) and (2.23), is given by the following equations:*

$$\begin{bmatrix} \boldsymbol{P}_u^f - \boldsymbol{M}^{(q)}\boldsymbol{Q}\boldsymbol{M}^{(q)T} & \boldsymbol{M}^{(q)}\boldsymbol{A} \\ \boldsymbol{A}^T\boldsymbol{M}^{(q)T} & \boldsymbol{P'}_{u0}^{-1} + \boldsymbol{X}_1 \end{bmatrix} \succeq 0, \tag{2.62}$$

$$\boldsymbol{X}_1 \preceq \boldsymbol{C}^T\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{C}, \tag{2.63}$$

$$\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}, \quad \boldsymbol{U} \in \mathbb{R}^{n_y \times n_z}_{n_z \leq n_y}. \tag{2.64}$$

where

$$\boldsymbol{A}(\eta^{-1}\boldsymbol{I} + \frac{1}{\gamma^2}\boldsymbol{C}^T\boldsymbol{R}^{-1}\boldsymbol{C})^{-1}\boldsymbol{A}^T$$

$$+ \boldsymbol{Q} \equiv \boldsymbol{P'}_{u0} \tag{2.65}$$

where

$$\eta \equiv f(-[\lambda_1(\boldsymbol{A}\boldsymbol{A}^T - \boldsymbol{I}) + \lambda_1(\boldsymbol{Q})\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})$$

$$\times \sigma_{n_y}^2(\boldsymbol{C}^T)], 2\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{C}^T), 2\lambda_1(\boldsymbol{Q})$$

where condition number of $\boldsymbol{H}$ is less than or equal to γ and $f(a,b,c)$ is defined as,

$$f(a,b,c) \equiv \frac{-a + \sqrt{a^2 + bc}}{b}. \tag{2.66}$$

and $\boldsymbol{H} := \boldsymbol{R}^{-1/2}\bar{\boldsymbol{H}}$ and the "economy size" SVD of $\bar{\boldsymbol{H}}$ is $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^T$

Proof. In Theorem 7, the upper bound theorem, the variable η is defined as,

$$\eta \equiv f(-[\lambda_1(\boldsymbol{A} + \boldsymbol{A}^T + \Delta\boldsymbol{A}^T\boldsymbol{A}) + \Delta\lambda_1(\boldsymbol{Q})\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})$$

$$\times \sigma_{n_y}^2(\boldsymbol{B})], 2\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{B}), 2\lambda_1(\boldsymbol{Q})$$

Now,

$$\boldsymbol{P}_{u0} \equiv (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\eta^{-1}\boldsymbol{I} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q},$$

$$\preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\eta^{-1}\boldsymbol{I} + \frac{1}{\gamma^2}\Delta\boldsymbol{C}^T\boldsymbol{R}^{-1}\boldsymbol{C})^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}'_{u0},$$

Using Theorem 7 we have,

$$\boldsymbol{P} \preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}_{u0}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q} \equiv \boldsymbol{P}_{u1},$$

$$\preceq (\Delta\boldsymbol{A} + \boldsymbol{I})^T(\boldsymbol{P}'^{-1}_{u0} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A} + \boldsymbol{I}) + \Delta\boldsymbol{Q}$$

Now we want to upper bound $\boldsymbol{M}^{(q)}\boldsymbol{P}\boldsymbol{M}^{(q)T}$ by $\boldsymbol{P}_u^f$. That is ensured if we have,

$$\boldsymbol{M}^{(q)}\boldsymbol{P}\boldsymbol{M}^{(q)T} \preceq \boldsymbol{M}^{(q)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}$$

$$\times (\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}(\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(q)T}$$

$$+ \boldsymbol{M}^{(q)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(q)T} \preceq \boldsymbol{P}_u^f.$$

We have,

$$\boldsymbol{M}^{(q)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(q)T} + \boldsymbol{M}^{(q)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(q)T} \preceq \boldsymbol{P}_u^f.$$

$$\boldsymbol{M}^{(q)}(\Delta\boldsymbol{A}+\boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1} + \Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T)^{-1}$$

$$\times (\Delta\boldsymbol{A}+\boldsymbol{I})\boldsymbol{M}^{(q)T} \preceq \boldsymbol{P}_u^f - \boldsymbol{M}^{(q)}\Delta\boldsymbol{Q}\boldsymbol{M}^{(q)T}. \tag{2.67}$$

Lets assume that $\boldsymbol{H} := \boldsymbol{R}^{-1/2}\bar{\boldsymbol{H}}$ and the SVD of $\bar{\boldsymbol{H}}$ is $\boldsymbol{U}\Lambda\boldsymbol{V}^T$. We have,

$$\Delta\boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T$$

$$= \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\bar{\boldsymbol{H}}(\bar{\boldsymbol{H}}^T\bar{\boldsymbol{H}})^{-1}\bar{\boldsymbol{H}}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T$$

$$= \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\bar{\boldsymbol{H}}(\boldsymbol{V}\Lambda^T\boldsymbol{U}^T\boldsymbol{U}\Lambda\boldsymbol{V}^T)^{-1}\bar{\boldsymbol{H}}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T$$

$$= \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\bar{\boldsymbol{H}}(\boldsymbol{V}\bar{\Lambda}^2\boldsymbol{V}^T)^{-1}\bar{\boldsymbol{H}}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T$$

$$= \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\Lambda\boldsymbol{V}^T(\boldsymbol{V}\bar{\Lambda}^2\boldsymbol{V}^T)^{-1}\boldsymbol{V}\Lambda^T\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T$$

$$= \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\Lambda\boldsymbol{V}^T(\boldsymbol{V}(\bar{\Lambda}^2)^{-1}\boldsymbol{V}^T)\boldsymbol{V}\Lambda^T\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T$$

$$= \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\Lambda(\bar{\Lambda}^2)^{-1}\Lambda^T\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T$$

$$\tag{2.68}$$

For a compressive mapping $\boldsymbol{H}^T$ and taking "economy size" SVD this becomes,

$$\Delta \boldsymbol{B}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{B}^T = \Delta \boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T$$

where $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$ From (2.67) we have,

$$\boldsymbol{M}^{(q)}(\Delta \boldsymbol{A} + \boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1} + \Delta \boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T)^{-1}$$
$$\times (\Delta \boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(q)^T} \preceq \boldsymbol{P}^f_u - \boldsymbol{M}^{(q)}\Delta \boldsymbol{Q}\boldsymbol{M}^{(q)^T}. \qquad (2.69)$$

where $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$ We introduce a variable $\boldsymbol{X}_1$ such that,

$$\boldsymbol{M}^{(q)}(\Delta \boldsymbol{A} + \boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1} + \Delta \boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T)^{-1}$$
$$\times (\Delta \boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(q)^T} \preceq$$
$$\boldsymbol{M}^{(q)}(\Delta \boldsymbol{A} + \boldsymbol{I})^T({\boldsymbol{P}'_{u0}}^{-1} + \boldsymbol{X}_1)^{-1}$$
$$\times (\Delta \boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(q)^T}$$
$$\preceq \boldsymbol{P}^f_u - \boldsymbol{M}^{(q)}\Delta \boldsymbol{Q}\boldsymbol{M}^{(q)^T}. \qquad (2.70)$$

The feasible set of $\boldsymbol{U}$ is thus the solution to the following three equations,

$$\begin{bmatrix} \boldsymbol{P}^f_u - \boldsymbol{M}^{(q)}\Delta \boldsymbol{Q}\boldsymbol{M}^{(q)^T} & \boldsymbol{M}^{(q)}(\Delta \boldsymbol{A} + \boldsymbol{I})^T \\ (\Delta \boldsymbol{A} + \boldsymbol{I})\boldsymbol{M}^{(q)^T} & {\boldsymbol{P}'_{u0}}^{-1} + \boldsymbol{X}_1 \end{bmatrix} \succeq 0, \qquad (2.71)$$

$$\boldsymbol{X}_1 \preceq \Delta \boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T, \qquad (2.72)$$

$$\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}, \quad \boldsymbol{U} \in \mathbb{R}^{n_y \times n_z}_{n_z \leq n_y}. \qquad (2.73)$$

The equality $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$ is an orthonormal constraint. Equation (2.72) and (2.73) combined is the most general case of the quadratic feasibility problem. Using $\Delta = 1$,

replacing $\boldsymbol{A} + \boldsymbol{I}$ by $\boldsymbol{A}^T$ and $\boldsymbol{B}$ by $\boldsymbol{C}^T$ we recover the theorem. $\quad\square$

2.7.2.1   Algorithm to solve for feasible $\boldsymbol{H}^T$

Algorithm 2

We look at (2.72) and inflate $\boldsymbol{X}_1$ to $\delta\boldsymbol{X}_1$ where $\delta >> 1$ to construct a convex cost function:

$$F(\boldsymbol{U}) = (\delta\boldsymbol{X}_1 - \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T)^2 \qquad (2.74)$$

This cost function along with the Stiefel manifold acts as a surrogate problem to solving a feasible set for (2.72) along with the Stiefel manifold constraint. The idea is to keep reducing the δ parameter to 1, while making sure that the resulting $\boldsymbol{U}$ satisfies (2.72).

The outline of our approach is as follows:

1. Solve (2.71) to generate one solution of $\boldsymbol{X}_1$.

2. Initialize $\delta >> 1$

3. Minimize the cost function

$$F(\boldsymbol{U}) = (\delta\boldsymbol{X}_1 - \Delta\boldsymbol{B}\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{B}^T)^2 \qquad (2.75)$$

   on the Stiefel manifold, $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$ using [8].

4. In each iteration reduce δ towards 1.

5. Stop if (2.72) is not feasible for new δ.

6. Choose the previous $\boldsymbol{U}$ as your solution.

After we have a feasible $\boldsymbol{U}$ satisfying all the constraints, we generate $\boldsymbol{H}^T$, where $\boldsymbol{H} := \boldsymbol{R}^{-1/2}\bar{\boldsymbol{H}}$ and the SVD of $\bar{\boldsymbol{H}}$ is $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^T$. The choice of $\boldsymbol{\Lambda}$ is not completely arbitrary. We need to ensure that the condition number on $\boldsymbol{H}$ which is assumed to be less than γ is satisfied. The inflation parameter δ is reduced exponentially to 1. The reduction methodology will effect the convergence rate for our proposed algorithm to search for a feasible $\boldsymbol{U}$.

We first transform the cost function using $\Delta = 1$, replacing $\boldsymbol{A} + \boldsymbol{I}$ by $\boldsymbol{A}^T$ and $\boldsymbol{B}$ by $\boldsymbol{C}^T$ to:

$$F(\boldsymbol{U}) = (\delta \boldsymbol{X}_1 - \boldsymbol{C}^T \boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{C})^2 \tag{2.76}$$

where $\boldsymbol{X}_1$ is generated by solving,

$$\begin{bmatrix} \boldsymbol{P}_u^f - \boldsymbol{M}^{(q)}\boldsymbol{Q}\boldsymbol{M}^{(q)T} & \boldsymbol{M}^{(q)}\boldsymbol{A} \\ \boldsymbol{A}^T\boldsymbol{M}^{(q)T} & {\boldsymbol{P}'_{u0}}^{-1} + \boldsymbol{X}_1 \end{bmatrix} \succeq 0 \tag{2.77}$$

where

$$\boldsymbol{A}(\eta^{-1}\boldsymbol{I} + \frac{1}{\gamma^2}\boldsymbol{C}^T\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{C})^{-1}\boldsymbol{A}^T$$

$$+ \boldsymbol{Q} \equiv \boldsymbol{P}'_{u0} \tag{2.78}$$

67

where

$$\eta \equiv f(-[\lambda_1(\boldsymbol{A}\boldsymbol{A}^T - \boldsymbol{I}) + \lambda_1(\boldsymbol{Q})\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})$$

$$\times \sigma_{n_y}^2(\boldsymbol{C}^T)], 2\frac{1}{\gamma^2}\lambda_{n_y}(\boldsymbol{R}^{-1})\sigma_{n_y}^2(\boldsymbol{C}^T), 2\lambda_1(\boldsymbol{Q})$$

where condition number of $\boldsymbol{H}$ is less than or equal to γ.

2.7.2.2   Newton's method to minimize cost function $\boldsymbol{F}(\boldsymbol{U})$ on the Stiefel manifold,

$$\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$$

- Given $\boldsymbol{U}$ such that $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$

  – Computing $\boldsymbol{G} = \boldsymbol{F_U} - \boldsymbol{U}\boldsymbol{F_U}^T\boldsymbol{U}$, where

$$\boldsymbol{F_U} = -2(\delta\boldsymbol{X}_1 - \boldsymbol{C}^T\boldsymbol{R}^{-1/2}\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{R}^{-1/2}\boldsymbol{C})$$

$$\times \boldsymbol{C}^T\boldsymbol{R}^{-1/2}\boldsymbol{U} \tag{2.79}$$

  – Computing $\boldsymbol{\Delta} = -\text{Hess}^{-1}\boldsymbol{G}$ such that $\boldsymbol{U}^T\boldsymbol{\Delta}$ is skew symmetric and

$$\boldsymbol{F_{UU}}(\boldsymbol{\Delta}) - \boldsymbol{U}\text{skew}(\boldsymbol{F_U}^T\boldsymbol{\Delta}) - \text{skew}(\boldsymbol{\Delta}\boldsymbol{F_U}^T)\boldsymbol{U}$$

$$- \frac{1}{2}\boldsymbol{\Pi}\boldsymbol{\Delta}\boldsymbol{U}^T\boldsymbol{F_U} = \boldsymbol{G} \tag{2.80}$$

  where $\text{skew}(\boldsymbol{X}) = (\boldsymbol{X} - \boldsymbol{X}^T)/2$ and $\boldsymbol{\Pi} = \boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^T$

- Moving from $\boldsymbol{U}$ in the direction of $\boldsymbol{\Delta}$ to $\boldsymbol{U}(1)$ using the geodesic formula

$$\boldsymbol{U}(t) = \boldsymbol{U}\boldsymbol{M}(t) + \boldsymbol{Q}_1\boldsymbol{N}(t)$$

where $\boldsymbol{Q_1}\boldsymbol{R_1}$ is the compact QR-decomposition of $(\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^T)\boldsymbol{\Delta}$. We assume $\boldsymbol{A}_1 = \boldsymbol{U}^T\boldsymbol{\Delta}$, and $\boldsymbol{M}(t)$ and $\boldsymbol{N}(t)$ are given by the matrix exponential

$$\begin{bmatrix} \boldsymbol{M}(t) \\ \boldsymbol{N}(t) \end{bmatrix} = \mathrm{exp}t \begin{bmatrix} \boldsymbol{A}_1 & -\boldsymbol{R}_1^T \\ \boldsymbol{R}_1 & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{0} \end{bmatrix} \tag{2.81}$$

- Repeat until the absolute value of the difference is cost function between two consecutive steps is within a specified bound.

2.8 Numerical Example

The system considered here is a n_x dimensional discrete time linear Gaussian system. The $\boldsymbol{B}$ matrices are chosen to be identity. The $\boldsymbol{Q}$ matrix is $\boldsymbol{I}$. The $\boldsymbol{A}$ and $\boldsymbol{C}$ matrices are chosen such that $[\boldsymbol{A}, \boldsymbol{C}]$ pair is detectable and $[\boldsymbol{A}, \boldsymbol{B}\boldsymbol{Q}^{1/2}]$ pair is stabilizable. We choose $\boldsymbol{R}$ to be a diagonal matrix. Hence, the spectrum of $\boldsymbol{R}$, i.e. $\{\lambda_i\}$ are its diagonal elements. We choose Theorem 4 and show results for *minimizing l_1 norm on* $\boldsymbol{\lambda}$ ($\boldsymbol{R} := \mathrm{diag}(\boldsymbol{\lambda})$), for a prescribed lower bound on $\boldsymbol{P}$, where $n_x = 10$ and $n_y = 10$. The matrix $\boldsymbol{C}$ in this example is chosen to be $2\boldsymbol{I}$. The matrices $\mathbf{P}^{lb}$ and $\mathbf{P}^{ub}$ are first calculated. The eigen values of

$$\mathrm{eig}(\mathbf{P}^{ub}) = [1.000\ 1.001\ 1.012\ 1.123\ 1.186\ 2.139\ 3.172\ 4.705\ 9.096\ 279.143],$$

while the eigenvalues of $\mathbf{P}^{lb}$ all are equal to 1.

We then select the prescribed lower bound $\boldsymbol{P}_l^f$ to be $(1/16) \times (\mathbf{P}^{ub} + 15\mathbf{P}^{lb})$. This convex combination ensures a smooth transition from $\mathbf{P}^{lb}$ to $\mathbf{P}^{ub}$ when $\boldsymbol{R}$ goes from $\boldsymbol{0}$ to ∞. We calculate $\varphi' = 1.0000193$ and $\boldsymbol{P}'_{u0}$. We select the upper bound λ_u^f to be

0.03.

The eigen values of $\boldsymbol{P}'_{l0}$:

$$\mathrm{eig}(\boldsymbol{P}'_{l0}) = [28.689\ 2.601\ 2.028\ 1.599\ 1.480\ 1.103\ 1.078\ 1.006\ 1.000\ 1.000].$$

We solve the optimization problems using CVX in Matlab. The minimum l_1 norm cost is 18336.433 . On a 2GHz Intel Core i5 machine, the l_1 problem takes 1.20 seconds.
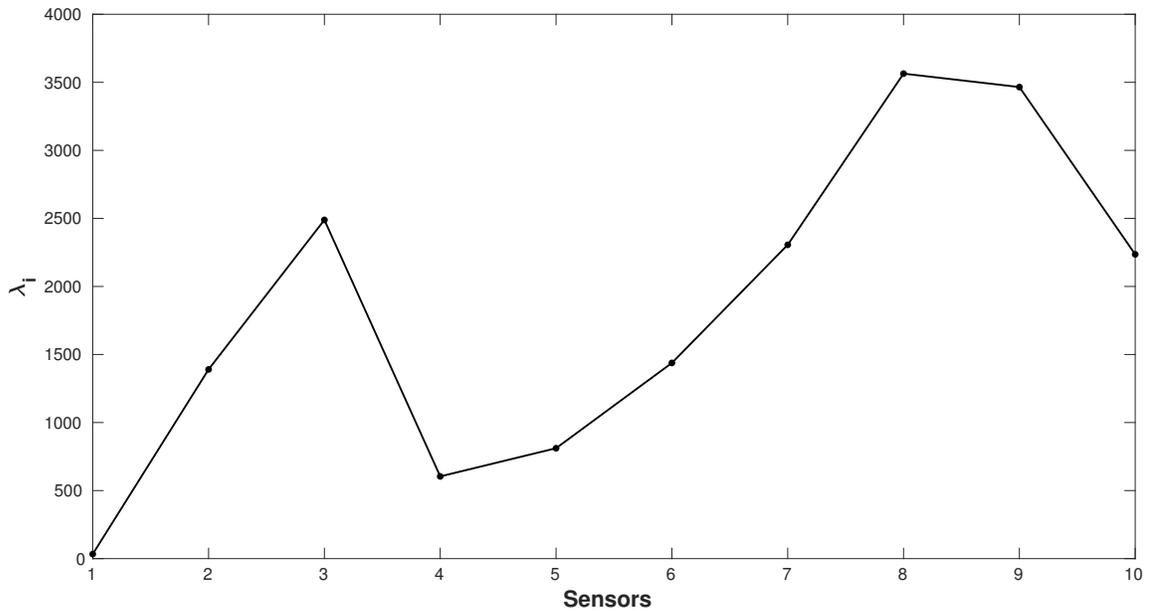
Figure 2.1: Plot of sensor covariance values for 10 sensors for prescribed lower bound on $\boldsymbol{P}$. Circle denotes covariance values calculated from minimization of l_1 norm of the vector $\boldsymbol{\lambda}$. Reprinted with permission from [18].

The $\boldsymbol{\lambda}$ vector of sensors for l_1 norm shown in fig(2.1) are:

$$\boldsymbol{\lambda} = [33.333 \ 1390.292 \ 2488.968 \ 604.108 \ 811.228$$

$$1437.797 \ 2305.879 \ 3563.793 \ 3465.244 \ 2235.787]$$

The $\boldsymbol{R} := \mathrm{diag}(\boldsymbol{\lambda})$ noise covariance that we calculated is the minimum noise that needs to be in the measurements to ensure that the steady state error covariance matrix is greater than the prescribed lower bound $\boldsymbol{P}_l^f$. This is verified by calculating the eigen values of the $\boldsymbol{P} - \boldsymbol{P}_l^f$ matrix, which turns out to be all positive. Matrix $\boldsymbol{P}$ is the DARE solution for the optimal $\boldsymbol{R}$. We notice that there is a large gap between the lower bound and the final steady state value of $\boldsymbol{P}$. This is due to the fact that we used eigen value approximations in deriving the result. An ad-hoc method to reduce this gap is to iteratively reduce the magnitude of the $\boldsymbol{\lambda}$ till the eigenvalues of $\boldsymbol{P} - \boldsymbol{P}_l^f$ remain all positive. We found out that we can reduce the $\boldsymbol{\lambda}$ by a factor of 0.08 and still ensure $\boldsymbol{P} \succeq \boldsymbol{P}_l^f$.

The calculated $\boldsymbol{R}$ can be assumed to be comprised of actual measurement noise due to the system $\boldsymbol{R}_a$ and synthetic noise $\boldsymbol{R}_s$. Since in most practical cases $\boldsymbol{R}_a$ is known, our algorithm effectively calculates the minimum synthetic noise $\boldsymbol{R}_s$ that needs to be added to the actual measurement to ensure privacy with respect to state estimation.

2.9   Conclusions

In this chapter we formulate an algorithms to calculate the measurement noise covariance and measurement compression matrix which ensures that the steady state error covariance of the state estimates are lower-bounded by a prescribed bound.

3.   OPTIMAL SENSOR PRECISION FOR STATE ESTIMATION OF LINEAR TIME-VARYING DISCRETE-TIME SYSTEMS WITH BOUNDED ERRORS

3.1   Introduction

Given a set of sensors with known noise variance, Kalman filtering provides the state estimate with minimum error variance. In this paper, we look at the inverse problem: what is the sensor noise for which the error variance is less than a given upper bound? It is possible that for a set of sensors, there are several solutions to this problem. Therefore, there is scope for optimization. In this paper, we seek to determine the noisiest sensors, for which the upper bound on the error is satisfied. We show in the paper that the optimization is convex with respect to the inverse of the noise variance, which we refer to as the sensor precision. Therefore, the optimization minimizes the sensor precisions, for which the estimation error is below the prescribed bound.

This problem is of significant engineering value. For example, in emerging sensor networks, designers are challenged by the tradeoff between cost and utility. Since sensor cost is directly proportional to the sensor precision, the inclusion of unnecessarily precise sensors in the design is undesirable. However, for such applications, it is not trivial to determine the precision of the sensors, for which the utility is achieved. Or, if there is a future requirement to improve the accuracy, determining which sensors need to be improved is another non-trivial task. These problems are hard for large-scale systems with complex interactions. The proposed framework can address both these problems in a convex optimization framework.

The proposed framework also impacts other important problems in sensing, such as

sensor scheduling and selection. Existing algorithms for sensor scheduling [34, 35, 90, 39, 70, 65, 42, 75, 37, 12] and selection [44, 21, 14, 62, 78, 89] assume that the sensors noise variance is given. The framework in this paper can be used to optimally determine them.

While the focus of this paper is on determining the optimal sensor precisions, the presented framework can also be applied to schedule and select sensors. Starting with a dictionary of sensors with unknown precisions, minimizing the l_1 norm of the sensor precisions will promote sparseness in the solution. Sensors with zero precisions can be eliminated from the system, and sensors with non zero precisions will guarantee the required estimation accuracy. Therefore, using the proposed framework, it is possible to simultaneously determine the optimal sensor precisions and prune out unnecessary sensors. This is appealing because the problem can be solved in a convex optimization framework in a very general setting. This is in contrast with the NP-hard formulations and heuristic methods to solve them.

The problem of determining optimal sensor precision was first introduced in [50], in the context of output-feedback controller design for continuous-time systems without uncertainty, and with steady-state performance guarantees. Recently, we extended that work to determine optimal sensing precisions for continuous-time robust output-feedback control, with guaranteed $\mathcal{H}_2$ performance [69].

3.1.1   Key Contribution

In this paper, we look at the problem of determining the optimal sensor precision for state estimation of linear time-varying, discrete-time, stochastic systems, with multi-rate sensing. It is assumed that the measurements are periodic over m time steps,

73

and the state estimation is performed at the mk time resolution, where k represents the base clock of the discrete-time system. The objective is to determine the set of least-precise sensors, for which the errors in the state estimates are upper bounded by a prescribed value.

The main results are presented as four theorems, describing convex optimization problems, the solution of which achieves the aforementioned objective. The first two theorems address the problem of determining optimal sensor precisions to bound estimation error over time mk to $m(k+1)$. The remaining two theorems determine the optimal sensor precision to bound the steady-state error for m-periodic time-varying systems.

To the best of our knowledge, this is the first paper that determines optimal sensor precision for state-estimation of linear time-varying discrete-time systems, with multi-rate sensing.

3.1.2   Layout

Section 3.2 presents the preliminaries for this paper, where we define the model of the dynamical system and the sensor model assumed in the paper. We also introduce the inverse problem to determine the optimal precisions for a given upper bound in the estimation error. Section 3.3 presents the first two theorems for determining the sensing precision if the objective is to bound the error after one-time step. The results here are formulated in a batch processing framework, where the prior at time x_{km} and sensors at times $t_{km+1}, \cdots, t_{(k+1)m}$ are given, and the objective is to determine the sensor precisions such that the error variance of the posterior at $t_{(k+1)m}$ is below a prescribed value. In section 3.4, we present the next two theorems for determining the

sensor precisions to bound the steady-state estimation error. Section3.5, highlights the engineering relevance of the proposed framework, where the theoretical results are applied to state-estimation problems from flight mechanics and astrodynamics.

3.2   Preliminaries

We focus on determining the optimal sensor precision for linear time-varying discrete-time stochastic systems described by the model of the form:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_k \boldsymbol{x}_k + \boldsymbol{B}_k \boldsymbol{w}_k, \tag{3.1a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}_k \boldsymbol{x}_k + \boldsymbol{n}_k, \tag{3.1b}$$

where $k = 0, 1, 2, ...$ are the time indices, $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ is the n_x dimensional state of the *model* at time instant k, $\boldsymbol{w}_k \in \mathbb{R}^{n_w}$ is the n_w dimensional zero-mean Gaussian additive process noise variable with $\mathbb{E}[\boldsymbol{w}_k \boldsymbol{w}_l^T] = \boldsymbol{Q}_k$ where $\mathbb{E}[.]$ denotes the expected value. The n_{y_k} dimensional observations at time k is denoted by $\boldsymbol{y}_k \in \mathbb{R}^{n_{y_k}}$, which is corrupted by an n_y dimensional additive observation noise $\boldsymbol{n}_k \in \mathbb{R}^{n_{y_k}}$ at time instant k. The sensor noise at each time instant is a zero mean Gaussian random variable with $\mathbb{E}[\boldsymbol{n}_k \boldsymbol{n}_l^T] = \boldsymbol{R}_k$. The initial conditions are $\mathbb{E}[\boldsymbol{x}_0] = \boldsymbol{\mu}_0$ and $\mathbb{E}\left[(\boldsymbol{x}_0 - \boldsymbol{\mu}_0)(\boldsymbol{x}_0 - \boldsymbol{\mu}_0)^T\right] = \boldsymbol{\Sigma}_0$. The process noise $\boldsymbol{w}_k$, observation noise $\boldsymbol{n}_k$, and initial state variable $\boldsymbol{x}_0$ are assumed to be independent.

The optimal state estimator for the stochastic system in (3.1) is the Kalman filter,

defined by

$$\boldsymbol{K}_k = \boldsymbol{\Sigma}_k^- \boldsymbol{C}_k^T \left[\boldsymbol{C}_k \boldsymbol{\Sigma}_k^- \boldsymbol{C}_k^T + \mathbf{R}_k\right]^{-1}, \qquad \text{(Kalman Gain)}$$

$$\boldsymbol{\mu}_k^- = \boldsymbol{A}_k \boldsymbol{\mu}_{k-1}^+, \qquad \text{(Mean Propagation)}$$

$$\boldsymbol{\Sigma}_k^- = \boldsymbol{A}_k \boldsymbol{\Sigma}_{k-1}^+ \boldsymbol{A}_k^T + \boldsymbol{B}_k \boldsymbol{Q}_k \boldsymbol{B}_k^T, \qquad \text{(Variance Propagation)}$$

$$\boldsymbol{\mu}_k^+ = \boldsymbol{\mu}_k^- + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}_k \boldsymbol{\mu}_k^-), \qquad \text{(Mean Update)}$$

$$\boldsymbol{\Sigma}_k^+ = (\boldsymbol{I}_{n_x} - \boldsymbol{K}_k \boldsymbol{C}_k) \boldsymbol{\Sigma}_k^-, \qquad \text{(Variance Update)}$$

$$\boldsymbol{\mu}_0^+ = \boldsymbol{\mu}_0, \qquad \text{(Initial State Mean)}$$

$$\boldsymbol{\Sigma}_0^+ = \boldsymbol{\Sigma}_0, \qquad \text{(Initial State Variance)}$$

where $\boldsymbol{\Sigma}_k^-, \boldsymbol{\Sigma}_k^+ \in \mathbb{R}^{n_x \times n_x}$ are the prior and posterior variances at time instant k respectively. The variables $\boldsymbol{\mu}_k^-, \boldsymbol{\mu}_k^+ \in \mathbb{R}^{n_x}$ are the prior and posterior mean of $\boldsymbol{x}_k$ and denote the estimate of the true state, and $\boldsymbol{K}_k$ is the Kalman gain, at time k.

Conventionally, $\boldsymbol{R}_k$ is given and the Kalman filtering results in state estimates with minimum error variance. That is, given $\boldsymbol{R}_k$, we determine $\boldsymbol{K}_k$ that results in the optimal posterior error variance, i.e.

$$\boldsymbol{K}_k := \arg \min_{\boldsymbol{K}_k} \mathbf{tr}\left[\boldsymbol{\Sigma}_k^+\right],$$

where

$$\boldsymbol{\Sigma}_k^+ := \boldsymbol{\Sigma}_k^- - \boldsymbol{\Sigma}_k^- \boldsymbol{C}_k^T \left(\boldsymbol{C}_k \boldsymbol{\Sigma}_k^- \boldsymbol{C}_k^T + \boldsymbol{R}_k\right)^{-1} \boldsymbol{C}_k \boldsymbol{\Sigma}_k^-.$$

In this paper, we treat $\boldsymbol{R}_k$ as a *variable*, and for a given desired $\boldsymbol{\Sigma}^d$, ask the following question: what should optimal $\boldsymbol{R}_k$ be such that $\boldsymbol{\Sigma}_k^+ \leq \boldsymbol{\Sigma}^d$? Assuming $\boldsymbol{R}_k$ to be

diagonal, i.e.

$$\boldsymbol{R}_k := \mathbf{diag}\left(\boldsymbol{r}_k\right),$$

where $\boldsymbol{r}_k := \begin{bmatrix} r_1 & r_2 & \cdots & r_{n_{y_k}} \end{bmatrix}^T$, with $r_i > 0$; and optimality is determined with respect to some function of $\boldsymbol{r}_k$.

In this paper, we choose the cost function to determine the *noisiest* set of sensors, i.e. maximize $\mathbf{tr}\left[\boldsymbol{R}_k\right]$, for which $\boldsymbol{\Sigma}_k^+ \leq \boldsymbol{\Sigma}^d$. As shown later, it is convenient to formulate the problem in terms of sensor precisions, defined by $\boldsymbol{S}_k$, which is the inverse of sensor noise $\boldsymbol{R}_k$, i.e. $\boldsymbol{S}_k := \boldsymbol{R}_k^{-1}$, resulting in $s_i := 1/r_i$. Defining $\boldsymbol{s} := \begin{bmatrix} s_1 & s_2 & \cdots & s_{n_y} \end{bmatrix}^T$, we can determine the least precise sensors, i.e. minimize $\mathbf{tr}\left[\boldsymbol{S}_k\right]$, for which $\boldsymbol{\Sigma}_k^+ \leq \boldsymbol{\Sigma}^d$ is guaranteed. Since more precise sensors are more expensive, satisfying required accuracy with least precise sensors has favorable economic implications.

Minimization of $\mathbf{tr}\left[\boldsymbol{S}_k\right]$ also has sparseness implications as $\mathbf{tr}\left[\boldsymbol{S}_k\right]$, for $\boldsymbol{S}_k \geq 0$, is equivalent to $\|\boldsymbol{s}\|_1$. Since it is well-known that l_1 norm is sparseness promoting, minimizing $\mathbf{tr}\left[\boldsymbol{S}_k\right]$ will result in a sparse solution that satisfies $\boldsymbol{\Sigma}_k^+ \leq \boldsymbol{\Sigma}^d$, if a sparse solution exists for the problem. Consequently, sensors with zero precisions would not contribute to achieving $\mathbf{tr}\left[\boldsymbol{\Sigma}_k^+\right] \leq \boldsymbol{\Sigma}^d$, and thus can be removed from the system.

With this background, we next present the formulation to determine the optimal precision for multi-rate information fusion in the Kalman filtering framework.

3.3 Optimal Sensing Precision for Update after One Time Step

Here we present the formulation that determines the optimal sensing precision that guarantees bounded estimation error after one update. The problem is formulated in a batch processing framework, where m measurements are collected before the state

is updated. Specifically, given the state uncertainty at time t_{km}, the objective is to determine the precisions of these m measurements, such that the state estimate at time $t_{(k+1)m}$ has bounded error.

Consider sensing over m time steps, as shown below where $\boldsymbol{y}_{km+j} \in \mathbb{R}^{n_{y_{km+j}}}$ and
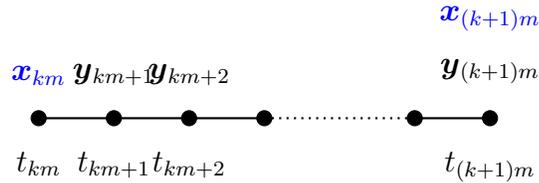
Figure 3.1: Multi-rate measurements over m time steps. Reprinted with permission from [18].

captures $n_{y_{km+j}} \in \mathbb{Z}^+$ measurements at time $(km + j)$. This allows us to model multi-rate sensing, with m being the least-common-multiple of the various sensing intervals. With each measurement $\boldsymbol{y}_{km+j}$, for $j = 1, \cdots, m$, we associate sensor noises $\boldsymbol{r}_{km+j} \in \mathbb{R}^{n_{y_{km+j}}}$.

In conventional Kalman filtering, the sensor noises $\boldsymbol{r}_{km+j}$ are known and the objective is to estimate the state at time $t_{(k+1)m}$, given the posterior at time t_{km} and measurements $\boldsymbol{y}_{km+1}, \cdots, \boldsymbol{y}_{(k+1)m}$, at times $t_{km+1}, \cdots, t_{(k+1)m}$. This scenario is common in control system applications where the control-loop is band limited (to prevent excitation of high-frequency dynamics), but the sensing loop can be faster. In such a scenario, the Kalman filter determines the state estimate every m time steps, by batch-processing m measurements. In this paper, we are interested in maximizing $\boldsymbol{r}_{km+j}$ for which the posterior state estimation error at time $t_{(k+1)m}$ satisfies a given upper bound.

3.3.1 Augmented Dynamical System

To determine the posterior at time $t_{(k+1)m}$, we need to propagate the state uncertainty from t_{km} to $t_{(k+1)m}$, to obtain the prior at $t_{(k+1)m}$. This is done by lifting discrete-time signals defined over times t_k to signals defined over times t_{km}, for $k = 0, 1, \cdots, \infty$. We define lifted signals $\boldsymbol{X}_k \in \mathbb{R}^{mn_x}$ and $\boldsymbol{Y}_k \in \mathbb{R}^{n_{y_{k,m}}}$ as a vector with m consecutive state vectors and measurements stacked vertically, respectively, i.e.

$$\boldsymbol{X}_k := \begin{pmatrix} \boldsymbol{x}_{km+1} \\ \vdots \\ \boldsymbol{x}_{(k+1)m} \end{pmatrix}, \tag{3.2}$$

$$\boldsymbol{Y}_k := \begin{pmatrix} \boldsymbol{y}_{km+1} \\ \vdots \\ \boldsymbol{y}_{(k+1)m} \end{pmatrix}, \tag{3.3}$$

where $m \geq 1$, and $n_{y_{k,m}} = \sum_{j=1}^{m} n_{y_{km+j}}$.

Using the system defined in equation (3.1a), the state $\boldsymbol{X}_k$ can be expressed as

$$\boldsymbol{X}_k = \boldsymbol{\mathcal{A}}_k \boldsymbol{x}_{km} + \boldsymbol{\mathcal{B}}_k \boldsymbol{W}_k, \tag{3.4}$$

where

$$\mathcal{A}_k := \begin{bmatrix} \boldsymbol{A}_{km} \\ \boldsymbol{A}_{km+1}\boldsymbol{A}_{km} \\ \vdots \\ \displaystyle\prod_{i=0}^{m-1} \boldsymbol{A}_{km+i} \end{bmatrix}, \tag{3.5}$$

$$\mathcal{B}_k := \begin{bmatrix} \boldsymbol{B}_{km} & 0 & ... & 0 \\ \boldsymbol{A}_{km+1}\boldsymbol{B}_{km} & \boldsymbol{B}_{km+1} & ... & 0 \\ \vdots & \vdots & & \vdots \\ \displaystyle\prod_{i=1}^{m-1} \boldsymbol{A}_{km+i}\boldsymbol{B}_{km} & ... & ... & \boldsymbol{B}_{km+m-1} \end{bmatrix}, \tag{3.6}$$

$$\boldsymbol{W}_k := \begin{pmatrix} \boldsymbol{w}_{km} \\ \vdots \\ \boldsymbol{w}_{km+m-1} \end{pmatrix}, \tag{3.7}$$

with

$$\prod_{i=i_1}^{i_2} \boldsymbol{A}_{k+i} := \boldsymbol{A}_{k+i_2} \times \cdots \times \boldsymbol{A}_{k+i_1+1}\boldsymbol{A}_{k+i_1}, \tag{3.8}$$

if $i_2 \geq i_1$, and

$$\prod_{i=i_1}^{i_2} \boldsymbol{A}_{k+i} := \boldsymbol{I}_{n_x}, \tag{3.9}$$

if $i_2 < i_1$.

The augment measurement model from (3.1b) is

$$\boldsymbol{Y}_k = \boldsymbol{\mathcal{C}}_k \boldsymbol{X}_k + \boldsymbol{N}_k, \tag{3.10}$$

where

$$\boldsymbol{\mathcal{C}}_k := \mathbf{diag}\left(\boldsymbol{C}_{km+1}, ..., \boldsymbol{C}_{(k+1)m}\right), \text{ and} \tag{3.11}$$

$$\boldsymbol{N}_k := \begin{pmatrix} \boldsymbol{n}_{km+1} \\ \vdots \\ \boldsymbol{n}_{(k+1)m} \end{pmatrix}. \tag{3.12}$$

We define the augment process noise and observation noise variances as:

$$\begin{aligned} \boldsymbol{\mathcal{Q}}_k &:= \mathbb{E}\left[\boldsymbol{W}_k \boldsymbol{W}_k^T\right], \\ &= \mathbf{diag}\left(\mathbb{E}\left[\boldsymbol{w}_{km} \boldsymbol{w}_{km}^T\right], \cdots, \mathbb{E}\left[\boldsymbol{w}_{km+m-1} \boldsymbol{w}_{km+m-1}^T\right]\right), \\ &= \mathbf{diag}\left(\boldsymbol{Q}_{km}, \cdots, \boldsymbol{Q}_{km+m-1}\right), \end{aligned} \tag{3.13}$$

and

$$\begin{aligned} \boldsymbol{\mathcal{R}}_k &:= \mathbb{E}\left[\boldsymbol{N}_k \boldsymbol{N}_k^T\right], \\ &= \mathbf{diag}\left(\mathbb{E}\left[\boldsymbol{n}_{km} \boldsymbol{n}_{km}^T\right], \cdots, \mathbb{E}\left[\boldsymbol{n}_{km+m-1} \boldsymbol{n}_{km+m-1}^T\right]\right), \\ &= \mathbf{diag}\left(\boldsymbol{R}_{km}, \cdots, \boldsymbol{R}_{km+m-1}\right), \end{aligned} \tag{3.14}$$

where

$$\boldsymbol{Q}_{km+j} := \mathbb{E}\left[\boldsymbol{w}_{km+j}\boldsymbol{w}_{km+j}^T\right],$$

$$\boldsymbol{R}_{km+j} := \mathbb{E}\left[\boldsymbol{n}_{km+j}\boldsymbol{n}_{km+j}^T\right].$$

3.3.2 Uncertainty Propagation and Measurement Update

The prior statistics of $\boldsymbol{X}_k$ are related to the posterior statistics of $\boldsymbol{x}_{km}$ i.e. $(\boldsymbol{\mu}_{km}^+, \boldsymbol{\Sigma}_{km}^+)$ as

$$\bar{\boldsymbol{X}}_k^- := \mathbb{E}\left[\boldsymbol{X}_k\right] = \boldsymbol{\mathcal{A}}_k\boldsymbol{\mu}_{km}^+, \tag{3.15}$$

$$\boldsymbol{P}_k^- := \mathbb{E}\left[(\boldsymbol{X}_k^- - \bar{\boldsymbol{X}}_k^-)(\boldsymbol{X}_k^- - \bar{\boldsymbol{X}}_k^-)^T\right],$$

$$= \boldsymbol{\mathcal{A}}_k\boldsymbol{\Sigma}_{km}^+\boldsymbol{\mathcal{A}}_k^T + \boldsymbol{\mathcal{B}}_k\boldsymbol{\mathcal{Q}}_k\boldsymbol{\mathcal{B}}_k^T. \tag{3.16}$$

Prior statistics of the augmented state, i.e. $\bar{\boldsymbol{X}}_k^-$ and $\boldsymbol{P}_k^-$, can be updated using the augmented measurements $\boldsymbol{Y}_k$ to obtain posterior $(\bar{\boldsymbol{X}}_k^+, \boldsymbol{P}_k^+)$, using similar steps as in standard Kalman filtering, i.e.

$$\bar{\boldsymbol{X}}_k^+ := \boldsymbol{\mathcal{A}}_k\boldsymbol{\mu}_{km-q}^+ + \boldsymbol{\mathcal{K}}_k(\boldsymbol{Y}_k - \boldsymbol{\mathcal{C}}_k\boldsymbol{\mathcal{A}}_k\boldsymbol{\mu}_{km-q}^+), \tag{3.17}$$

$$\boldsymbol{P}_k^+ := (\mathbf{I} - \boldsymbol{\mathcal{K}}_k\boldsymbol{\mathcal{C}}_k)\boldsymbol{P}_k^-, \tag{3.18}$$

where

$$\boldsymbol{\mathcal{K}}_k := \boldsymbol{P}_k^-\boldsymbol{\mathcal{C}}_k^T\left[\boldsymbol{\mathcal{C}}_k\boldsymbol{P}_k^-\boldsymbol{\mathcal{C}}_k^T + \boldsymbol{\mathcal{R}}_k\right]^{-1}. \tag{3.19}$$

82

The state at time $t_{(k+1)m}$ can be determined from $\boldsymbol{X}_k$ as

$$\boldsymbol{x}_{(k+1)m} := \boldsymbol{M}_m \boldsymbol{X}_k,$$

where

$$\boldsymbol{M}_m := \begin{bmatrix} \boldsymbol{0}_{n_x \times n_x(m-1)} & \boldsymbol{I}_{n_x} \end{bmatrix}.$$

The posterior statistics of $\boldsymbol{x}_{(k+1)m}$ can then be determined from the posterior statistics of $\boldsymbol{X}_k$ using

$$\boldsymbol{\mu}_{(k+1)m}^+ := \boldsymbol{M}_m \bar{\boldsymbol{X}}_k^+, \tag{3.20}$$

$$\boldsymbol{\Sigma}_{(k+1)m}^+ := \boldsymbol{M}_m \boldsymbol{P}_k^+ \boldsymbol{M}_m^T. \tag{3.21}$$

3.3.3 Optimal Sensor Precision for a Single Measurement Update

Here we present a convex optimization framework for determining the nosiest sensors, for which the estimation errors are below a given upper bound after *one measurement update*. That is, maximize $\mathbf{tr}\left[\boldsymbol{\mathcal{R}}_k\right]$ or minimize $\mathbf{tr}\left[\boldsymbol{\mathcal{S}}_k\right]$ where $\boldsymbol{\mathcal{S}}_k := \boldsymbol{\mathcal{R}}_k^{-1}$, for which $\boldsymbol{\Sigma}_{(k+1)m}^+ \leq \boldsymbol{\Sigma}^d$, given $\boldsymbol{\Sigma}_{km}^+$. This is achieved by solving the following optimization problem.

Theorem 11. *Optimal sensor precision $\boldsymbol{s}_k \in \mathbb{R}^{n_{y_{k,m}}} \geq 0$, which satisfies $\boldsymbol{\Sigma}_{(k+1)m}^+ \leq$*

Σ^d, is given by the solution of the following optimization problem,

$$\left.\begin{aligned} &\min_{\boldsymbol{s}_k} \mathbf{tr}\left[\boldsymbol{W}\boldsymbol{S}_k\right], \;\; \textit{subject to} \\[6pt] &\begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_m \boldsymbol{P}_k^- \\[4pt] (*)^T & \boldsymbol{L} + \boldsymbol{L}\boldsymbol{S}_k \boldsymbol{L} \end{bmatrix} \geq 0, \\[6pt] &0 \leq \boldsymbol{s}_k \leq \boldsymbol{s}_k^{max}, \end{aligned}\right\} \qquad (3.22)$$

where $(*)^T$ represents symmetric terms, and

$$\boldsymbol{S}_k := \mathbf{diag}\left(\boldsymbol{s}_k\right),$$

$$\boldsymbol{M}_{11} := \Sigma^d - \boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{M}_m^T + \boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{L}^{-1} \boldsymbol{P}_k^- \boldsymbol{M}_m^T,$$

$$\boldsymbol{L} := \boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k^- \boldsymbol{\mathcal{C}}_k^T.$$

The variable $\boldsymbol{W}$ is a diagonal matrix, which is user defined and serves as a normalizing weight on $\boldsymbol{S}_k$.

Proof. Inequality $\Sigma^+_{(k+1)m} \leq \Sigma^d$, is equivalent to

$$\boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{M}_m^T -$$

$$\boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{\mathcal{C}}_k^T \left(\boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k^- \boldsymbol{\mathcal{C}}_k^T + \boldsymbol{\mathcal{R}}_k\right)^{-1} \boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k^- \boldsymbol{M}_m^T \leq \Sigma^d. \quad (3.23)$$

Using matrix-inversion lemma on $\left(\boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k^- \boldsymbol{\mathcal{C}}_k^T + \boldsymbol{\mathcal{R}}_k\right)^{-1}$, (3.23) simplifies to

$$\boldsymbol{\Sigma}^d - \boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{M}_m^T + \boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{L}^{-1} \boldsymbol{P}_k^- \boldsymbol{M}_m^T$$

$$- \boldsymbol{M}_m \boldsymbol{P}_k^- \left(\boldsymbol{L} + \boldsymbol{L} \boldsymbol{\mathcal{R}}_k^{-1} \boldsymbol{L} \right)^{-1} \boldsymbol{P}_k^- \boldsymbol{M}_m^T \geq 0,$$

where $\boldsymbol{L} := \boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k^- \boldsymbol{\mathcal{C}}_k^T$. Substituting $\boldsymbol{\mathcal{S}}_k := \boldsymbol{\mathcal{R}}_k^{-1}$, and using Schur complement we get the following linear matrix inequality in $\boldsymbol{Z}$ and $\boldsymbol{\mathcal{S}}_k$,

$$\begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_m \boldsymbol{P}_k^- \\ (\ast)^T & \boldsymbol{L} + \boldsymbol{L} \boldsymbol{\mathcal{S}}_k \boldsymbol{L} \end{bmatrix} \geq 0, \tag{3.24}$$

where

$$\boldsymbol{M}_{11} := \boldsymbol{\Sigma}^d - \boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{M}_m^T + \boldsymbol{M}_m \boldsymbol{P}_k^- \boldsymbol{L}^{-1} \boldsymbol{P}_k^- \boldsymbol{M}_m^T.$$

Noting that $\boldsymbol{\mathcal{S}}_k := \mathbf{diag}\,(\boldsymbol{s}_k)$, for $\boldsymbol{s}_k \in \mathbb{R}^{n_{y_{k,m}}} \geq 0$, optimal precision can be determined by minimizing the cost function $\mathbf{tr}\,[\boldsymbol{W} \boldsymbol{\mathcal{S}}_k]$. Practical considerations may upper-bound maximum precision, which is incorporated in the formulation using the constraint

$$\boldsymbol{s}_k \leq \boldsymbol{s}_k^{\max}. \tag{3.25}$$

Inequalities (3.24), (3.25), along with minimization of $\mathbf{tr}\,[\boldsymbol{W} \boldsymbol{\mathcal{S}}_k]$, result in the optimization problem in (3.22). $\qquad\square$

Theorem 11 presents the optimization problem for determining optimal precisions to bound the errors after *one time step* at time $t_{(k+1)m}$, given the posterior uncertainty

at time t_{km}.

Remark 1. It is also possible to bound the trace of $\mathbf{\Sigma}^+_{(k+1)m}$, i.e. $\mathbf{tr}\left[\mathbf{\Sigma}^+_{(k+1)m}\right] \le \gamma_d$, where γ_d is user specified. With this relaxation, the optimization problem in (3.22) modifies to

$$
\left.\begin{aligned}
&\min_{\boldsymbol{s}_k, \boldsymbol{F}} \mathbf{tr}\left[\boldsymbol{W}\boldsymbol{\mathcal{S}}_k\right], \text{ subject to} \\[2mm]
&\begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_m \boldsymbol{P}^-_k \\ (*)^T & \boldsymbol{L} + \boldsymbol{L}\boldsymbol{\mathcal{S}}_k\boldsymbol{L} \end{bmatrix} \ge 0, \\[2mm]
&0 \le \boldsymbol{s}_k \le \boldsymbol{s}^{\max}_k, \\[2mm]
&\boldsymbol{F} \ge 0, \\[2mm]
&\mathbf{tr}\left[\boldsymbol{F}\right] \le \gamma_d,
\end{aligned}\right\} \tag{3.26}
$$

where

$$
\boldsymbol{\mathcal{S}}_k := \mathbf{diag}\left(\boldsymbol{s}_k\right),
$$

$$
\boldsymbol{M}_{11} := \boldsymbol{F} - \boldsymbol{M}_m \boldsymbol{P}^-_k \boldsymbol{M}^T_m + \boldsymbol{M}_m \boldsymbol{P}^-_k \boldsymbol{L}^{-1} \boldsymbol{P}^-_k \boldsymbol{M}^T_m,
$$

$$
\boldsymbol{L} := \boldsymbol{\mathcal{C}}_k \boldsymbol{P}^-_k \boldsymbol{\mathcal{C}}^T_k,
$$

and $\boldsymbol{W}$ is the normalizing weight on $\boldsymbol{\mathcal{S}}_k$, as in theorem 11. In (3.26), a new variable $\boldsymbol{F} \in \mathbb{S}^{n_x}_+$ is introduced to impose $\mathbf{tr}\left[\mathbf{\Sigma}^+_{(k+1)m}\right] \le \gamma$, where $\mathbb{S}^{n_x}_+$ denotes space of symmetric positive definite matrices of dimension $n_x \times n_x$.

The above optimization problems require $\boldsymbol{L}^{-1}$, which could be problematic in some applications. Next, we present an alternate formulation, which avoids the computation of $\boldsymbol{L}^{-1}$. This is stated as the following theorem.

Theorem 12. *Optimal sensor precision* $\boldsymbol{s}_k \in \mathbb{R}^{n_{y_{k,m}}} \ge 0$, *which satisfies*

$\mathbf{tr}\left[\mathbf{\Sigma}_{(k+1)m}^+\right] \le \gamma_d$, is given by the solution of the following optimization problem,

$$
\left.
\begin{array}{c}
\min\limits_{\boldsymbol{s}_k, \boldsymbol{\mathcal{K}}_k, \boldsymbol{F}} \mathbf{tr}\left[\boldsymbol{W}\boldsymbol{\mathcal{S}}_k\right], \ \ subject \ to \\[2mm]
\begin{bmatrix}
\boldsymbol{F} & \boldsymbol{M}_{12}\sqrt{\boldsymbol{P}_k^-} & \boldsymbol{\mathcal{K}}_k \\[2mm]
(*)^T & \boldsymbol{I}_{mn_x} & \boldsymbol{0}_{mn_x \times n_{y_{k,m}}} \\[2mm]
(*)^T & (*)^T & \boldsymbol{\mathcal{S}}_k
\end{bmatrix} \ge 0 \\[8mm]
0 \le \boldsymbol{s}_k \le \boldsymbol{s}_k^{max},
\end{array}
\right\}
\tag{3.27}
$$

where $\boldsymbol{M}_{12} := \boldsymbol{M}_m(\boldsymbol{I}_{mn_x} - \boldsymbol{\mathcal{K}}_k\boldsymbol{\mathcal{C}}_k)$. The variable $\boldsymbol{W}$ is a diagonal matrix, which is user defined and serves as a normalizing weight on $\boldsymbol{\mathcal{S}}_k$.

Proof. We can write the posterior error variance as

$$
\boldsymbol{P}_k^+ = (\boldsymbol{I}_{mn_x} - \boldsymbol{\mathcal{K}}_k\boldsymbol{\mathcal{C}}_k)\boldsymbol{P}_k^-(\boldsymbol{I}_{mn_x} - \boldsymbol{\mathcal{K}}_k\boldsymbol{\mathcal{C}}_k)^T + \boldsymbol{\mathcal{K}}_k\boldsymbol{\mathcal{R}}_k\boldsymbol{\mathcal{K}}_k^T.
\tag{3.28}
$$

The optimal $\boldsymbol{\mathcal{K}}_k$ is determined by minimizing $\mathbf{tr}\left[\boldsymbol{P}_k^+\right]$ and is given by (3.19). However, in this formulation, we leave $\boldsymbol{\mathcal{K}}_k \in \mathbb{R}^{mn_x \times n_{y_{k,m}}}$ as a variable, and write $\mathbf{tr}\left[\boldsymbol{M}_m\boldsymbol{P}_k^+\boldsymbol{M}_m^T\right] \le \gamma_d$ equivalently as

$$
\boldsymbol{F} - \boldsymbol{M}_{12}\boldsymbol{P}_k^-\boldsymbol{M}_{12}^T - \boldsymbol{M}_m\boldsymbol{\mathcal{K}}_k^T\boldsymbol{\mathcal{R}}_k\boldsymbol{\mathcal{K}}_k^T\boldsymbol{M}_m^T \ge 0,
$$

$$
\mathbf{tr}\left[\boldsymbol{F}\right] \le \gamma_d,
$$

where $\boldsymbol{M}_{12} := \boldsymbol{M}_m(\boldsymbol{I}_{mn_x} - \boldsymbol{\mathcal{K}}_k\boldsymbol{\mathcal{C}}_k)$ and $\boldsymbol{F} \in \mathbb{S}_+^{n_x}$. Representing $\sqrt{\boldsymbol{P}_k^-}$ as the principal matrix square-root of $\boldsymbol{P}_k^-$, substituting $\boldsymbol{\mathcal{S}}_k := \boldsymbol{\mathcal{R}}_k^{-1}$, and using Schur complement we

get the following LMI,

$$
\begin{bmatrix}
\boldsymbol{F} & \boldsymbol{M}_{12}\sqrt{\boldsymbol{P}_k^-} & \boldsymbol{M}_m \boldsymbol{\mathcal{K}}_k \\
(*)^T & \boldsymbol{I}_{mn_x} & \boldsymbol{0}_{mn_x \times n_{y_{k,m}}} \\
(*)^T & (*)^T & \boldsymbol{S}_k
\end{bmatrix} \geq 0.
$$

Combining the inequalities and minimizing $\mathbf{tr}\left[\boldsymbol{W}\boldsymbol{S}_k\right]$ we get (3.27). $\qquad\square$

Remark 2. It is easy to show that the optimal gain $\boldsymbol{\mathcal{K}}_k^*$, obtained by solving the optimization problem in (3.27), is the minimum variance gain for the noise variance $\boldsymbol{\mathcal{R}}_k^* := (\boldsymbol{S}_k^*)^{-1}$.

Let $\boldsymbol{P}_k^+(\boldsymbol{\mathcal{R}}_k, \boldsymbol{\mathcal{K}}_k)$ be defined by (3.28), and the minimum variance posterior $\boldsymbol{P}_{k,\mathrm{mv}}^+(\boldsymbol{\mathcal{R}}_k)$ is defined by

$$
\boldsymbol{P}_{k,\mathrm{mv}}^+(\boldsymbol{\mathcal{R}}_k) := \boldsymbol{P}_k^- - \boldsymbol{P}_k \boldsymbol{\mathcal{C}}_k^T \left(\boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k \boldsymbol{\mathcal{C}}_k^T + \boldsymbol{\mathcal{R}}_k\right)^{-1} \boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k.
$$

It is easy to verify that $\boldsymbol{P}_{k,\mathrm{mv}}^+(\boldsymbol{\mathcal{R}}_k)$ is a monotonic function of $\boldsymbol{\mathcal{R}}_k$, i.e. $\boldsymbol{P}_{k,\mathrm{mv}}^+(\boldsymbol{\mathcal{R}}_k) \geq \boldsymbol{P}_{k,\mathrm{mv}}^+(\boldsymbol{\mathcal{R}}_k^*)$ for any $\boldsymbol{\mathcal{R}} \geq \boldsymbol{\mathcal{R}}_k^*$.

Since by definition,

$$
\boldsymbol{P}_k^+(\boldsymbol{\mathcal{R}}_k^*, \boldsymbol{\mathcal{K}}_k^*) \geq \boldsymbol{P}_{k,\mathrm{mv}}^+(\boldsymbol{\mathcal{R}}_k^*),
$$

$\boldsymbol{\mathcal{R}}_k^*$ is maximum when

$$
\boldsymbol{P}_k^+(\boldsymbol{\mathcal{R}}_k^*, \boldsymbol{\mathcal{K}}_k^*) = \boldsymbol{P}_{k,\mathrm{mv}}^+(\boldsymbol{\mathcal{R}}_k^*),
$$

or

$$
\boldsymbol{\mathcal{K}}_k^* = \boldsymbol{\mathcal{K}}_{k,\mathrm{mv}}^* := \boldsymbol{P}_k \boldsymbol{\mathcal{C}}_k^T \left(\boldsymbol{\mathcal{C}}_k \boldsymbol{P}_k \boldsymbol{\mathcal{C}}_k^T + \boldsymbol{\mathcal{R}}_k\right)^{-1}.
$$

Remark 3. In both theorem 11 and theorem 12, the sparseness of the solution

can be improved by iteratively solving the optimization problem (3.22) with weights $\boldsymbol{W}_{j+1} := (\boldsymbol{S}_k^* + \epsilon \boldsymbol{I})_j^{-1}$, with $\boldsymbol{W}_1 := \boldsymbol{I}_{n_{y_{k,m}}}$, where subscript j denotes the iteration index [67, 11].

3.4 Optimal Sensor Precision for Bounded Steady-State Errors

In this section, we present the result that determines the optimal sensor precision for bounded steady-state error, assuming the system to be m-periodic. If the system in (3.1a) is m-periodic, i.e. $\boldsymbol{A}_{km+j} = \boldsymbol{A}_{(k+1)m+j}$, $\boldsymbol{B}_{km+j} = \boldsymbol{B}_{(k+1)m+j}$, and $\boldsymbol{C}_{km+j} = \boldsymbol{C}_{(k+1)m+j}$ for $j = 1, \cdots, m$; it will be of interest to determine the sensing precisions that bound the steady-state errors, assuming it exists.

3.4.1 Augmented Dynamical System

From (3.4), the augmented dynamics of the m-periodic system is given by,

$$\boldsymbol{x}_{(k+1)m} = \boldsymbol{M}_m \boldsymbol{\mathcal{A}}_k \boldsymbol{x}_{km} + \boldsymbol{M}_m \boldsymbol{\mathcal{B}}_k \boldsymbol{W}_k. \tag{3.29}$$

In this section we generalize the sensor model in (3.1b) by including the process noise in the measurement. This scenario, for example, occurs in measurements from accelerometers where the disturbance forces algebraically impact accelerations. The new measurement model is therefore,

$$\boldsymbol{y}_k = \boldsymbol{C}_k \boldsymbol{x}_k + \boldsymbol{D}_k \boldsymbol{w}_k + \boldsymbol{n}_k. \tag{3.30}$$

Consequently, the augmented sensor model is

$$\boldsymbol{Y}_k = \boldsymbol{\mathcal{C}}_k \boldsymbol{X}_k + \boldsymbol{\mathcal{D}}_k \boldsymbol{W}_k + \boldsymbol{N}_k,$$

$$= \boldsymbol{\mathcal{C}}_k (\boldsymbol{\mathcal{A}}_k \boldsymbol{x}_{km} + \boldsymbol{\mathcal{B}}_k \boldsymbol{W}_k) + \boldsymbol{\mathcal{D}}_k \boldsymbol{W}_k + \boldsymbol{N}_k,$$

$$= \boldsymbol{\mathcal{C}}_k \boldsymbol{\mathcal{A}}_k \boldsymbol{x}_{km} + (\boldsymbol{\mathcal{C}}_k \boldsymbol{\mathcal{B}}_k + \boldsymbol{\mathcal{D}}_k) \boldsymbol{W}_k + \boldsymbol{N}_k, \tag{3.31}$$

where $\boldsymbol{\mathcal{D}}_k := \mathbf{diag}\left(\boldsymbol{D}_{km+1}, ..., \boldsymbol{D}_{(k+1)m}\right)$.

The presence of $\boldsymbol{W}_k$ in (3.31) makes derivation of the Kalman filter complicated. This is circumvented by assuming the process noise to be colored, or filtered white noise. That is, we model the process noise as

$$\boldsymbol{Z}_{k+1} = \boldsymbol{\mathcal{G}} \boldsymbol{Z}_k + \boldsymbol{\mathcal{H}} \boldsymbol{\Lambda}_k, \ \boldsymbol{W}_k = \boldsymbol{Z}_k, \tag{3.32}$$

where $\boldsymbol{\Lambda}_k$ is white noise, $\boldsymbol{Z}_k$ is the filter state, and the pair $(\boldsymbol{\mathcal{G}}_k, \boldsymbol{\mathcal{H}}_k)$ defines the filter.

If white noise $\boldsymbol{\lambda}_k \in \mathbb{R}^{n_w}$ is filtered via

$$\boldsymbol{z}_{km+1} = \boldsymbol{G} \boldsymbol{z}_{km} + \boldsymbol{H} \boldsymbol{\lambda}_{km}, \tag{3.33}$$

then for the augmented system,

$$\boldsymbol{Z}_k := \begin{pmatrix} \boldsymbol{z}_{km} \\ \vdots \\ \boldsymbol{z}_{(k+1)m-1} \end{pmatrix}, \ \boldsymbol{\Lambda}_k := \begin{pmatrix} \boldsymbol{\lambda}_{km} \\ \vdots \\ \boldsymbol{\lambda}_{(k+1)m-1} \end{pmatrix}, \tag{3.34}$$

$\boldsymbol{\mathcal{G}} := \boldsymbol{I}_q \otimes \boldsymbol{G}$, and $\boldsymbol{\mathcal{H}} := \boldsymbol{I}_q \otimes \boldsymbol{H}$, where $\boldsymbol{G} \in \mathbb{R}^{n_w \times n_w}$ and $\boldsymbol{H} \in \mathbb{R}^{n_w \times n_w}$ define the

filter in (3.33).

Introducing a new state variable

$$\boldsymbol{\Gamma}_k := \begin{bmatrix} \boldsymbol{x}_{km} \\ \boldsymbol{Z}_k \end{bmatrix} \in \mathbb{R}^{N_x}, \tag{3.35}$$

where $N_x := n_x + mn_w$, we can write the dynamics of $\boldsymbol{\Gamma}_k$ and measurement $\boldsymbol{Y}_k$ as

$$\boldsymbol{\Gamma}_{k+1} = \boldsymbol{\mathcal{A}}_m \boldsymbol{\Gamma}_k + \boldsymbol{\mathcal{B}}_m \boldsymbol{\Lambda}_k, \tag{3.36a}$$

$$\boldsymbol{Y}_k = \boldsymbol{\mathcal{C}}_m \boldsymbol{\Gamma}_k + \boldsymbol{N}_k. \tag{3.36b}$$

where

$$\boldsymbol{\mathcal{A}}_m := \begin{bmatrix} \boldsymbol{M}_m \boldsymbol{\mathcal{A}}_k & \boldsymbol{M}_m \boldsymbol{\mathcal{B}}_k \\ \boldsymbol{0}_{qn_w \times n_x} & \boldsymbol{\mathcal{G}} \end{bmatrix}, \tag{3.37a}$$

$$\boldsymbol{\mathcal{B}}_m := \begin{bmatrix} \boldsymbol{0}_{n_x \times qn_w} \\ \boldsymbol{\mathcal{H}} \end{bmatrix}, \tag{3.37b}$$

$$\boldsymbol{\mathcal{C}}_m := \begin{bmatrix} \boldsymbol{\mathcal{C}}_k \boldsymbol{\mathcal{A}}_k & (\boldsymbol{\mathcal{C}}_k \boldsymbol{\mathcal{B}}_k + \boldsymbol{\mathcal{D}}_k) \end{bmatrix}. \tag{3.37c}$$

Note that for the m-periodic system, matrices $\boldsymbol{\mathcal{A}}_m$, $\boldsymbol{\mathcal{B}}_m$, and $\boldsymbol{\mathcal{C}}_m$ are time invariant.

States $\boldsymbol{x}_{km}$ can be recovered from $\boldsymbol{\Gamma}_k$ as

$$\boldsymbol{x}_{km} = \boldsymbol{M}_x \boldsymbol{\Gamma}_k, \tag{3.38}$$

where $\boldsymbol{M}_x$ is a mask-matrix defined by

$$\boldsymbol{M}_x := \begin{bmatrix} \boldsymbol{I}_{n_x} & \boldsymbol{0}_{n_x \times q n_w} \end{bmatrix}. \tag{3.39}$$

We next define

$$\boldsymbol{\mathcal{Q}}_m := \mathbb{E}\left[\boldsymbol{\Lambda}_k \boldsymbol{\Lambda}_k^T\right], \tag{3.40a}$$

$$\boldsymbol{\mathcal{R}}_m := \mathbb{E}\left[\boldsymbol{N}_k \boldsymbol{N}_k^T\right]. \tag{3.40b}$$

For steady-state analysis, we assume $(\boldsymbol{\mathcal{A}}_m, \boldsymbol{\mathcal{R}}_m^{1/2}\boldsymbol{\mathcal{C}}_m)$ is detectable and $(\boldsymbol{\mathcal{A}}_m, (\boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}_m^T)^{1/2})$ is stabilizable.

3.4.2 Steady-state Variance

Let $\bar{\boldsymbol{\Gamma}}_k^-$ and $\boldsymbol{P}_k^-$ be the prior mean and variance of $\boldsymbol{\Gamma}_k$ at time k. This defines the prior random variable $\boldsymbol{\Gamma}_k^- \sim \mathcal{N}(\bar{\boldsymbol{\Gamma}}_k^-, \boldsymbol{P}_k^-)$, where $\mathcal{N}(\cdot, \cdot)$ defines a Gaussian distribution.

In Kalman filtering we assume the posterior is a linear function of the prior and the measurement, i.e.

$$\boldsymbol{\Gamma}_k^+ := (\boldsymbol{I}_{N_x} - \boldsymbol{\mathcal{K}}_k \boldsymbol{\mathcal{C}}_m)\boldsymbol{\Gamma}_k^- + \boldsymbol{\mathcal{K}}_k \boldsymbol{Y}_k, \tag{3.41}$$

where $\boldsymbol{\mathcal{K}}_k \in \mathbb{R}^{N_x \times n_{y_{k,m}}}$ is the unknown gain.

The coefficient $\boldsymbol{\mathcal{K}}_k$ is determined by minimizing the posterior variance. However, in this formulation, we leave it as a free variable along with $\boldsymbol{\mathcal{R}}_m$. Both these variables will be jointly determined in a single optimization problem, presented in theorem 14.

Using (3.41), the posterior variance is given by

$$\boldsymbol{P}_k^+ = (\boldsymbol{I}_{N_x} - \mathcal{K}_k\mathcal{C}_m)\boldsymbol{P}_k^-(\boldsymbol{I}_{N_x} - \mathcal{K}_k\mathcal{C}_m)^T$$

$$+ \mathcal{K}_k\mathcal{R}_m\mathcal{K}_k^T. \quad (3.42)$$

Using (3.36a), the prior mean and variance of $\boldsymbol{\Gamma}_k$ at time $k+1$ is given by

$$\bar{\boldsymbol{\Gamma}}_{k+1}^- = \mathcal{A}_m\bar{\boldsymbol{\Gamma}}_k^+, \quad (3.43a)$$

$$\boldsymbol{P}_{k+1}^- = \mathcal{A}_m\boldsymbol{P}_k^+\mathcal{A}_m^T + \mathcal{B}_m\mathcal{Q}_m\mathcal{B}_m^T, \quad (3.43b)$$

which defines the random variable $\boldsymbol{\Gamma}_{k+1}^- \sim \mathcal{N}(\bar{\boldsymbol{\Gamma}}_{k+1}^-, \boldsymbol{P}_{k+1}^-)$.

Replacing $\boldsymbol{P}_k^-$ from (3.42) in (3.43b), we get the propagation equation for the prior variance

$$\boldsymbol{P}_{k+1}^- = \mathcal{A}_m(\boldsymbol{I}_{N_x} - \mathcal{K}_k\mathcal{C}_m)\boldsymbol{P}_k^-(\boldsymbol{I}_{N_x} - \mathcal{K}_k\mathcal{C}_m)^T\mathcal{A}_m^T$$

$$+ \mathcal{A}_m\mathcal{K}_k\mathcal{R}_m\mathcal{K}_k^T\mathcal{A}_m^T + \mathcal{B}_m\mathcal{Q}_m\mathcal{B}_m^T. \quad (3.44)$$

Steady-state variance $\boldsymbol{P}_\infty$ is determined by solving

$$\boldsymbol{P}_\infty = \mathcal{A}_m(\boldsymbol{I}_{N_x} - \mathcal{K}_\infty\mathcal{C}_m)\boldsymbol{P}_\infty(\boldsymbol{I}_{N_x} - \mathcal{K}_\infty\mathcal{C}_m)^T\mathcal{A}_m^T$$

$$+ \mathcal{A}_m\mathcal{K}_\infty\mathcal{R}_m\mathcal{K}_\infty^T\mathcal{A}_m^T + \mathcal{B}_m\mathcal{Q}_m\mathcal{B}_m^T, \quad (3.45)$$

where $\mathcal{K}_\infty$ is the steady-state gain. The steady-state variance of $\boldsymbol{x}_{(k+1)m}$ is then given by $\boldsymbol{\Sigma}_\infty := \boldsymbol{M}_x\boldsymbol{P}_\infty\boldsymbol{M}_x^T$.

Remark 4. Equations (3.44) becomes the Riccati difference equation (RDE) if $\mathcal{K}_k$ is

determined by minimizing the posterior variance. Consequently, it transforms (3.45) to the algebraic Riccati equation (ARE). That is, for $\boldsymbol{\mathcal{K}}_k$ given by (3.19), (3.44) transforms to

$$\boldsymbol{P}_{k+1} =$$

$$\boldsymbol{\mathcal{A}}_m(\boldsymbol{P}_k - \boldsymbol{P}_k\boldsymbol{\mathcal{C}}_m^T[\boldsymbol{\mathcal{C}}_m\boldsymbol{P}_k\boldsymbol{\mathcal{C}}_m^T + \boldsymbol{\mathcal{R}}_m]^{-1}\boldsymbol{\mathcal{C}}_m\boldsymbol{P}_k)\boldsymbol{\mathcal{A}}_m^T$$
$$+ \boldsymbol{\mathcal{B}}_m\boldsymbol{\mathcal{Q}}_m\boldsymbol{\mathcal{B}}_m^T, \quad (3.46)$$

and (3.45) transforms to

$$\boldsymbol{P}_\infty =$$

$$\boldsymbol{\mathcal{A}}_m(\boldsymbol{P}_\infty - \boldsymbol{P}_\infty\boldsymbol{\mathcal{C}}_m^T[\boldsymbol{\mathcal{C}}_m\boldsymbol{P}_\infty\boldsymbol{\mathcal{C}}_m^T + \boldsymbol{\mathcal{R}}_m]^{-1}\boldsymbol{\mathcal{C}}_m\boldsymbol{P}_\infty)\boldsymbol{\mathcal{A}}_m^T$$
$$+ \boldsymbol{\mathcal{B}}_m\boldsymbol{\mathcal{Q}}_m\boldsymbol{\mathcal{B}}_m^T, \quad (3.47)$$

Equation (3.47) has a unique positive semi-definite solution if $(\boldsymbol{\mathcal{A}}_m, \boldsymbol{\mathcal{R}}_m^{1/2}\boldsymbol{\mathcal{C}}_m)$ is detectable and $(\boldsymbol{\mathcal{A}}_m, (\boldsymbol{\mathcal{B}}_m\boldsymbol{\mathcal{Q}}_m\boldsymbol{\mathcal{B}}_m^T)^{1/2})$ is stabilizable.

3.4.3   Optimal Sensor Precision for Steady-State Estimation

Let $\boldsymbol{P}_\infty^d$ be the desired steady-state error, and let us assume that it is the solution of (3.47) for some $\boldsymbol{\mathcal{R}}_m^d$, i.e.

$$\boldsymbol{P}_\infty^d =$$

$$\boldsymbol{\mathcal{A}}_m(\boldsymbol{P}_\infty^d - \boldsymbol{P}_\infty^d\boldsymbol{\mathcal{C}}_m^T\left[\boldsymbol{\mathcal{C}}_m\boldsymbol{P}_\infty^d\boldsymbol{\mathcal{C}}_m^T + \boldsymbol{\mathcal{R}}_m^d\right]^{-1}\boldsymbol{\mathcal{C}}_m\boldsymbol{P}_\infty^d)\boldsymbol{\mathcal{A}}_m^T$$
$$+ \boldsymbol{\mathcal{B}}_m\boldsymbol{\mathcal{Q}}_m\boldsymbol{\mathcal{B}}_m^T.$$

Therefore, for any $\boldsymbol{\mathcal{R}}_m \leq \boldsymbol{\mathcal{R}}_m^d$,

$$\boldsymbol{P}_\infty^d \geq$$

$$\boldsymbol{\mathcal{A}}_m(\boldsymbol{P}_\infty^d - \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T [\boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T + \boldsymbol{\mathcal{R}}_m]^{-1} \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d) \boldsymbol{\mathcal{A}}_m^T$$

$$+ \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}_m^T, \quad (3.48)$$

which makes the solution of the RDE monotonic from $\boldsymbol{P}_\infty^d$.

According to Lemma 2 in [7], if for some k the solution of the RDE in (3.46) is monotonic, i.e. $\boldsymbol{P}_k \geq \boldsymbol{P}_{k+1}$, then $\boldsymbol{P}_{k+i} \geq \boldsymbol{P}_{k+i+1}$ for all $i \geq 1$. Therefore, (3.48), guarantees that the evolution of $\boldsymbol{P}_\infty^d$ is monotonic, and actual steady-state error $\boldsymbol{P}_\infty$ is guaranteed to satisfy $\boldsymbol{P}_\infty^d \geq \boldsymbol{P}_\infty$. The gap in the inequality can be minimized by maximizing $\mathbf{tr}\,[\boldsymbol{\mathcal{R}}_m]$. The optimization problem is presented as the following theorem.

Theorem 13. *Optimal sensor precision* $\boldsymbol{s}_k \in \mathbb{R}^{n_{y_k,m}} \geq 0$, *which satisfies* $\boldsymbol{\Sigma}_\infty \leq \boldsymbol{\Sigma}_\infty^d$ *is given by the solution of the following optimization problem,*

$$\left. \begin{array}{l} \displaystyle\min_{\boldsymbol{s}_m} \mathbf{tr}\,[\boldsymbol{W}\boldsymbol{\mathcal{S}}_m] \text{ subject to} \\[2ex] \begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_x \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T \\[1ex] (*)^T & \boldsymbol{L}_m + \boldsymbol{L}_m \boldsymbol{\mathcal{S}}_m \boldsymbol{L}_m \end{bmatrix} \geq 0, \\[3ex] 0 \leq \boldsymbol{s}_m \leq \boldsymbol{s}_{max}, \end{array} \right\} \quad (3.49)$$

where $\boldsymbol{\mathcal{S}}_m := \mathbf{diag}\,(\boldsymbol{s}_m)$, $\boldsymbol{\Sigma}_\infty^d$ *and* $\boldsymbol{P}_\infty^d$ *are given,*

$$\boldsymbol{M}_{11} := \boldsymbol{M}_x \Big(\boldsymbol{P}_\infty^d - \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{A}}_m^T - \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}_m^T$$

$$+ \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T \boldsymbol{L}_m^{-1} \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{A}}_m^T \Big) \boldsymbol{M}_x^T,$$

and $\boldsymbol{L}_m := \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T$. *The variable $\boldsymbol{W}$ is a diagonal matrix, which is user defined and serves as a normalizing weight on $\boldsymbol{\mathcal{S}}_m$.*

Proof. Pre and post multiplying (3.48) with $\boldsymbol{M}_x$ and $\boldsymbol{M}_x^T$ respectively, we get

$$\boldsymbol{\Sigma}_\infty^d \geq$$

$$\boldsymbol{M}_x \boldsymbol{\mathcal{A}}_m (\boldsymbol{P}_\infty^d - \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T [\boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T + \boldsymbol{\mathcal{R}}_m]^{-1}$$
$$\times \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d) \boldsymbol{\mathcal{A}}_m^T \boldsymbol{M}_x^T + \boldsymbol{M}_x \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}_m^T \boldsymbol{M}_x^T$$

Using matrix inversion lemma on $[\boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T + \boldsymbol{\mathcal{R}}_m]^{-1}$ we get

$$\boldsymbol{M}_x (\boldsymbol{P}_\infty^d - \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{A}}_m^T - \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}_m^T) \boldsymbol{M}_x^T$$
$$+ \boldsymbol{M}_x \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T \boldsymbol{L}_m^{-1} \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{A}}_m^T \boldsymbol{M}_x^T$$
$$- \boldsymbol{M}_x \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T \left(\boldsymbol{L}_m + \boldsymbol{L}_m \boldsymbol{\mathcal{R}}_m^{-1} \boldsymbol{L}_m \right)^{-1}$$
$$\times \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{A}}_m^T \boldsymbol{M}_x^T \geq 0,$$

where $\boldsymbol{L}_m := \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T$. Substituting $\boldsymbol{\mathcal{S}}_m := \boldsymbol{\mathcal{R}}_m^{-1}$, and using Schur complement we get the following linear matrix inequality,

$$\begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_x \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T \\ (*)^T & \boldsymbol{L}_m + \boldsymbol{L}_m \boldsymbol{\mathcal{S}}_m \boldsymbol{L}_m \end{bmatrix} \geq 0.$$

where

$$\boldsymbol{M}_{11} := \boldsymbol{M}_x \Big(\boldsymbol{P}_\infty^d - \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{A}}_m^T - \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}_m^T$$
$$+ \boldsymbol{\mathcal{A}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{C}}_m^T \boldsymbol{L}_m^{-1} \boldsymbol{\mathcal{C}}_m \boldsymbol{P}_\infty^d \boldsymbol{\mathcal{A}}_m^T \Big) \boldsymbol{M}_x^T.$$

Optimal precision is determined by minimizing $\mathbf{tr}\left[\boldsymbol{\mathcal{S}}_m\right]$. □

Theorem 13 can be used to determine the optimal precisions to bound the steady-state error, given $\boldsymbol{P}_\infty^d$ and $\boldsymbol{\Sigma}_\infty^d$. Like in theorem 11, providing these quantities can be challenging in some applications. For arbitrary, $\boldsymbol{P}_\infty^d$ and $\boldsymbol{\Sigma}_\infty^d$, the LMI in (3.49) can be infeasible. Also, since $\boldsymbol{\Sigma}_\infty^d = \boldsymbol{M}_x \boldsymbol{P}_\infty^d \boldsymbol{M}_x^T$, it is unclear how $\boldsymbol{M}_x \boldsymbol{P}_\infty^d \boldsymbol{M}_x^T$ can be constructed from $\boldsymbol{\Sigma}_\infty^d$, assuming we know a feasible $\boldsymbol{\Sigma}_\infty^d$. We circumvent this by treating $\boldsymbol{P}_\infty^d$ as a variable of optimization, and relaxing relax the accuracy constraint to $\mathbf{tr}\left[\boldsymbol{M}_x \boldsymbol{P}_\infty^d \boldsymbol{M}_x^T\right] \leq \gamma_d$, where γ_d is given. Introducing $\boldsymbol{P}_\infty^d$ as a variable makes the optimization problem in (3.49) nonconvex. We next present a convex formulation with $\boldsymbol{P}_\infty^d$ as variable.

Theorem 14. *Optimal sensor precision* $\boldsymbol{s} \in \mathbb{R}^{n_{y_{k,m}}} \geq 0$, *which satisfies* $\mathbf{tr}\left[\boldsymbol{M}_x \boldsymbol{P}_\infty^d \boldsymbol{M}_x^T\right] \leq \gamma_d$ *is given by the solution of the following optimization problem,*

$$\min_{\boldsymbol{s}_m, \boldsymbol{Z}, \boldsymbol{P}_\infty^d, \boldsymbol{\mathcal{K}}_\infty} \mathbf{tr}\left[\boldsymbol{W}\boldsymbol{\mathcal{S}}_m\right] \text{ subject to} \tag{3.50a}$$

$$\begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_x \boldsymbol{\mathcal{A}}_m (\boldsymbol{I}_{N_x} - \boldsymbol{\mathcal{K}}_\infty \boldsymbol{\mathcal{C}}_m) & \boldsymbol{M}_x \boldsymbol{\mathcal{A}}_m \boldsymbol{\mathcal{K}}_\infty \\ (*)^T & \boldsymbol{Z} & \boldsymbol{0}_{N_x \times n_{y_{k,m}}} \\ (*)^T & (*)^T & \boldsymbol{\mathcal{S}}_m \end{bmatrix} \geq 0, \tag{3.50b}$$

$$\begin{bmatrix} \boldsymbol{I}_{N_x} & \boldsymbol{P}_\infty^d & \boldsymbol{Z} \\ \boldsymbol{P}_\infty^d & \frac{1}{\delta}\boldsymbol{I}_{N_x} & \boldsymbol{0}_{N_x \times N_x} \\ \boldsymbol{Z} & \boldsymbol{0}_{N_x \times N_x} & \delta \boldsymbol{I}_{N_x} \end{bmatrix} \geq 0, \tag{3.50c}$$

$$\mathbf{tr}\left[\boldsymbol{M}_x \boldsymbol{P}_\infty^d \boldsymbol{M}_x^T\right] \leq \gamma_d, \tag{3.50d}$$

$$0 \leq \boldsymbol{s}_m \leq \boldsymbol{s}_{max}, \tag{3.50e}$$

$\boldsymbol{M}_{11} := \boldsymbol{M}_x \left(\boldsymbol{P}_\infty^d - \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}_m^T\right) \boldsymbol{M}_x^T$, $\boldsymbol{\mathcal{S}}_m := \mathbf{diag}\left(\boldsymbol{s}_m\right)$, $n_s := \sum_{j=1}^m p_j$, *and* p_j *is*

97

the dimension of the jth sensor. Variables $\gamma_d > 0$ and $\delta > 0$ are user specified. The variable $\boldsymbol{W}$ is a diagonal matrix, which is also user defined, and serves as a normalizing weight on $\boldsymbol{\mathcal{S}}_m$.

Proof. From (3.44), monotonicity of $\boldsymbol{P}^d_\infty$ is guaranteed if

$$\boldsymbol{P}^d_\infty \geq \boldsymbol{\mathcal{A}}_m(\boldsymbol{I}_{N_x} - \boldsymbol{\mathcal{K}}_\infty \boldsymbol{\mathcal{C}}_m)\boldsymbol{P}^d_\infty(\boldsymbol{I}_{N_x} - \boldsymbol{\mathcal{K}}_\infty \boldsymbol{\mathcal{C}}_m)^T \boldsymbol{\mathcal{A}}^T_m$$
$$+ \boldsymbol{\mathcal{A}}_m \boldsymbol{\mathcal{K}}_\infty \boldsymbol{\mathcal{R}}_m \boldsymbol{\mathcal{K}}^T_\infty \boldsymbol{\mathcal{A}}^T_m + \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}^T_m.$$

Introducing a new variable $\boldsymbol{Z} \in \mathbb{S}^{N_x}_+$, and the relaxation

$$\boldsymbol{Z}^{-1} \geq \boldsymbol{P}^d_\infty,$$

the condition for monotonicity can then be written as

$$\boldsymbol{P}^d_\infty \geq \boldsymbol{\mathcal{A}}_m(\boldsymbol{I}_{N_x} - \boldsymbol{\mathcal{K}}_\infty \boldsymbol{\mathcal{C}}_m)\boldsymbol{Z}^{-1}(\boldsymbol{I}_{N_x} - \boldsymbol{\mathcal{K}}_\infty \boldsymbol{\mathcal{C}}_m)^T \boldsymbol{\mathcal{A}}^T_m$$
$$+ \boldsymbol{\mathcal{A}}_m \boldsymbol{\mathcal{K}}_\infty \boldsymbol{\mathcal{R}}_m \boldsymbol{\mathcal{K}}^T_\infty \boldsymbol{\mathcal{A}}^T_m + \boldsymbol{\mathcal{B}}_m \boldsymbol{\mathcal{Q}}_m \boldsymbol{\mathcal{B}}^T_m.$$

However, we want to enforce monotonicity of $\boldsymbol{M}_x \boldsymbol{P}^d_\infty \boldsymbol{M}^T_x$, i.e.

$$\boldsymbol{M}_x\boldsymbol{P}_\infty^d\boldsymbol{M}_x^T \geq \boldsymbol{M}_x\boldsymbol{\mathcal{A}}_m(\boldsymbol{I}_{N_x} - \boldsymbol{M}_x\boldsymbol{\mathcal{K}}_\infty\boldsymbol{\mathcal{C}}_m)\boldsymbol{Z}^{-1}$$

$$\times (\boldsymbol{I}_{N_x} - \boldsymbol{M}_x\boldsymbol{\mathcal{K}}_\infty\boldsymbol{\mathcal{C}}_m)^T\boldsymbol{\mathcal{A}}_m^T\boldsymbol{M}_x^T$$

$$+ \boldsymbol{M}_x\boldsymbol{\mathcal{A}}_m\boldsymbol{\mathcal{K}}_\infty\boldsymbol{\mathcal{R}}_m\boldsymbol{\mathcal{K}}_\infty^T\boldsymbol{\mathcal{A}}_m^T\boldsymbol{M}_x^T + \boldsymbol{M}_x\boldsymbol{\mathcal{B}}_m\boldsymbol{\mathcal{Q}}_m\boldsymbol{\mathcal{B}}_m^T\boldsymbol{M}_x^T.$$

Using Schur complement, and substituting $\boldsymbol{\mathcal{S}}_m := \boldsymbol{\mathcal{R}}_m^{-1}$, we get

$$\begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_x\boldsymbol{\mathcal{A}}_m(\boldsymbol{I}_{N_x} - \boldsymbol{\mathcal{K}}_\infty\boldsymbol{\mathcal{C}}_m) & \boldsymbol{M}_x\boldsymbol{\mathcal{A}}_m\boldsymbol{\mathcal{K}}_\infty \\ (*)^T & \boldsymbol{Z} & \boldsymbol{0}_{N_x \times n_{y_{k,m}}} \\ (*)^T & (*)^T & \boldsymbol{\mathcal{S}}_m \end{bmatrix} \geq 0,$$

where $\boldsymbol{M}_{11} := \boldsymbol{M}_x\left(\boldsymbol{P}_\infty^d - \boldsymbol{\mathcal{B}}_m\boldsymbol{\mathcal{Q}}_m\boldsymbol{\mathcal{B}}_m^T\right)\boldsymbol{M}_x^T$.

The relaxation $\boldsymbol{Z}^{-1} \geq \boldsymbol{P}_\infty^d$ can be written as $\boldsymbol{P}_\infty^d\boldsymbol{Z} \leq \boldsymbol{I}_{N_x}$, which is non convex. However, we know that

$$\boldsymbol{P}_\infty^d\boldsymbol{Z} + \boldsymbol{Z}\boldsymbol{P}_\infty^d \leq \delta\boldsymbol{P}_\infty^d\boldsymbol{P}_\infty^d + \frac{1}{\delta}\boldsymbol{Z}\boldsymbol{Z},$$

for a given δ. Therefore,

$$\delta\boldsymbol{P}_\infty^d\boldsymbol{P}_\infty^d + \frac{1}{\delta}\boldsymbol{Z}\boldsymbol{Z} \leq \boldsymbol{I}_{N_x}, \tag{3.51}$$

guarantees $\boldsymbol{P}_\infty^d\boldsymbol{Z} \leq \boldsymbol{I}_{N_x}$. The inequality in (3.51), can be written as the following linear matrix inequality

$$\begin{bmatrix} \boldsymbol{I}_{N_x} & \boldsymbol{P}_\infty^d & \boldsymbol{Z} \\ \boldsymbol{P}_\infty^d & \frac{1}{\delta}\boldsymbol{I}_{N_x} & \boldsymbol{0}_{N_x \times N_x} \\ \boldsymbol{Z} & \boldsymbol{0}_{N_x \times N_x} & \delta\boldsymbol{I}_{N_x} \end{bmatrix} \geq 0.$$

The optimal precision is given by minimizing $\mathbf{tr}\left[\boldsymbol{\mathcal{S}}_m\right]$. □

Remark 5. The parameter δ can be tweaked, to improve the solution, using techniques from successive convex over-bounding techniques described in [82].

Remark 6. Like in theorem 11, and theorem 12, the sparseness of the solution can be improved by iteratively solving the optimization problem in theorems 13 and 14, with weights $\boldsymbol{W}_{j+1} := (\boldsymbol{\mathcal{S}}_m^*)_j^{-1}$, with $\boldsymbol{W}_1 := \boldsymbol{I}_{n_s}$, where subscript j denotes the iteration index.

Remark 7. The optimal $\boldsymbol{\mathcal{S}}_m^*$ can be conservative, since it guarantees $\boldsymbol{P}_\infty^d$ is contractive. The actual steady-state variance, denoted by $\boldsymbol{P}_\infty$, can be much smaller than $\boldsymbol{P}_\infty^d$. At the same time, it is possible that $\boldsymbol{P}_\infty$ is smaller than $\boldsymbol{P}_\infty^d$ without requiring $\boldsymbol{P}_\infty^d$ to be monotonic. Thus, theorems 13 and 14 are conservative and result in more precision than that required to achieve $\boldsymbol{P}_\infty^d$.

We remove the conservativeness by applying the following strategy. The optimization problems in theorems 13 and 14 are solved to determine the sparse solution $\boldsymbol{\mathcal{S}}_m^*$ for which $\boldsymbol{P}_\infty^d$ is contractive. It is then scaled by ξ to reduced the gap between $\boldsymbol{P}_\infty^d$ and $\boldsymbol{P}_\infty$, where optimal ξ^* is obtained using the bisection algorithm described in Algorithm 2.

3.5   Examples

Next, we apply the theory to two estimation problems related to aerospace engineering. The first example is related to the state estimation of an aircraft, and the second example is related to the tracking of a space object.

Algorithm 2 Bisection algorithm for optimal scaling of sensor precision.

Define: $\xi_{\min} := 0$
Define: $\xi_{\max} := 10^3$ # Something large
Solve optimization problem (3.49) to get $\boldsymbol{\mathcal{R}}_m^* := (\boldsymbol{\mathcal{S}}_m^*)^{-1}$.
Define: MAXITER = 100 # Something large
for i = 1:MAXITER
    $\xi = \frac{1}{2}(\xi_{\min} + \xi_{\max})$
    $\boldsymbol{P}_{ss}$:= solution of ARE in (3.45) with noise $\xi \boldsymbol{\mathcal{R}}_m^*$
    if **tr** $\left[\boldsymbol{M}_x \boldsymbol{P}_{ss} \boldsymbol{M}_x^T \right] < \gamma_d$
        $\xi_{\min} := \xi$
    else
        $\xi_{\max} := \xi$
    end
end

3.5.1   Flight Control Example

Here we demonstrate practical applications of the result presented in theorem 14. It is applied to a steady-state estimation problem for an aircraft model. We first present the details of the aircraft model. We then present two examples, which highlight different applications of theorem 14.

3.5.1.1   Model

Let us consider the longitudinal motion model of an aircraft, where the states of the system are as velocity $V(ft/s)$, angle of attack $\alpha(rad)$, pitch angle $\theta(rad)$, and pitch

rate $q(rad/s)$, i.e.

$$\boldsymbol{x} := \begin{bmatrix} V \\ \alpha \\ \theta \\ q \end{bmatrix}. \tag{3.52}$$

We consider onboard sensors that measure body acceleration $\dot{u}\,(ft/s^2)$ along roll axis, body acceleration $\dot{w}\,(ft/s^2)$ along yaw axis, angle of attack $\alpha(rad)$, pitch rate $q(rad/s)$, and dynamic pressure $\bar{q} := \frac{1}{2}\rho V^2\,(lb/ft^2)$, where ρ is the atmospheric density. Variables u and w are defined as $u := V\cos(\alpha)$, and $w := V\sin(\alpha)$.

Therefore, the vector of measured outputs is

$$\boldsymbol{y} := [\dot{u}, \dot{w}, \alpha, q, \bar{q}]^T. \tag{3.53}$$

In aircraft, these measurements are available from accelerometers, angle-of-attack sensors, gyro sensors, and pitot tube, respectively.

The dynamics and measurement model is given by the following equations

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}d, \tag{3.54a}$$

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}d + \boldsymbol{n}, \tag{3.54b}$$

102

where *

$$
A = \begin{bmatrix} -0.0179 & 33.2244 & -32.1700 & 0.6728 \\ -0.0001 & -1.4528 & 0 & 0.9323 \\ 0 & 0 & 0 & 1.0000 \\ -0.0000 & -4.1970 & 0 & -1.8836 \end{bmatrix},
$$

$$
B = \begin{bmatrix} 0.5697 & -0.0029 & 0 & -0.4670 \end{bmatrix}^T,
$$

$$
C = 10^3 \times \begin{bmatrix} -0.0000 & 0.0332 & -0.0322 & 0.0007 \\ -0.0001 & -1.3544 & 0 & 0.8692 \\ 0 & 0.0010 & 0 & 0 \\ 0 & 0 & 0 & 0.0010 \\ 0.0017 & 0 & 0 & 0 \end{bmatrix},
$$

$$
D = \begin{bmatrix} 0.5697 & -2.7345 & 0 & 0 & 0 \end{bmatrix}^T,
$$

d is the disturbance acting on the system, and n is the sensor noise. Note that there is a direct feed-through term because the disturbance directly impacts the acceleration measurements. In this example, we model the disturbance as a filtered white noise, filtered by $\frac{1}{s/\omega_c+1}$, where ω_c is the cutoff frequency. The disturbance is generated by a vibrating control surface in the aircraft.

Let the filter state be x_d, and the filter dynamics is given by

$$
\dot{x}_d = \omega_c(-x_d + w), \text{ and } d = x_d,
$$

*More accurate data is available upon request.

where w is white noise with a given variance. In this example, we choose the variance to be 5 deg^2. This corresponds to small angular deflections in the control surface, which is measured in degrees.

3.5.1.2 Optimal Sensor Precision for Standard Discrete-Time Kalman Filtering

Here we discretize the continuous time system with sampling time $\Delta t := 0.01$ seconds and determine the least precision needed to achieve a steady-error that satisfies $\mathbf{tr}\left[\boldsymbol{M}_x \boldsymbol{P}_\infty \boldsymbol{M}_x^T\right] \leq \gamma_d$ for $\gamma_d := 0.1$, where $\boldsymbol{P}_\infty$ quantifies the actual steady-state error. In this example, theorem 14 is applied with $m = 1$. For the filter, the cutoff frequency ω_c is chosen to be 10 rad/s. Finally, $\delta = 200$ was chosen to implement constraint in (3.50c).

Fig.(3.2) shows the sensor precisions from the unweighted optimization (indicated by legend "1"), the precisions from iteratively weighted optimization to improve sparseness (indicated by legend "1/$\mathbf{s}$"), and finally, the scaled precision to remove the conservativeness in the optimal solution (indicated by legend "$\mathbf{s}/\xi^*$"). The top-panel in fig.(3.2) shows the sparse solution, and the bottom panel shows the same data on the logarithmic scale. We observe that out of the five sensors chosen in the design, only two significantly contribute to the required estimation accuracy.

Iteratively weighted optimization significantly improves the sparseness in the solution by several orders of magnitude. We also observe the solution of the weighted optimization is conservative and the precisions can be further reduced to get closer to the boundary of $\mathbf{tr}\left[\boldsymbol{M}_x \boldsymbol{P}_\infty \boldsymbol{M}_x^T\right] \leq \gamma_d$.

Fig.(3.3) shows the scaled optimal precisions for $\xi^* = 64.1106$. The precision values in fig.(3.3) indicate that only angular velocity measurement q and dynamic pressure

data $\bar{q}$ are needed in higher precision, to estimate all four states of the system with the required accuracy.
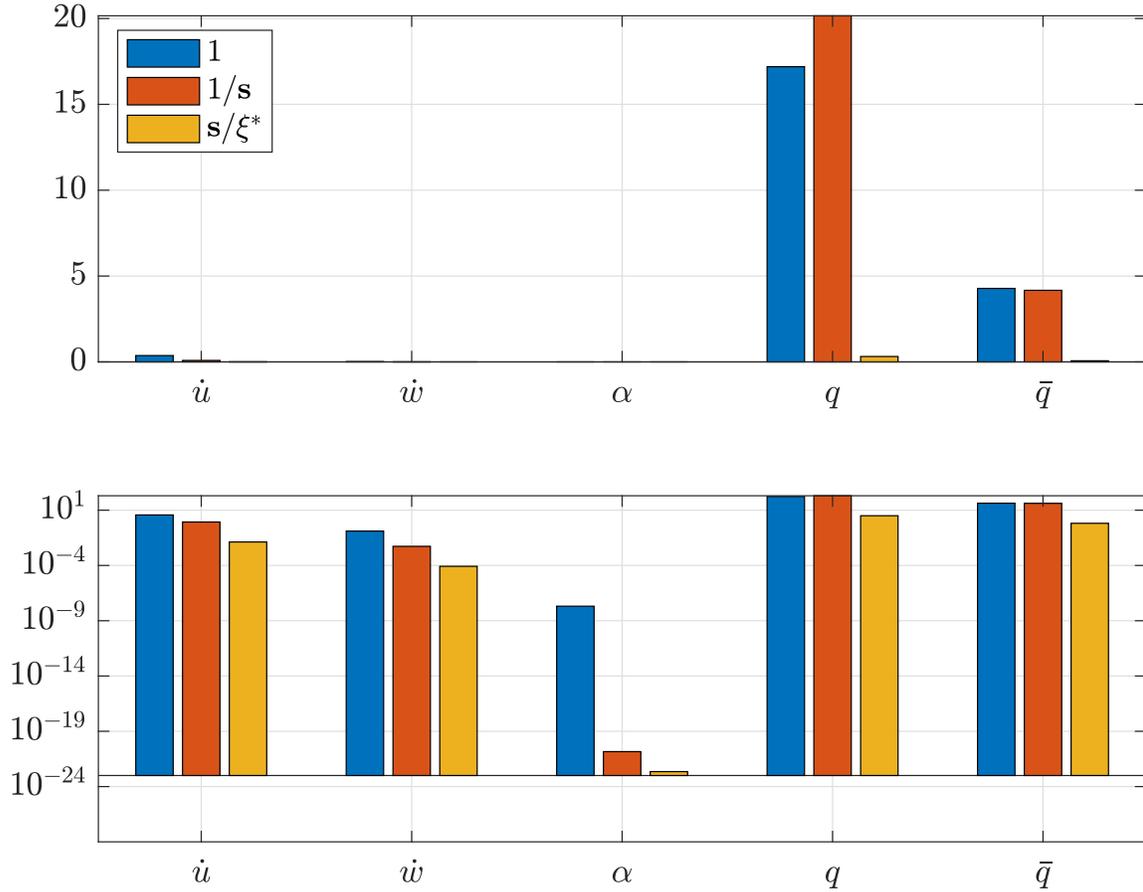
Figure 3.2: Sensor precisions for the five sensors from different algorithms. In the legend, 1 indicates solution from unweighted optimization, $1/\mathbf{s}$ indicates solution from iteratively weighted optimization to improve sparseness, and $\mathbf{s}/\xi$ indicates scaled optimal solution using algorithm 2. The bottom-panel shows the same data as the top-panel, but in logarithmic scale.

To completely remove the sensor for $\dot{u}, \dot{v}$, and α, we can set their corresponding precisions to exactly zero, prior to ξ scaling. Since detectability and stabilizability conditions are satisfied for this sensor configuration, (3.45) has a unique solution.
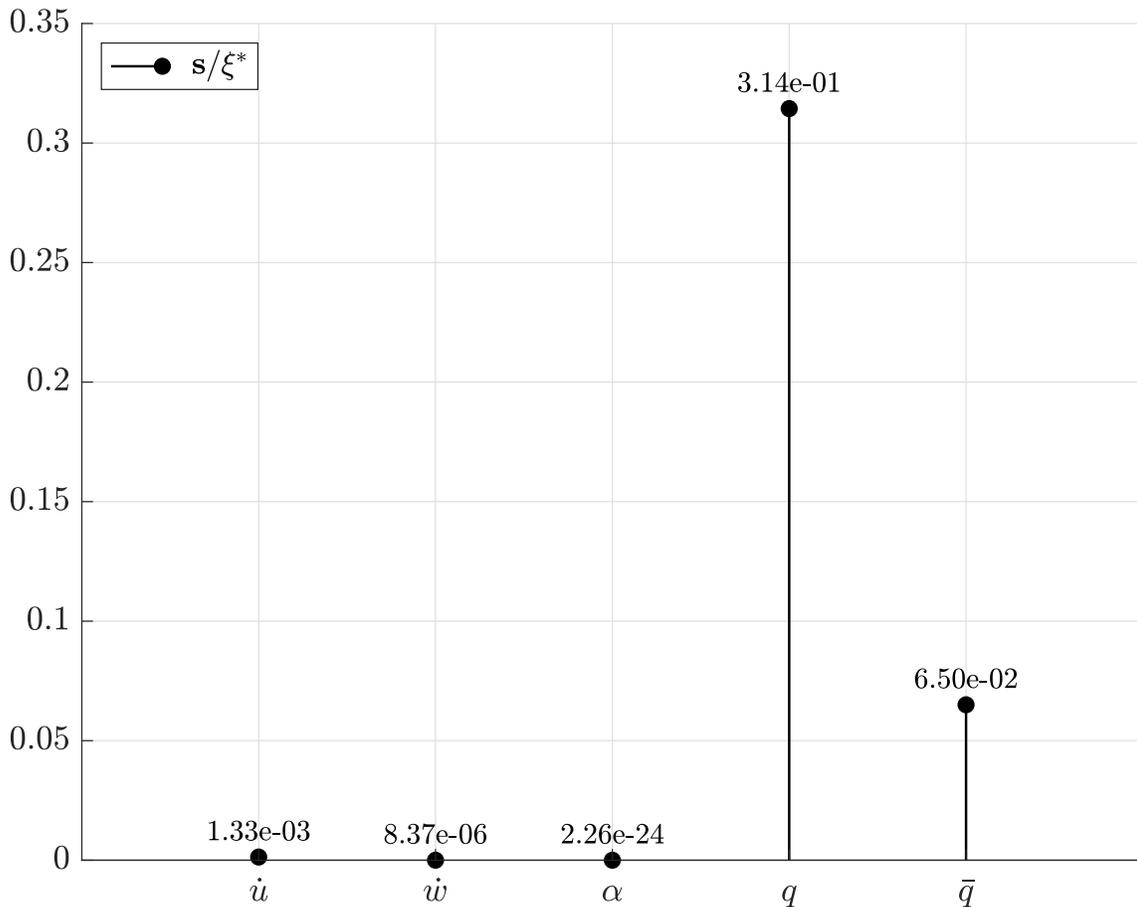
Figure 3.3: Optimal scaled sensor precisions satisfying $\mathbf{tr}\left[\boldsymbol{M}_x\boldsymbol{P}_\infty\boldsymbol{M}_x^T\right] \leq \gamma_d$ for $\gamma_d := 0.1$.

Therefore, using algorithm 2 we can determine the optimal ξ scaling that guarantees $\mathbf{tr}\left[\boldsymbol{M}_x\boldsymbol{P}_\infty\boldsymbol{M}_x^T\right] \leq \gamma_d$. For this example, we get $\xi^* = 3.375$, and the scaled sensor precisions are shown in fig.(3.4). We see from fig.(3.4) that the required precisions for q and $\bar{q}$ in this case are much higher than those in fig.(3.3).

It is noteworthy, that without the sensors for $\dot{u}, \dot{v}$, and α, optimization problem in theorem 14 is infeasible. The infeasibility also occurs without the constraint $\mathbf{tr}\left[\boldsymbol{M}_x\boldsymbol{P}_\infty\boldsymbol{M}_x^T\right] \leq \gamma_d$. Therefore, for this sensor configuration, there does not exist any $\boldsymbol{P}_\infty^d \in \mathbb{S}_+^{N_x}$ that results in monotonic solutions of (3.44), even though detectability

106
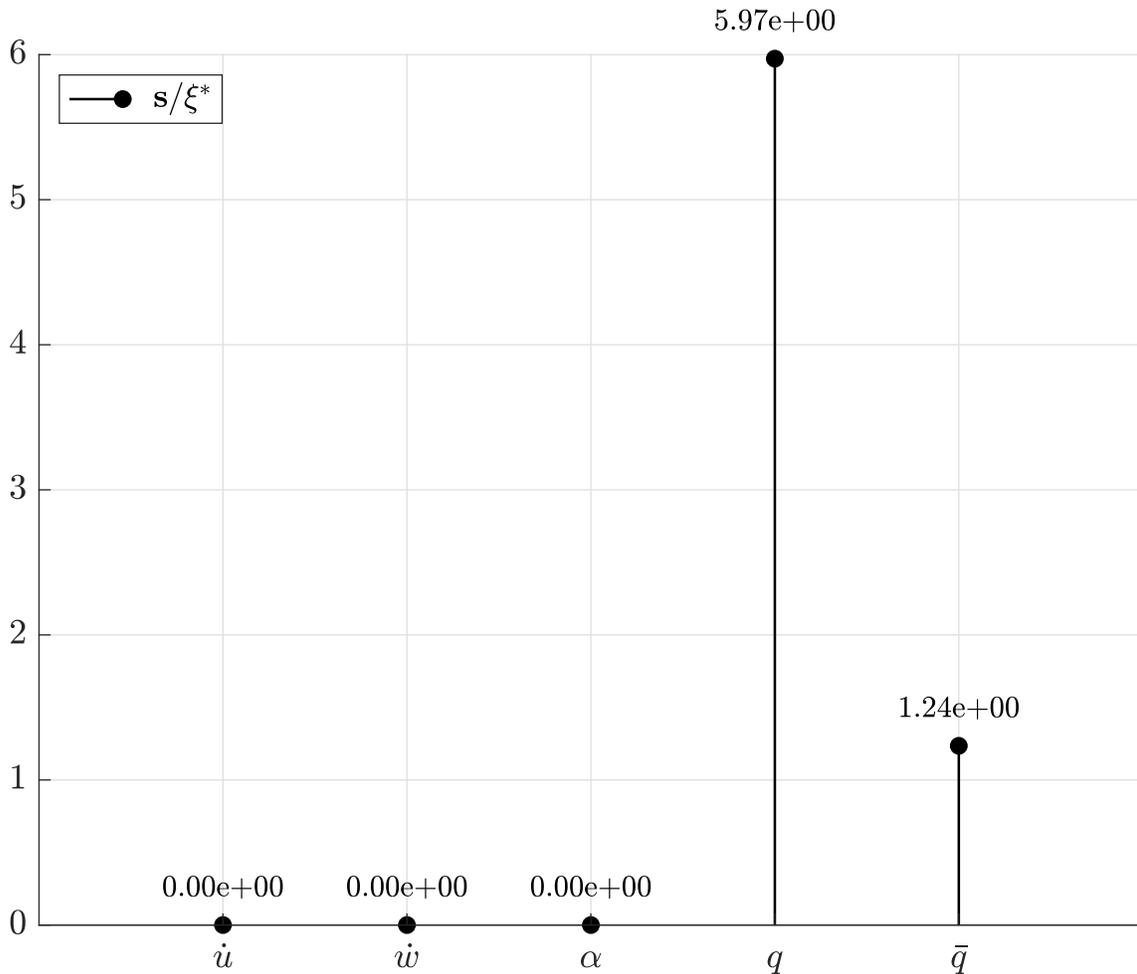
Figure 3.4: State estimation with only q and $\bar{q}$: required precisions to achieve $\mathbf{tr}\left[\boldsymbol{M}_x\boldsymbol{P}_\infty\boldsymbol{M}_x^T\right] \le \gamma_d$ for $\gamma_d := 0.1$.

and stabilizability conditions are satisfied. This highlights the conservativeness of the results, as mentioned in remark 7.

Therefore, from a sensor pruning perspective, an ad-hoc approach would be to start with a dictionary of sensors and determine the optimal sensor precisions using theorem 14, then assign zero precisions to those sensors with small precisions, and finally apply algorithm 2 to arrive at the optimal precisions of the reduced number of sensors. More sophisticated algorithms [21, 51, 78] for sensor pruning or sensor selection

107

can also be applied. These algorithms assume sensor precisions are known, which can be determined from the optimization problems presented in this paper.

3.5.1.3 Tradeoff Between Sensing-Rate & Sensor Precision

In this section, we apply theorem 14 to explore the tradeoff between sensing rate and sensing precision. We use the same F16 example described in §3.5.1.1. In this example, the continuous-time model is discretized with $dt = 1/1000$, using Tustin's method. The augmented system is created with $q = 10$. This formulation captures a scenario where the sensor data is available at 1 Khz, but the state estimates are needed at 100 Hz. We assume that the state's estimates are used by some control law executing at 100 Hz.

In this example, the 5 physical sensors in (3.53) are treated as virtual sensors over 10 time steps and are assigned an unknown precision. Thus, there are 50 virtual sensors. Optimization in (3.50), results in the sparse precisions shown in fig.(3.5). The y-axis are the five physical sensors, and the x-axis are times steps from 1 to q. The heat-map shows the precisions of the 5 sensors across the 10 time steps. The three panels in fig.(3.5) are solutions with $s_{\max} = 5, 2.5, 1$ respectively, and scaled by ξ^* determined by Algorithm 2. They have the same required accuracy, defined by $\gamma_d = 0.1$. In the top panel, we see that the required precisions for sensors with 1000 Hz are much lower than the results shown in fig.(3.4), which is for 100 Hz sensing. Therefore, less precise data at a higher rate can achieve the same accuracy. It is also interesting to note that the sensors with nonzero precision are different in the two cases. In the 100 Hz example, the angular velocity sensor is the most precise, followed by the dynamic pressure sensor. Other sensors have very low precision. In the 1000 Hz example, the angular velocity sensor and angle of attack sensors have very low

precisions, but acceleration measurements have relatively higher precisions, with the dynamic pressure sensor the most precise. We also observe that the dynamic pressure sensor plays an important role in both the cases and the angle-of-attack sensor has very low precision in both cases.

It is also interesting to note that the sensor values at the beginning and the end of the 10 time-step window have higher precision. However, as $s_{\max}$ is reduced, we observe that intermediate values of $\bar{q}$ are needed to achieve the same accuracy. We can infer from this observation, that if sensor precision is poor, we can achieve a higher estimation accuracy by fusing data at a higher rate. Theorem 14, reveals data from which sensors are needed at a higher rate, and the corresponding precisions, to achieve this accuracy. From a real-time scheduling perspective, this is very useful because from fig.(3.5) we can determine exactly when to poll the sensors. This optimizes sensor polling, reduces the associated delays, and improves real-time schedulability.

3.5.2   Satellite Tracking Problem

Here we consider the problem of determining the optimal sensor precision for tracking a space object with the required accuracy. We first describe the model for the satellite dynamics and then present the result from the optimization problem in theorem 12.

3.5.2.1   Model

Here we consider a simple satellite dynamics model given by

$$\ddot{r} = -\frac{\mu_E}{r^2} + \dot{\theta}^2 r + \frac{3J_2}{2r^4}\left(3\sin(\theta)^2 - 1\right), \tag{3.55a}$$

$$\ddot{\theta} = -\frac{2\dot{\theta}\dot{r}}{r} - \frac{3J_2}{r^4}\cos(\theta)\sin(\theta). \tag{3.55b}$$

Length and time in the dynamics are normalized using R_E (radius of Earth), and T_p (time for one orbit) respectively. Fig.(3.6) shows the trajectory of the satellite, with normalized initial condition

$$r_0 = \frac{R_E + h}{R_E} = 1.0533, \tag{3.56a}$$

$$\dot{r}_0 = 0, \tag{3.56b}$$

$$\theta_0 = 0, \tag{3.56c}$$

$$\dot{\theta}_0 = \left(\frac{V_\theta T_p}{R_E}\right)\frac{1}{r_0} = 6.2832. \tag{3.56d}$$

The parameters necessary to simulate the system are provided in table 3.1.

$R_E = 6378.1363$ km	$\mu_E = 398600.4415$ km^3/s^2
$T_p = 5.48 \times 10^3$ s	$J_2 = 1.7555 \times 10^{10}$ km^5/s^2
$V_\theta = 7.7027$ km/s	$h = 340$ km

Table 3.1: Parameters in the satellite dynamics model.

Equation (3.55) is linearized about a nominal trajectory to obtain a continuous-time periodic system. We augment the linear model with process noise, to account for the effects of sporadic thrusts that are necessary for orbital station keeping. The augmented model is given by,

$$\dot{\boldsymbol{x}} = \boldsymbol{A}(t)\boldsymbol{x} + \boldsymbol{B}\boldsymbol{w}(t), \tag{3.57}$$

where $\boldsymbol{x} := \begin{bmatrix} r & \dot{r} & \theta & \dot{\theta} \end{bmatrix}^T$ is the state vector, $\boldsymbol{w}(t) := \begin{bmatrix} w_r(t) & w_\theta(t) \end{bmatrix}^T$ is a zero-mean

Gaussian random process,

$$\boldsymbol{A}(t) := \begin{bmatrix} 0 & 1.0 & 0 & 0 \\ a_{21}(t) & 0 & a_{23}(t) & 12.59 \\ 0 & 0 & 0 & 1.0 \\ a_{41}(t) & -12.21 & a_{43}(t) & 0 \end{bmatrix}, \tag{3.58a}$$

$$\boldsymbol{B} := \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \tag{3.58b}$$

with

$$a_{21}(t) = 0.416\cos(12.4t) + 126.4,$$

$$a_{23}(t) = 0.2113\sin(12.4t),$$

$$a_{41}(t) = 0.2774\sin(12.4t),$$

$$a_{43}(t) = -0.1408\cos(12.4t).$$

In this example, we assume the mass of the satellite is 100 kg, and the satellite sporadically applies maximum of 1mN of thrust for orbital station keeping. The normalized accelerations due to these thrusts are modeled as zero-mean Gaussian random processes $w_r(t)$ and $w_\theta(t)$, with $\mathbb{E}\left[w_r w_r^T\right] = \mathbb{E}\left[w_\theta w_\theta^T\right] = 0.0471^2$. Fig.(3.7) shows the propagation of mean $\boldsymbol{\mu}(t)$ and variance $\boldsymbol{\Sigma}(t)$ for the time-varying linear

111

system, with

$$\boldsymbol{\mu}(t_0) := \begin{bmatrix} 50/R_E \\ 0 \\ 0 \\ 0 \end{bmatrix}, \text{ and} \tag{3.59a}$$

$$\boldsymbol{\Sigma}(t_0) := 0.01 \times \mathbf{diag}\left(\boldsymbol{\mu}(t_0)\right). \tag{3.59b}$$

The evolution equation for $\boldsymbol{\mu}(t)$ and $\boldsymbol{\Sigma}(t)$ are given by

$$\dot{\boldsymbol{\mu}}(t) = \boldsymbol{A}(t)\boldsymbol{\mu}(t), \tag{3.60}$$

$$\dot{\boldsymbol{\Sigma}}(t) = \boldsymbol{A}(t)\boldsymbol{\Sigma}(t) + \boldsymbol{\Sigma}(t)\boldsymbol{A}^T(t) + \boldsymbol{B}\boldsymbol{Q}\boldsymbol{B}^T, \tag{3.61}$$

where $\boldsymbol{Q} := 0.0471^2 \times \boldsymbol{I}_2$.

We discretize the normalized time interval $[0, 1]$ with $dt = 0.1$, resulting in the temporal grid $\{t_k\}$, where $t_k := kdt$. We assume that measurements are available at these times. The dynamics in (3.57) is discretized over $\{t_k\}$, and is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_d(t_k)\boldsymbol{x}_k + \boldsymbol{w}_k, \tag{3.62}$$

where

$$\boldsymbol{A}_d(t_k) := \boldsymbol{\Phi}(t_{k+1}, t_k), \tag{3.63}$$

$$\boldsymbol{w}_k := \int_{t_k}^{t_{k+1}} \boldsymbol{\Phi}(\tau, t_k)\boldsymbol{B}\boldsymbol{w}(\tau)d\tau, \tag{3.64}$$

and $\boldsymbol{\Phi}(\cdot, \cdot)$ is the state-transition matrix, which is obtained by numerical integration of the fundamental matrix.

It is easy to verify that if $\mathbb{E}\left[\boldsymbol{w}(t)\right] = 0$, then $\mathbb{E}\left[\boldsymbol{w}_k\right] = 0$. Therefore, $\boldsymbol{w}_k$ is a zero-mean random process. To determine the optimal precision, we need to quantify $\boldsymbol{Q}_k :=$ $\mathbb{E}\left[\boldsymbol{w}_k\boldsymbol{w}_k^T\right]$, which is difficult to determine from (3.64). Instead, we use the covariance $\boldsymbol{\Sigma}(t)$, determined by solving (3.61), and the discrete-time covariance propagation equation, to determine the time-varying $\boldsymbol{Q}_k$. It is given by

$$\boldsymbol{Q}_k := \boldsymbol{\Sigma}(t_{k+1}) - \boldsymbol{A}_d(t_k)\boldsymbol{\Sigma}(t_k)\boldsymbol{A}_d^T(t_k). \tag{3.65}$$

3.5.2.2   Optimal Precision for One-Time Step Update

In this example, we consider a set of 10 laser-ranging sensors located on the surface of the Earth, at angular positions $\theta(t_k)$. For the periodic system described above, the objective is to determine the optimal sensor precisions such that $\mathbf{tr}\left[\boldsymbol{\Sigma}^+(t_k = 1)\right] \leq \gamma_d$, given $\boldsymbol{\Sigma}^-(t_k = 0)$. Here we apply theorem 12 to determine the optimal sensor precisions, which are shown in fig.(3.8) for various values of $\boldsymbol{s}_{\max}$.

The optimization is done with $\boldsymbol{\Sigma}(t_k = 0) = \boldsymbol{\Sigma}(t_0)$, and $\gamma_d = 0.1 \times \mathbf{tr}\left[\boldsymbol{\Sigma}^-(t_k = 1)\right]$, where $\boldsymbol{\Sigma}^-(t_k = 1)$ is the prior obtained at $t_k = 1$. It is obtained by propagating $\boldsymbol{\Sigma}(t_k = 0)$ using (3.61). Variance $\boldsymbol{\Sigma}(t_0)$ is defined in (3.59b). The value of γ_d specifies that we want the trace of the posterior to be 10% of the prior at $t_k = 1$.

From fig.(3.8), we see that there is a tradeoff between sensor precision and sensing frequency. With low precision sensors ($\boldsymbol{s}_{\max} = 819.60$), we need to sense the satellite at all the sites to guarantee $\mathbf{tr}\left[\boldsymbol{\Sigma}^+(t_k = 1)\right] < \gamma_d$. As $\boldsymbol{s}_{\max}$ is increased, the

sensing becomes more sparse. With $s_{\max} = 2500$, we only need to get two range measurements, in order to estimate the state at $t_k = 1$ with the required accuracy.

This is significant for large-scale spatio-temporal sensing, and especially for tracking space objects. Currently, there are about $500K$ space objects, and only $30K$ are tracked. The number of sensing sites is significantly lower than that. Therefore, by applying theorem 12, it is possible to determine the sparse sensing schedule for each object, thereby increasing the ability to track more objects with guaranteed accuracy.

Precision in laser-ranging sensors is determined by the energy of the laser beam and the reflectivity of the object being sensed. For a given object, a lower value of precision implies lower energy requirements, which results in optimal sensor designs. Modern satellites have reflectors, which allows precise sensing with low powered lasers. Whereas, some space objects (e.g. asteroids, etc) have poor reflectivity, which fundamentally limits how accurately it can be sensed. This limit can be incorporated using the variable $s_{\max}$.

3.6   Summary & Conclusion

In this paper, we presented convex optimization problem-formulations to determine optimal sensor precisions that guarantee a specified estimation accuracy. The formulation is presented in a general multi-rate sensing framework, with linear time-varying discrete-time system dynamics. Optimality is achieved by minimizing sensor precisions, subject to upper bound on the estimation error, as defined in the discrete-time Kalman filtering framework. Since the minimization of precisions is done with respect to the l_1 norm, the proposed optimization framework can also be used to determine

sparse sensing architectures. This will be valuable in the design of large-scale sensor networks. We have shown the engineering value of the proposed theory by applying it to realistic flight mechanics and astrodynamics problems.

3.7   Acknowledgements
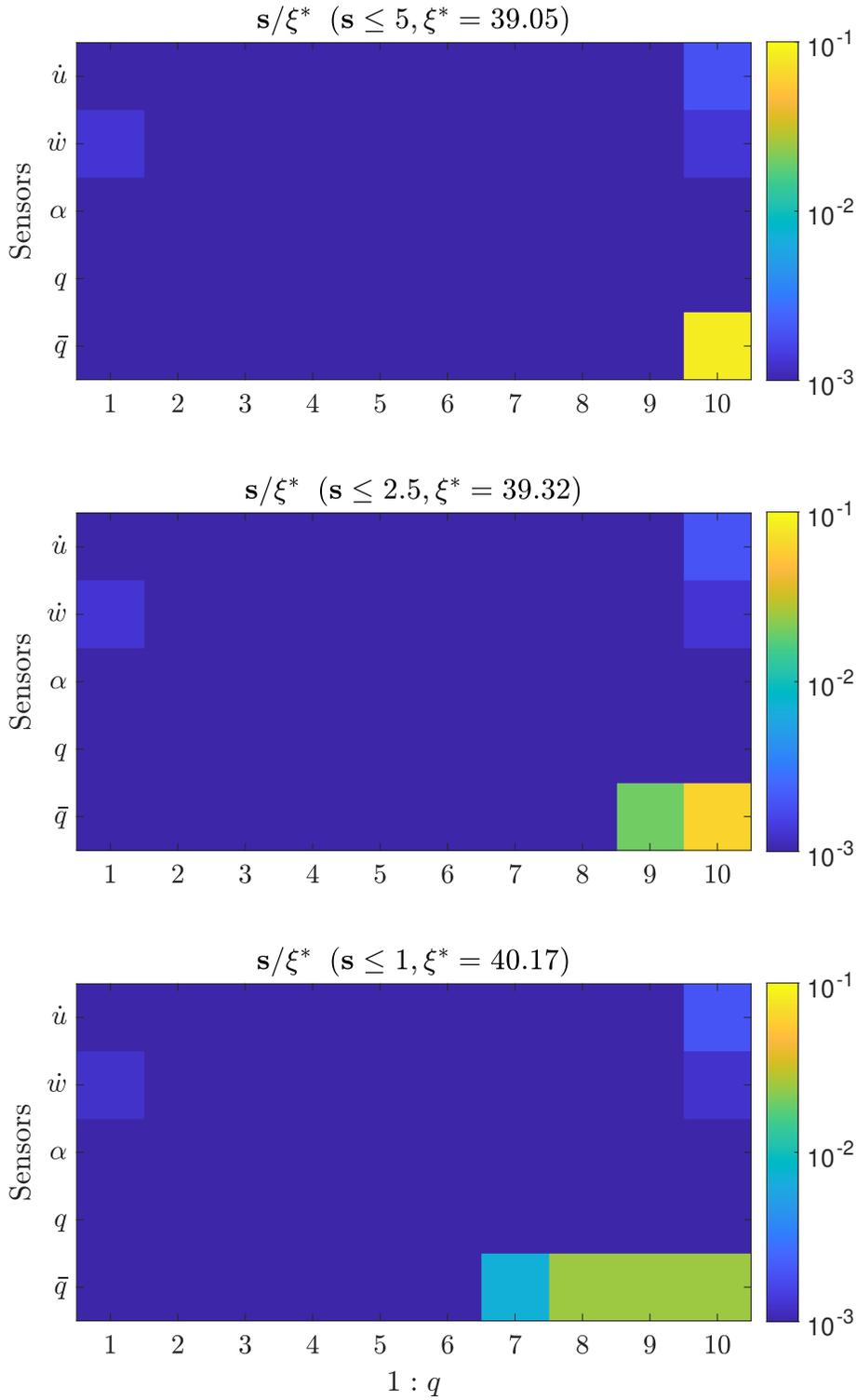
Figure 3.5: Optimal scaled precisions for high sensing rate, for $\gamma_d := 0.1$ guaranteeing $\mathbf{tr}\left[\boldsymbol{M}_x\boldsymbol{P}_\infty\boldsymbol{M}_x^T\right] \leq \gamma_d$.
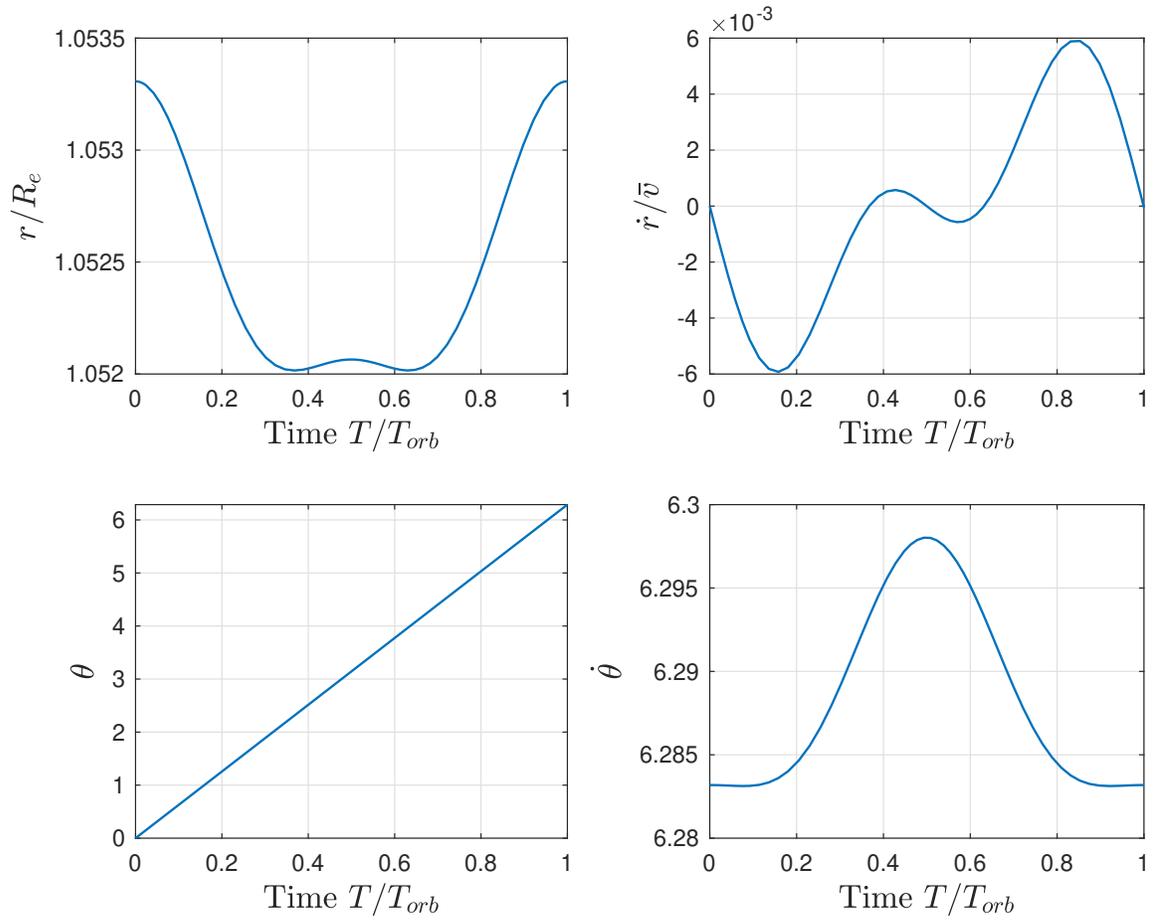
Figure 3.6: Reference trajectories (normalized) about which (3.55) has been linearized.
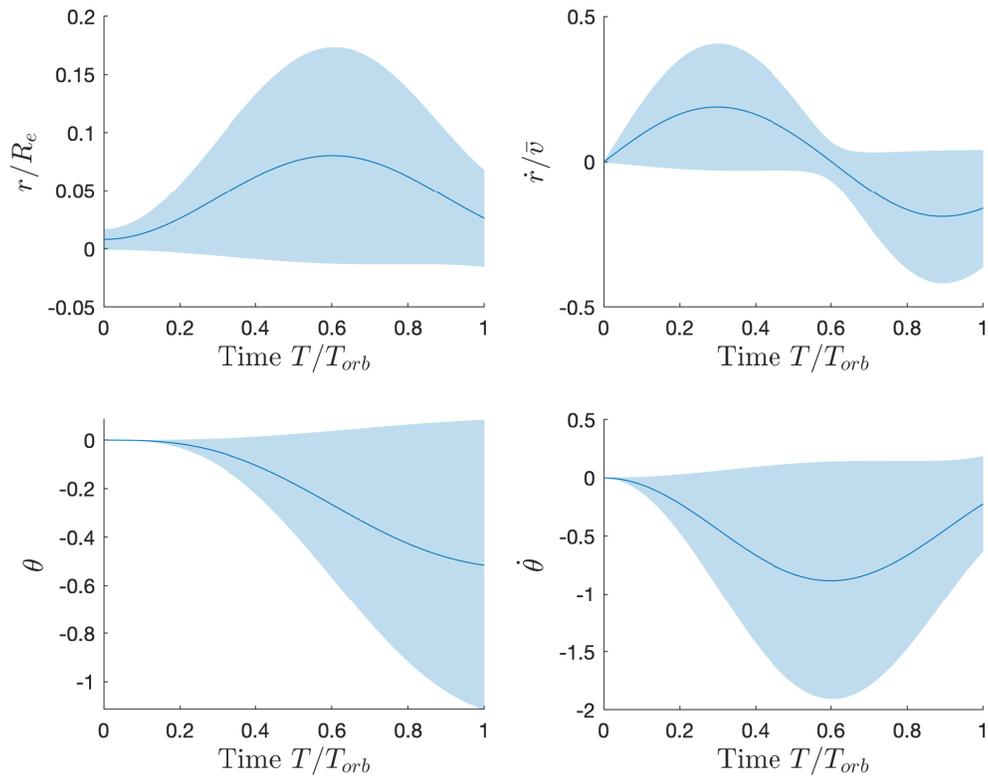
Figure 3.7: Uncertainty propagation with the linear time-varying periodic system. Solid line shows the evolution of the mean perturbation, and the shaded region shows $\boldsymbol{\mu}_i \pm \sqrt{\boldsymbol{\Sigma}_{ii}}$ for $i = 1, 2, 3$ and 4.
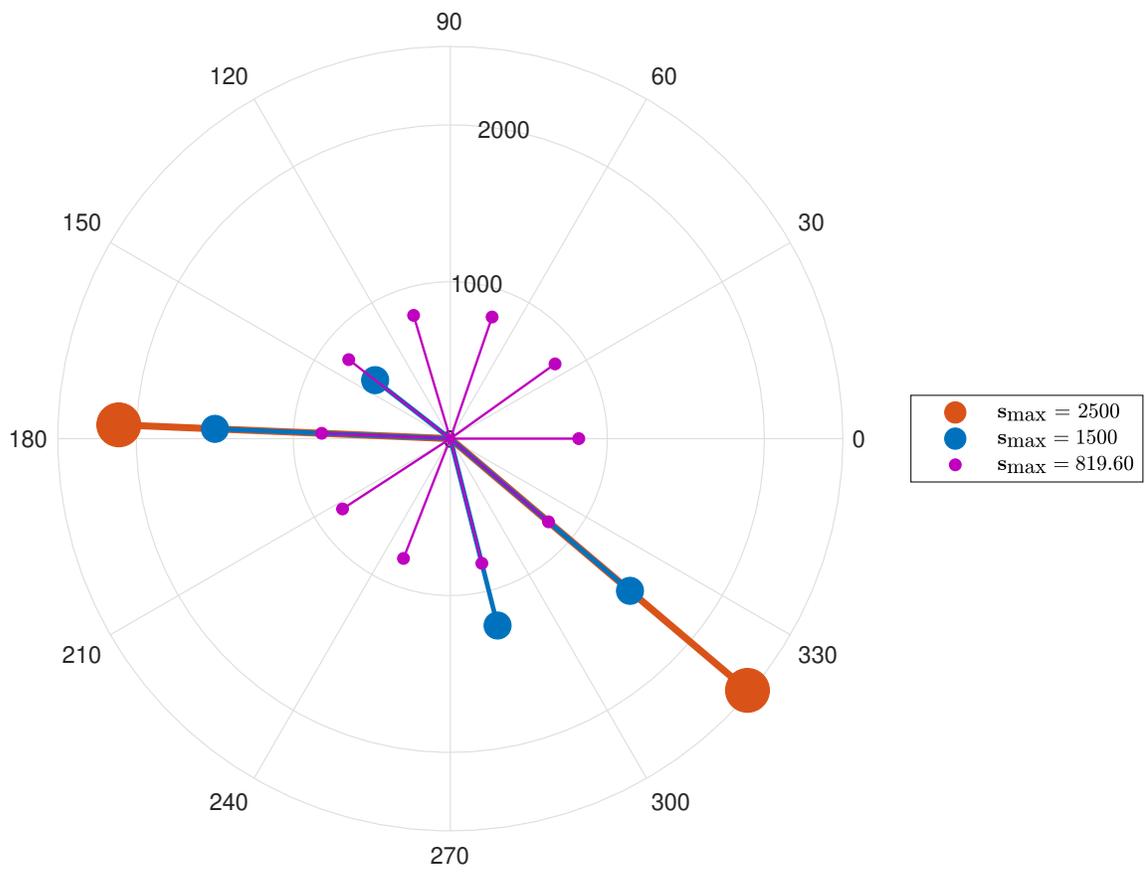
Figure 3.8: Optimal precisions for $\boldsymbol{s}_{\max} = 2500, 1500, 819.60$.

4. PRIVACY AND UTILITY AWARE DATA SHARING FOR SPACE SITUATIONAL AWARENESS FROM ENSEMBLE AND UNSCENTED KALMAN FILTERING PERSPECTIVE*

4.1   Introduction

As space becomes more congested, maintaining a timely and accurate picture of space activities simultaneously becomes both more important and difficult. The seriousness of this problem has been highlighted by the 2009 collision between U.S. Iridium LLC and Russian Federation Cosmos satellites, which destroyed both the satellites and added more than 2,000 additional pieces of debris in space. Prior to that, in 2007, China used a hit-to-kill interceptor to destroy their Fengyun-1C satellite. Since then, U.S. Space Surveillance Network has cataloged more than 2,200 trackable debris fragments larger than 10 centimeters originating from this collision [84]. It cannot be emphasized enough that as the number of assets orbiting Earth increases the danger and effects of collisions also increases. The new Starlink Mission of SpaceX is planned to have 12,000 satellites [29]. With the increased number of space objects, we can no longer assume that the space is *big* and collisions between space objects will rarely occur.

In fact, close approaches and even collisions occur with increasing regularity, with at least a dozen collisions having occurred in LEO. There are also strong indicators of at least five collisions in GEO [57, 1, 2, 43, 59] since the dawn of the space age.

There could easily have been more collisions in both LEO and GEO that have not been publicly disclosed [63]. More recently on September 2 in 2019, the European Space Agency (ESA) maneuvered one of its Earth science satellites Aeolus, to avoid a potential catastrophic collision with a SpaceX Starlink satellite [30]. The maneuver took place just about half an orbit before closest approach, indicating the importance of maintaining situational awareness in space.

Space situational awareness (SSA) refers to the ability to view, understand and predict the physical location of natural and manmade objects in orbit around the Earth, with the objective of avoiding collisions. To address the growing SSA challenge, the U.S. and other nations, along with commercial operators, have established a new approach to exchange information regarding space objects (USSTRATCOM & Space Data Association), hoping to increase the safety of satellite operations. While there are significant benefits to sharing of data from the SSA perspective, there are several privacy/security related concerns from both commercial and military perspectives, which lead to conservative data-sharing policy. For example, the policy for sharing SSA data from military owned sensors, unveiled by U.S. Strategic Command in 2014, has led to removal of more SSA data from public access. This includes removal of data on the estimated size of space objects in the public satellite catalog, and limitations on what data is provided privately to satellite operators. Many of these restrictions stem from the desire to hide national utility satellites and their activities [83]. This has led to some operators question the accuracy, and especially the completeness, of the information provided to them by the US Department of Defense (DoD). For example, some South Korean government officials estimate that their country receives data on only about 40 percent of the objects tracked by the DoD, due to sensitivity of U.S. assets [48]. This has resulted in lack of confidence in

the DoD-provided data, especially in the commercial sector. This lack of confidence in the data also undermines any collision warning issued based on this data, since it is expensive for operators to perform a maneuver. Thus, a security-conscious data sharing policy, impedes utility and commercial growth of the space industry.

On the other hand, improving utility raises several privacy/security concerns. National security concerns have become more critical, owing to the recent development of anti-satellite weapons and other counter-space capabilities. Accurate knowledge of space assets present risks from directed energy weapons, electronic jamming, kinetic energy threats, and other orbital threats [64, 40, 38]. From a commercial perspective, there are concerns that access to highly accurate SSA data will allow operators to assess their competitor's coverage limits, detection capabilities, and details of operation. This can be detrimental to the emerging space economy. In summary, low-accuracy SSA data increases risk of collision but reduces risk from counter-space operations and protects details of operations, i.e. it improves privacy but degrades utility. Whereas, high-accuracy SSA data improves utility but degrades privacy/security.

Currently, the US military adds synthetic noise to the public domain SSA data, much like the early GPS data model, which impacts how accurately the space objects can be tracked. Currently, this noise level is chosen conservatively and is mostly privacy or national-security conscious. With the deployment of mega-constellations in low earth orbits [66], this conservative approach will not work and will impede accurate space traffic management. Consequently, in June 2018, the Space Policy Directive 3, directed the US Department of Defense to give the publicly releasable portion of its space situational awareness data to the Commerce Department [41]. This initiative will allow non-military entities to create and sell SSA services to governments and

satellite operators. Thus, the SSA data can be commoditized [48], with stratified pricing models reflecting different levels of accuracy, enabling creation of several value-added services in space-data analytics. These new developments in SSA data sharing, present an important question: *what should be the accuracy in the SSA data that satisfies given utility and privacy objectives?* What we lack is a framework that addresses this question in a methodological manner, enabling more informed privacy and utility preserving policies for sharing SSA data.

Privacy in dynamical systems is an emerging area of work, and has been primarily in differential privacy [22, 60, 23, 15, 47, 46, 28], where the focus is to ensure that participation of an entity or an individual does not change the outcome significantly, and also guarantees privacy of an entity based on aggregated data from many entities. Our focus here is on inferential privacy [72, 32, 76], where we are trying to bound the inferences an adversary can make based on auxiliary information. In both these kinds of privacy, the mechanism for data obfuscation is either corruption of data with synthetic noise, or projection of data to a lower dimensional space. In this paper, we use synthetic noise to formulate various algorithms for privacy-utility tradeoff.

Our focus is on privacy of entities that are governed by dynamical systems. The dynamical system, which is a space object in this paper, is observed by various geographically distributed sensors. These observations are functions of states, and are noisy due to imperfect sensing. A filtering process uses these measurements to estimate the true state of the space object, which also has errors. The accuracy of the estimates is a measure of inference privacy. More error results in more privacy, and consequently less utility. For dynamical systems with Gaussian uncertainty models, Kalman filter gives the minimum variance estimates, which is function of the prior, the sensor model, and the sensor noise. In [76], the authors manipulate the mea-

surement data by compressing it using a linear transformation, hereby regulating the error covariance matrix or the inference privacy. This linear transformation, which creates *synthetic sensors*, is designed appropriately to achieve the privacy goals. In this paper, we regulate the inference privacy by manipulating the measurement noise covariance. This is done by adding synthetic noise to the measurements.

In this paper, we look at the trade-off between privacy and utility in the Ensemble Kalman Filtering (EnKF) [26] and Unscented Kalman Filtering (UKF) framework [45], which is commonly used for data assimilation and forecasting in SSA applications. We assume that the privacy and utility constraints are specified in terms of bounds on the error variances in the state-estimates. This paper presents a convex optimization framework that determines the optimal synthetic noise to be added to the sensor data, which will satisfy the given privacy and utility bounds. The formulation, however, is quite general and is not limited to SSA related problems. To the best of our knowledge, this is the first paper that addresses the privacy-utility trade-off using synthetic noise in the EnKF/UKF framework. Our future work will be on extending this work to other particle-filtering algorithms [58].

This paper is organized as follows. We first present the model for the space object used in the privacy-utility formulation. This is followed by a very brief technical summary of Ensemble and Unscented Kalman filtering. Key technical contributions are presented in §4.3, as theorems 15 to 18, and corollary 1. This is followed by an example that applies the proposed privacy-utility formulation to a realistic space-object tracking problem and considers various scenarios. The paper concludes with a summary section.

4.2   Background Material

We next present preliminaries on dynamical models of space objects, their sources of uncertainty, and measurement models. This is followed by a brief description of Ensemble and Unscented Kalman filtering. In particular, how the estimate and error variances are computed from particles, and updated from available measurements. This section also defines all the notations used to present the technical material.

4.2.1   System Model

We assume the motion of a space object is given by the differential equation (Cowell's Formulation)

$$\ddot{\boldsymbol{r}} = \frac{\mu_{\oplus}}{r^3}\boldsymbol{r} + \boldsymbol{a}_{pert}(\boldsymbol{r}, \dot{\boldsymbol{r}}, t, \boldsymbol{\psi}), \tag{4.1}$$

where $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are the position and velocity of the space object in 3 dimensions respectively. The vector $\boldsymbol{a}_{\text{pert}}$ encapsulates all the perturbing accelerations of the space object other than those due to the two-body point mass gravitational acceleration. These perturbations could be due to higher-order gravity terms, atmospheric drag, solar radiations, etc; and $\boldsymbol{\psi} \in \mathbb{R}^d$ parameterizes these effects. The perturbation involved in the dynamics of the satellite is essentially parametric [79], which simplifies the uncertainty propagation algorithm considerably, and particle-based methods can be used [36].

Sensor observations essentially provide range, azimuth and elevation information and are derived from signals of electromagnetic radiation. Sensors receive one-way and two-way data via radar or laser measurements, which have varying degree of

accuracy for a particular sensing objective such as obtaining range, range-rate and angular information for a satellite. Most sensors can provide observations at far higher frequencies than required. Thus, data density is not an issue. However, the proximity of data to other information is a more serious concern in satellite tracking. In some cases, due to mechanical design, geographical, and political constraints, observations are limited to a small arc of the orbit, which we denote as *short-arc* observations. If a satellite is observed over multiple revolutions, it is referred to as *long-arc* observation. Long-arc observations are preferred because they provide more accurate determination of the satellite's orbit. Short-arc observations on the other hand are important for more accurate near-term satellite motion. Thus the sensor data is available at multiple-time scales, and a data privacy-utility formulation must account for this multi-scale nature. For space situational awareness problems, keeping track of a satellite's location in its orbit is critical, which relies on the short-arc measurements. Data from these sensors are only available when the satellites are within sensor range. Thus, predictions must be made over long intervals between these sensor updates, which is associated with larger error growths. This impacts the tradeoff between accuracy and privacy. We expect that higher accuracy data is necessary to resolve errors after long-term uncertainty propagation, and lower accuracy data is necessary to resolve errors after long-term uncertainty propagation. That is, short-arc observations can have lower accuracy than long-arc observations, to achieve a given accuracy in the state-estimate. Consequently, the data privacy-utility policy must also be time varying, accounting for the variability in the data frequency.

Let $\boldsymbol{x} \in \mathbb{R}^n$ be the state vector describing the motion of a space object. While there exists many satellite based coordinates systems to choose from for orbital state-

estimation, it is well known that many of these systems are ill defined for some circular and elliptical orbits [79]. The equinoctial system eliminates this difficulty and is the best choice to quantify uncertainty in satellite dynamics. However, the data privacy-utility formulation presented in this paper is agnostic to the choice of the coordinate system.

Let $\boldsymbol{y} \in \mathbb{R}^{n_y}$ be the measurement variable (measuring range, range-rate and angular information for a satellite, etc), which is mathematically defined as

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}) + \boldsymbol{v}, \tag{4.2}$$

where $\boldsymbol{v}$ represents sensor noise, assumed to be zero-mean Gaussian random process, and $\boldsymbol{x}$ is the system state vector.

In this paper, we develop the data privacy-utility algorithm in discrete time, and write (4.1) and (4.2) in discrete time as

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{\psi}_k), \tag{4.3}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_k, \tag{4.4}$$

where $\boldsymbol{f}(\cdot, \cdot)$ represents the discrete-time dynamics and $\boldsymbol{x}_k := (x, y, z, \dot{x}, \dot{y}, \dot{z})^T$ is the state vector, and sensor noise $\boldsymbol{v}_k$ is assumed to be independent zero-mean Gaussian random variable, i.e. $\boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R}_k)$ and $\mathbb{E}\left[\boldsymbol{v}_k \boldsymbol{v}_l^T\right] = \boldsymbol{R}_k \delta_{kl}$. We further assume that $\boldsymbol{R}_k$ is a diagonal matrix, and define the inverse of $\boldsymbol{R}_k$ as the precision matrix $\boldsymbol{S}_k$. Finally, the only uncertainty considered here is the initial condition uncertainty, which is also assumed to be Gaussian i.e. $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ and independent of $\{\boldsymbol{v}_k\}$. Other variables have the same meaning as in (4.1) and (4.2).

To account for the multiple-time scales in the data, and exploit the periodic nature of the satellite motion, we augment the system in (4.1) and (4.2) to capture the motion over q time steps. Starting from time kq, for $k \in \{1, 2, \cdots\}$, we define the augmented discrete-time system dynamics as

$$\boldsymbol{X}_{k+1} = \boldsymbol{F}(\boldsymbol{X}_k, \boldsymbol{\Psi}_k), \ \boldsymbol{Y}_k = \boldsymbol{H}(\boldsymbol{X}_k) + \boldsymbol{V}_k, \tag{4.5}$$

where,

$$\left. \begin{aligned} \boldsymbol{X}_k &:= [\boldsymbol{x}_{kq-q+1}^T, \cdots, \boldsymbol{x}_{kq}^T]^T, \\ \boldsymbol{Y}_k &:= [\boldsymbol{y}_{kq-q+1}^T, \cdots, \boldsymbol{y}_{kq}^T]^T, \\ \boldsymbol{\Psi}_k &:= [\boldsymbol{\psi}_{kq-q+1}^T, \cdots, \boldsymbol{\psi}_{kq}^T]^T, \\ \boldsymbol{V}_k &:= [\boldsymbol{v}_{kq-q+1}^T, \cdots, \boldsymbol{v}_{kq}^T]^T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\mathcal{R}}_k), \\ \boldsymbol{\mathcal{R}}_k &:= \mathbf{diag}(\boldsymbol{R}_{kq-q+1}, \cdots, \boldsymbol{R}_{kq}), \end{aligned} \right\} \tag{4.6}$$

denotes stacked variables. Function $\boldsymbol{F}_k(.)$ is recursively generated using $\boldsymbol{f}(.)$. The initial condition $\boldsymbol{X}_0$ is determined by propagating the uncertain initial condition $\boldsymbol{x}_0$ over q time steps. It should be noted that the augmented model represents a q-step q-shift process, instead of a q-step sliding-window process. Augmenting the model to q time steps allows $\boldsymbol{Y}_k$ to include multi-rate sensor data $\boldsymbol{y}$, by defining q to be the lowest common multiple of the various sensing rates, resulting in a q-periodic system. In this paper, we formulate the privacy-utility policy for the multi-rate data, using the system in (4.5).

4.2.2 Review of Ensemble and Unscented Kalman Filter

In this paper, the data privacy-utility policy is developed in the EnKF and UKF framework, which is summarized next. The filtering process for the augmented model

128

(4.5) consists of uncertainty propagation using the state-dynamics to obtain the prior or model predicted estimate, and using the data from measurements to obtain the posterior estimate. In EnKF, *random* samples are generated directly from the state probability density function (PDF) using standard sampling techniques. Whereas, UKF uses a *deterministic* sampling technique known as the unscented transformation to pick a minimal set of sample points (called sigma points) around the mean. In both these approaches, the sample points are propagated using the nonlinear dynamics, from which the prior mean and variance of the states are computed. These are updated using measurements to arrive at the state estimate with minimum error variance. We briefly present next, the technical details of both these approaches, which will be necessary for formulating the data-privacy vs data-utility problem in the EnKF/UKF framework.

4.2.2.1 Ensemble Kalman Filter

In this section, we briefly present ensemble Kalman filtering. Let $\boldsymbol{\mathcal{X}}_k^+ \in \mathbb{R}^{nq \times N}$ be the matrix with N number of *posterior* samples $\boldsymbol{X}_k^{i+}$ at time k, i.e.

$$\boldsymbol{\mathcal{X}}_k^+ := \begin{bmatrix} \boldsymbol{X}_k^{1+} & \boldsymbol{X}_k^{2+} & \cdots & \boldsymbol{X}_k^{N+} \end{bmatrix}.$$

The sample mean is then given by,

$$\boldsymbol{\mu}_k^+ := \mathbb{E}\left[\boldsymbol{X}_k^+\right] \approx \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{X}_k^{i+} = \frac{1}{N}\boldsymbol{\mathcal{X}}_k^+\mathbf{1}_N,$$

where $\mathbf{1}_N \in \mathbb{R}^N$ is a column vector of N ones.

Defining,

$$\bar{\mathcal{X}}_k^+ := \begin{bmatrix} \boldsymbol{\mu}_k^+ & \cdots & \boldsymbol{\mu}_k^+ \end{bmatrix} = \boldsymbol{\mu}_k^+ \mathbf{1}^T = \frac{1}{N} \mathcal{X}_k^+ \mathbf{1}_N \mathbf{1}_N^T,$$

we can compute the variance from the samples $\boldsymbol{\Sigma}_{xx,k}^+$ as,

$$\mathbb{E}\left[(\boldsymbol{X}_k^{i+} - \boldsymbol{\mu}_k^+)(\boldsymbol{X}_k^{i+} - \boldsymbol{\mu}_k^+)^T\right] \approx \mathcal{X}_k^+ \boldsymbol{A} \mathcal{X}_k^{+T}. \tag{4.7}$$

where

$$\boldsymbol{A} := \left[\frac{1}{N-1} \left(\boldsymbol{I}_N - \frac{\mathbf{1}_N \mathbf{1}_N^T}{N} \right) \left(\boldsymbol{I}_N - \frac{\mathbf{1}_N \mathbf{1}_N^T}{N} \right) \right]$$

The state of each ensemble member at the next time step is determined using the dynamics model:

$$\boldsymbol{X}_{k+1}^{i-} = \boldsymbol{F}(\boldsymbol{X}_k^{i+}, \boldsymbol{\Psi}_k^i). \tag{4.8}$$

In the EnKF framework, the model predicted ensembles (or prior ensembles) are corrected using measurements, as proposed by Evensen and Van Leeuwen [26, 27]

$$\boldsymbol{X}_{k+1}^{i+} = \boldsymbol{X}_{k+1}^{i-} + \boldsymbol{\Sigma}_{xy,k+1}^- \left(\boldsymbol{\Sigma}_{yy,k+1}^- + \mathcal{R}_{k+1} \right)^{-1} \left(\boldsymbol{Y}_{k+1} - \boldsymbol{H}_{k+1}(\boldsymbol{X}_{k+1}^{i-}) + \boldsymbol{\epsilon}_k^i \right), \quad (4.9)$$

where $\boldsymbol{\epsilon}_k^i$ is sampled from $\mathcal{N}(\mathbf{0}, \mathcal{R}_k)$. Quantities $\boldsymbol{\Sigma}_{xy,k+1}^-$ and $\boldsymbol{\Sigma}_{yy,k+1}^-$ are computed from the samples using

$$\boldsymbol{\Sigma}_{xy,k+1}^- := \frac{1}{N-1} \left(\mathcal{X}_{k+1}^- - \bar{\mathcal{X}}_{k+1}^- \right) \left(\boldsymbol{H}_{k+1}(\mathcal{X}_{k+1}^-) - \boldsymbol{H}_{k+1}(\bar{\mathcal{X}}_{k+1}^-) \right)^T, \text{ and} \quad (4.10)$$

$$\boldsymbol{\Sigma}_{yy,k+1}^- := \frac{1}{N-1} \left(\boldsymbol{H}_{k+1}(\mathcal{X}_{k+1}^-) - \boldsymbol{H}_{k+1}(\bar{\mathcal{X}}_{k+1}^-) \right) \left(\boldsymbol{H}_{k+1}(\mathcal{X}_{k+1}^-) - \boldsymbol{H}_{k+1}(\bar{\mathcal{X}}_{k+1}^-) \right)^T.$$

$$\tag{4.11}$$

The variance update equation of the augmented model is given by

$$\Sigma^+_{\boldsymbol{xx},k+1} = \Sigma^-_{\boldsymbol{xx},k+1} - \Sigma^-_{\boldsymbol{xy},k+1} \left(\Sigma^-_{\boldsymbol{yy},k+1} + \mathcal{R}_{k+1} \right)^{-1} \Sigma^{-T}_{\boldsymbol{xy},k+1}, \qquad (4.12)$$

where $\Sigma^-_{\boldsymbol{xx},k+1} = \boldsymbol{\mathcal{X}}^-_{k+1} \boldsymbol{A} \boldsymbol{\mathcal{X}}^{-T}_{k+1}$. Later in the paper, we will be using (4.12) to determine the optimal privacy-utility tradeoff.

4.2.2.2 Unscented Kalman Filter

The main difference between EnKF and UKF is the generation of the samples and computation of the first two moments from the samples. The dynamic update step from $k \to k+1$ starts with generating deterministic points called σ points. Our dynamic model has no process noise. To capture the mean $\boldsymbol{\mu}^+_k$ of the state vector $\boldsymbol{X}^+_k$, where $\boldsymbol{X}^+_k \in \mathbb{R}^{nq}$, as well as the error covariance $\Sigma^+_{xx,k}$ the sigma points are chosen as

$$\boldsymbol{X}^{0+}_k = \boldsymbol{\mu}^+_k,$$
$$\boldsymbol{X}^{i+}_k = \boldsymbol{\mu}^+_k + \left(\sqrt{(nq+\rho)\Sigma^+_{\boldsymbol{xx},k}} \right)_i, i = 1, ..., nq,$$
$$\boldsymbol{X}^{i+}_k = \boldsymbol{\mu}^+_k - \left(\sqrt{(nq+\rho)\Sigma^+_{\boldsymbol{xx},k}} \right)_{i-nq}, i = nq+1, ..., 2nq,$$

with associated weights as

$$\boldsymbol{\omega}^{(m)}_0 = \rho/(nq+\rho),$$
$$\boldsymbol{\omega}^{(c)}_0 = \rho/(nq+\rho) + (1 - \alpha^2 + \beta),$$
$$\boldsymbol{\omega}^{(m)}_i = 1/\{2(nq+\rho)\}.$$

The weight vectors are:

$$\boldsymbol{\mathcal{W}}^m = [\boldsymbol{\omega}_0^{(m)} \boldsymbol{\omega}_1^{(m)} ... \ \boldsymbol{\omega}_{2nq+1}^{(m)}]^T,$$

$$\boldsymbol{\mathcal{W}}^c = [\boldsymbol{\omega}_0^{(c)} \boldsymbol{\omega}_1^{(c)} ... \ \boldsymbol{\omega}_{2nq+1}^{(c)}]^T,$$

where $\rho = \alpha^2(nq + \kappa) - nq$ is the scaling parameter, α is set to 0.001, κ is set to 0, and β is 2 in this work. The term $\left(\sqrt{(nq + \rho)\boldsymbol{\Sigma}_{xx,k}^+}\right)_i$ represents ith row of the matrix square root.

The propagated state of each ensemble member at time $k+1$ is generated exactly as EnKF by using (4.8). However for UKF, the corresponding prior mean and variance at time $k + 1$ are given by

$$\boldsymbol{\mu}_{k+1}^- = \boldsymbol{\mathcal{X}}_{k+1}^- \boldsymbol{\mathcal{W}}^m, \tag{4.13}$$

$$\boldsymbol{\Sigma}_{xx,k+1}^- = \boldsymbol{\mathcal{X}}_{k+1}^- \boldsymbol{B}_k \boldsymbol{\mathcal{X}}_{k+1}^{-T}, \tag{4.14}$$

where $\boldsymbol{B}_k := \boldsymbol{L}\boldsymbol{L}^T$, and $\boldsymbol{L} := \mathbf{diag}(\boldsymbol{\mathcal{W}}^c) - \boldsymbol{\mathcal{W}}^c \mathbf{1}_{2nq+1}^T$.

We next define the following terms that will be used in the following measurement update phase,

$$\boldsymbol{\mathcal{Y}}_{k+1}^- := \boldsymbol{H}(\boldsymbol{\mathcal{X}}_{k+1}^-), \tag{4.15}$$

$$\bar{\boldsymbol{\mathcal{Y}}}_{k+1}^- := \boldsymbol{\mathcal{Y}}_{k+1}^- \boldsymbol{\mathcal{W}}^m \mathbf{1}_{2nq+1}^T, \text{ and} \tag{4.16}$$

$$\bar{\boldsymbol{\mathcal{X}}}_{k+1}^- := \boldsymbol{\mathcal{X}}_{k+1}^- \boldsymbol{\mathcal{W}}^m \mathbf{1}_{2nq+1}^T, \tag{4.17}$$

132

where

$$\boldsymbol{\mathcal{Y}}_{k+1}^{-} := \begin{bmatrix} \boldsymbol{Y}_{k+1}^{1-} & \boldsymbol{Y}_{k+1}^{2-} & \cdots & \boldsymbol{Y}_{k+1}^{(2nq+1)-} \end{bmatrix}, \tag{4.18}$$

$$\boldsymbol{\mathcal{X}}_{k+1}^{-} := \begin{bmatrix} \boldsymbol{X}_{k+1}^{1-} & \boldsymbol{X}_{k+1}^{2-} & \cdots & \boldsymbol{X}_{k+1}^{(2nq+1)-} \end{bmatrix}. \tag{4.19}$$

were $\boldsymbol{Y}_{k+1}^{i-} = \boldsymbol{H}(\boldsymbol{X}_{k+1}^{i-})$ is the ith measurement sample generated using the measurement model without the measurement noise.

For the measurement update step, we first calculate $\boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-}$ and $\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^{-}$ as

$$\boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-} := \left(\boldsymbol{\mathcal{X}}_{k+1}^{-} - \bar{\boldsymbol{\mathcal{X}}}_{k+1}^{-} \right) \times \mathbf{diag}\left(\boldsymbol{\mathcal{W}}^c \right) \left(\boldsymbol{\mathcal{Y}}_{k+1}^{-} - \bar{\boldsymbol{\mathcal{Y}}}_{k+1}^{-} \right)^T \tag{4.20}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^{-} := \left(\boldsymbol{\mathcal{Y}}_{k+1}^{-} - \bar{\boldsymbol{\mathcal{Y}}}_{k+1}^{-} \right) \times \mathbf{diag}\left(\boldsymbol{\mathcal{W}}^c \right) \left(\boldsymbol{\mathcal{Y}}_{k+1}^{-} - \bar{\boldsymbol{\mathcal{Y}}}_{k+1}^{-} \right)^T, \tag{4.21}$$

and then obtain the updated variance using (4.12).

Remark 8. Since the variance update equation for EnKF and UKF are identical, this allows us to formulate a common data privacy-utility policy for both the filtering frameworks. Equation (4.12) is fundamental in determining the optimal noise for privacy-utility aware data-sharing.

4.3   Privacy-Utility Aware Data Sharing in EnKF and UKF

As defined in (4.12), the accuracy of the state-estimate from EnKF/UKF is quantified by $\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^{+}$, which is defined as

$$\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^{+} = \boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^{-} - \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-} \left(\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^{-} + \boldsymbol{\mathcal{R}}_{k+1} \right)^{-1} \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T},$$

and is the minimum variance for a given $\mathcal{R}_{k+1}$. Clearly, changing $\mathcal{R}_{k+1}$ will change $\mathbf{\Sigma}^+_{\boldsymbol{xx},k+1}$. From an estimation perspective, $\mathcal{R}_{k+1}$ is given by the sensing hardware, and defines the noise in the measurements $\boldsymbol{Y}_k$. However, from a data-sharing perspective, we are concerned with the privacy-utility tradeoff in sharing $\boldsymbol{Y}_k$ with end users, which can be influenced by changing $\mathcal{R}_{k+1}$. For a state variable, larger error covariance results in increase in privacy and decrease in its utility. On the other hand, a smaller error covariance, leads to increase in utility and decrease in privacy.

In this paper, we formulate a data privacy-utility trade-off policy in terms of synthetic noise to be added to $\boldsymbol{Y}_k$ to regulate how accurately end users are able to estimate the satellite states in the EnKF/UKF framework. Consequently, we modify (4.12), by defining

$$\mathcal{R}_{k+1} := \mathcal{R}^{\text{sensor}}_{k+1} + \mathcal{R}^{\text{data}}_{k+1},$$

where $\mathcal{R}^{\text{sensor}}_{k+1}$ is the known sensor noise variance and quantifies the accuracy of the measured data, and $\mathcal{R}^{\text{data}}_{k+1}$ defines the additional synthetic noise to be determined that should be added to the measured data to achieve a privacy-utility tradeoff. Usually, the sensor noise is assumed to be uncorrelated, which results in a diagonal $\mathcal{R}^{\text{sensor}}_{k+1}$ with positive entries. However, $\mathcal{R}^{\text{data}}_{k+1}$ can be assumed to be a general positive semidefinite matrix.

Therefore, $\mathbf{\Sigma}^+_{\boldsymbol{xx},k+1}$ is now defined as

$$\mathbf{\Sigma}^+_{\boldsymbol{xx},k+1} = \mathbf{\Sigma}^-_{\boldsymbol{xx},k+1} - \mathbf{\Sigma}^-_{\boldsymbol{xy},k+1} \left(\mathbf{\Sigma}^-_{\boldsymbol{yy},k+1} + \mathcal{R}^{\text{sensor}}_{k+1} + \mathcal{R}^{\text{data}}_{k+1} \right)^{-1} \mathbf{\Sigma}^{-T}_{\boldsymbol{xy},k+1}. \qquad (4.22)$$

We generalize the formulation by defining utility in terms of variable $\boldsymbol{X}_u := \boldsymbol{M}_u \boldsymbol{X}$, and privacy in terms of variable $\boldsymbol{X}_p := \boldsymbol{M}_p \boldsymbol{X}$, where $\boldsymbol{M}_u$, and $\boldsymbol{M}_p$ are known

matrices. Partitioning the augmented state space into privacy and utility variable is motivated by the work in [73]. The authors used it to partition the state variable at a particular instant into privacy and utility variables, i.e. partitioning along the dimension of the variable. In this paper, since we augment the state-vector to include time-series data, our partitioning matrix $\boldsymbol{M}_p$ and $\boldsymbol{M}_u$ partitions the augmented state-vector in both space and time. Consequently, we can achieve privacy-utility tradeoffs in space and time, and are highlighted in the examples.

The error variance in the estimates for $\boldsymbol{X}_u$ and $\boldsymbol{X}_p$ are given by

$$
\begin{aligned}
\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^+ &:= \boldsymbol{M}_u\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^+\boldsymbol{M}_u^T, \\
&= \boldsymbol{M}_u\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^-\boldsymbol{M}_u^T - \boldsymbol{M}_u\boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^-\left(\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^- + \boldsymbol{\mathcal{R}}_{k+1}^{\text{sensor}} + \boldsymbol{\mathcal{R}}_{k+1}^{\text{data}}\right)^{-1}\boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T}\boldsymbol{M}_u^T, \text{ and}
\end{aligned}
$$
(4.23)

$$
\begin{aligned}
\boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^+ &:= \boldsymbol{M}_p\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^+\boldsymbol{M}_p^T, \\
&= \boldsymbol{M}_p\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^-\boldsymbol{M}_p^T - \boldsymbol{M}_p\boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^-\left(\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^- + \boldsymbol{\mathcal{R}}_{k+1}^{\text{sensor}} + \boldsymbol{\mathcal{R}}_{k+1}^{\text{data}}\right)^{-1}\boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T}\boldsymbol{M}_p^T.
\end{aligned}
$$
(4.24)

Thus, the objective here is to determine $\boldsymbol{\mathcal{R}}_{k+1}^{\text{data}}$ that allows end users to estimate states $\boldsymbol{X}_u$ accurately enough, but not estimate states $\boldsymbol{X}_p$ too accurately, i.e. given $\boldsymbol{\Sigma}_p$ and $\boldsymbol{\Sigma}_u$, we would like to determine $\boldsymbol{\mathcal{R}}_{k+1}^{\text{data}}$ that satisfies $\boldsymbol{\Sigma}_p \leq \boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^+$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^+ \leq \boldsymbol{\Sigma}_u$ simultaneously, where $\boldsymbol{\Sigma}_p$ defines the accuracy limit from a privacy perspective and $\boldsymbol{\Sigma}_u$ defines the accuracy limit from a utility perspective.

Since $\boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^+$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^+$ depend on $\boldsymbol{\mathcal{R}}_{k+1}^{\text{data}}$, they are upper and lower bounded

by values obtained for $\mathcal{R}_{k+1}^{\text{data}} = \infty$ and $\mathcal{R}_{k+1}^{\text{data}} = \mathbf{0}$ respectively, i.e.

$$\boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^{+}(0) \leq \boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^{+}(\mathcal{R}_{k+1}^{\text{data}}) \leq \boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^{+}(\infty) = \boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^{-}, \tag{4.25}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{+}(0) \leq \boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{+}(\mathcal{R}_{k+1}^{\text{data}}) \leq \boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_u,k+1}^{+}(\infty) = \boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{-}, \tag{4.26}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^{+}(\cdot)$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{+}(\cdot)$ denote the functional dependence of the variances on $\mathcal{R}_{k+1}^{\text{data}}$. Consequently, these inequalities define the limits on the achievable privacy and utility.

We next present the main results of this paper, which are convex optimization formulations for achieving optimal privacy-utility tradeoff in the general EnKF/UKF framework. We use a relaxed definition for privacy and utility, by defining them in terms of trace of the variance matrices, i.e.

$$\text{Privacy: } \mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^{+}\right] \geq \gamma_p, \tag{4.27a}$$

$$\text{Utility: } \mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{+}\right] \leq \gamma_u, \tag{4.27b}$$

where γ_p and γ_u are user defined.

4.3.1 Maximum Noise Satisfying Utility Constraints

Here we present a convex optimization problem, which determines the *maximum* synthetic noise that can be added and still satisfy the upper bound on $\mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{+}\right]$ and is given by the following theorem.

Theorem 15. *The maximum noise that satisfies* $\mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{+}\right] \leq \gamma_u$, *is given by the solution of the following optimization problem*

$$\min_{\boldsymbol{\mathcal{S}}_{k+1}^{data} \geq 0, \boldsymbol{Q}_u \geq 0} \mathbf{tr}\left[\boldsymbol{\mathcal{S}}_{k+1}^{data}\right], \tag{4.28a}$$

subject to

$$\begin{bmatrix} \boldsymbol{Q}_u - \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^- \boldsymbol{M}_u^T + \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \boldsymbol{Z}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T & \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \\ \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T & \boldsymbol{Z} + \boldsymbol{Z} \boldsymbol{\mathcal{S}}_{k+1}^{data} \boldsymbol{Z} \end{bmatrix} \geq 0, \tag{4.28b}$$

$$\mathbf{tr}\left[\boldsymbol{Q}_u\right] \leq \gamma_u, \tag{4.28c}$$

where $\boldsymbol{Z} := \boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^- + \boldsymbol{\mathcal{R}}_{k+1}^{sensor}$. *The maximum noise in the data for which the utility constraint is satisfied is then given by* $\boldsymbol{\mathcal{R}}_{k+1}^{data} := \left(\boldsymbol{\mathcal{S}}_{k+1}^{data}\right)^{-1}$.

Proof. Using (4.23), $\mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_u \boldsymbol{x}_u, k+1}^+\right] \leq \gamma_u$, is equivalent to

$$\boldsymbol{Q}_u - \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^- \boldsymbol{M}_u^T + \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \left(\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^- + \boldsymbol{\mathcal{R}}_{k+1}^{sensor} + \boldsymbol{\mathcal{R}}_{k+1}^{data}\right)^{-1} \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T \geq 0,$$

and $\mathbf{tr}\left[\boldsymbol{Q}_u\right] \leq \gamma_u$. Using matrix inversion lemma (Hua's identity), we get

$$\left(\underbrace{\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^- + \boldsymbol{\mathcal{R}}_{k+1}^{sensor}}_{:=\boldsymbol{Z}} + \boldsymbol{\mathcal{R}}_{k+1}^{data}\right)^{-1} = \boldsymbol{Z}^{-1} - \left\{\boldsymbol{Z} + \boldsymbol{Z}\left(\boldsymbol{\mathcal{R}}_{k+1}^{data}\right)^{-1}\boldsymbol{Z}\right\}^{-1}.$$

Therefore, the inequality becomes

$$\boldsymbol{Q}_u - \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^- \boldsymbol{M}_u^T + \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \left[\boldsymbol{Z}^{-1} - \left\{\boldsymbol{Z} + \boldsymbol{Z}\left(\boldsymbol{\mathcal{R}}_{k+1}^{data}\right)^{-1}\boldsymbol{Z}\right\}^{-1}\right] \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T \geq 0,$$

or

$$\boldsymbol{Q}_u - \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^- \boldsymbol{M}_u^T + \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \boldsymbol{Z}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T -$$
$$\boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \left\{ \boldsymbol{Z} + \boldsymbol{Z} \left(\boldsymbol{\mathcal{R}}_{k+1}^{\mathrm{data}} \right)^{-1} \boldsymbol{Z} \right\}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T \geq 0.$$

Using Schur complement we get

$$\begin{bmatrix} \boldsymbol{Q}_u - \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^- \boldsymbol{M}_u^T + \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \boldsymbol{Z}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T & \boldsymbol{M}_u \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^- \\ \boldsymbol{\Sigma}_{\boldsymbol{xy},k+1}^{-T} \boldsymbol{M}_u^T & \boldsymbol{Z} + \boldsymbol{Z} \left(\boldsymbol{\mathcal{R}}_{k+1}^{\mathrm{data}} \right)^{-1} \boldsymbol{Z} \end{bmatrix} \geq 0.$$

Introducing a new variable $\boldsymbol{\mathcal{S}}_{k+1}^{\mathrm{data}} := \left(\boldsymbol{\mathcal{R}}_{k+1}^{\mathrm{data}} \right)^{-1}$, which is the precision of the data, we get the LMI in (4.28b). Therefore, maximization of $\mathbf{tr}\left[\boldsymbol{\mathcal{R}}_{k+1}^{\mathrm{data}} \right]$ becomes minimization of $\mathbf{tr}\left[\boldsymbol{\mathcal{S}}_{k+1}^{\mathrm{data}} \right]$. $\square$

The above optimization is performed every time a new batch of $\boldsymbol{Y}_{k+1}$ is shared, which is corrupted using $\boldsymbol{\mathcal{R}}_{k+1}^{\mathrm{data}}$.

For a special case of linear sensor model, i.e. $\boldsymbol{H}(\boldsymbol{X}_k) := \boldsymbol{C}\boldsymbol{X}_k$, we can substitute $\boldsymbol{\Sigma}_{\boldsymbol{xy}}^- := \boldsymbol{\Sigma}_{\boldsymbol{xx}}^- \boldsymbol{C}^T$ and $\boldsymbol{\Sigma}_{\boldsymbol{yy}}^- := \boldsymbol{C}\boldsymbol{\Sigma}_{\boldsymbol{xx}}^- \boldsymbol{C}^T$, in the above optimization problem.

Remark 9. In the above theorem, we need to compute the inverse of $\boldsymbol{Z} := (\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^- + \boldsymbol{\mathcal{R}}_{k+1}^{\mathrm{sensor}})$, which may be ill-conditioned, particularly when $\boldsymbol{\Sigma}_{\boldsymbol{yy},k+1}^-$ is rank deficient and $\boldsymbol{\mathcal{R}}_{k+1}^{\mathrm{sensor}}$ is small (corresponding to very precise sensor measurements, for example from laser ranging). This problem occurs in satellite tracking problems. We next present an alternate formulation, which does not require the matrix inverse, but the matrix square root, at the expense of increasing the problem size. This is given by the following result, assuming linear measurement model.

Theorem 16. *The maximum noise that satisfies* $\textbf{tr}\left[\Sigma^{+}_{\bm{x}_u\bm{x}_u,k+1}\right] \leq \gamma_u$, *is given by the solution of the following optimization problem*

$$\min_{\bm{\mathcal{S}}^{data}_{k+1}\geq 0,\,\bm{Q}_u\geq 0,\,\bm{K}} \textbf{tr}\left[\bm{\mathcal{S}}^{data}_{k+1}\right], \tag{4.29a}$$

subject to

$$\begin{bmatrix} \bm{Q}_u & \bm{M}_u(\bm{I}-\bm{KC})\sqrt{\Sigma^{-}_{\bm{xx},k+1}} & \bm{M}_u\bm{K} & \bm{M}_u\bm{K} \\ \sqrt{\Sigma^{-}_{\bm{xx},k+1}}(\bm{I}-\bm{KC})^T\bm{M}_u^T & \bm{I} & \bm{0} & \bm{0} \\ \bm{K}^T\bm{M}_u^T & \bm{0} & \left(\mathcal{R}^{sensor}_{k+1}\right)^{-1} & \bm{0} \\ \bm{K}^T\bm{M}_u^T & \bm{0} & \bm{0} & \bm{\mathcal{S}}^{data}_{k+1} \end{bmatrix} \geq 0, \tag{4.29b}$$

$$\textbf{tr}\left[\bm{Q}_u\right] \leq \gamma_u. \tag{4.29c}$$

Proof. Recalling that the posterior variance in Kalman filtering is also given by $\Sigma^{+}_{\bm{xx}} = (\bm{I}-\bm{KC})\Sigma^{-}_{\bm{xx}}(\bm{I}-\bm{KC})^T + \bm{KRK}^T$, $\textbf{tr}\left[\Sigma^{+}_{\bm{x}_u\bm{x}_u,k+1}\right] \leq \gamma_u$, is equivalent to

$$\bm{Q}_u - \bm{M}_u(\bm{I}-\bm{KC})\Sigma^{-}_{\bm{xx},k+1}(\bm{I}-\bm{KC})^T\bm{M}_u^T + \bm{M}_u\bm{K}\left(\mathcal{R}^{sensor}_{k+1} + \mathcal{R}^{data}_{k+1}\right)\bm{K}^T\bm{M}_u^T \geq 0, \tag{4.30}$$

along with $\textbf{tr}\left[\bm{Q}_u\right] \leq \gamma_u$.

Using Schur complement we can write (4.30) as,

$$\begin{bmatrix} \bm{Q}_u & \bm{M}_u(\bm{I}-\bm{KC})\sqrt{\Sigma^{-}_{\bm{xx},k+1}} & \bm{M}_u\bm{K} & \bm{M}_u\bm{K} \\ \sqrt{\Sigma^{-}_{\bm{xx},k+1}}(\bm{I}-\bm{KC})^T\bm{M}_u^T & \bm{I} & \bm{0} & \bm{0} \\ \bm{K}^T\bm{M}_u^T & \bm{0} & \left(\mathcal{R}^{sensor}_{k+1}\right)^{-1} & \bm{0} \\ \bm{K}^T\bm{M}_u^T & \bm{0} & \bm{0} & \left(\mathcal{R}^{data}_{k+1}\right)^{-1} \end{bmatrix} \geq 0.$$

139

Introducing a new variable $\boldsymbol{\mathcal{S}}_{k+1}^{\text{data}} := \left(\boldsymbol{\mathcal{R}}_{k+1}^{\text{data}}\right)^{-1}$, we get the LMI in (4.29b), in terms of $\boldsymbol{\mathcal{S}}_{k+1}^{\text{data}}$. Maximization of $\mathbf{tr}\left[\boldsymbol{\mathcal{R}}_{k+1}^{\text{data}}\right]$ becomes minimization of $\mathbf{tr}\left[\boldsymbol{\mathcal{S}}_{k+1}^{\text{data}}\right]$. $\qquad\square$

The above result simultaneously determines the Kalman gain $\boldsymbol{K}$ and the optimal noise in the data, for which upper bound on posterior variance is achieved.

Remark 10. The discussion so far has been on adding maximum synthetic noise to existing data, for which utility is achieved. This is relevant in situations where the collected data has multiple use with different accuracy needs. Thus, it is meaningful to sense at the highest accuracy and then add synthetic noise depending on accuracy needs. However, in some applications, it may be economical to determine the sensing accuracy directly, since higher accuracy is associated with higher cost. Theorem 16 can be modified to determine the optimal sensing precision for which the prescribed accuracy in the state estimate is achieved. It is given by the following result.

Corollary 1. *The minimum sensing precision that satisfies* $\mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^{+}\right] \leq \gamma_u$, *is given by the solution of the following optimization problem*

$$\min_{\boldsymbol{\lambda}\geq 0, \boldsymbol{Q}_u\geq 0, \boldsymbol{K}} \|\boldsymbol{\lambda}\|_1, \tag{4.31a}$$

subject to

$$\begin{bmatrix} \boldsymbol{Q}_u & \boldsymbol{M}_u(\boldsymbol{I}-\boldsymbol{KC})\sqrt{\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^{-}} & \boldsymbol{M}_u\boldsymbol{K} \\ \sqrt{\boldsymbol{\Sigma}_{\boldsymbol{xx},k+1}^{-}}(\boldsymbol{I}-\boldsymbol{KC})^T\boldsymbol{M}_u^T & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{K}^T\boldsymbol{M}_u^T & \boldsymbol{0} & \boldsymbol{\mathcal{S}}^{sensor} \end{bmatrix} \geq 0, \tag{4.31b}$$

$$\mathbf{tr}\left[\boldsymbol{Q}_u\right] \leq \gamma_u. \tag{4.31c}$$

where

$$\boldsymbol{S}^{sensor} := \boldsymbol{diag}(\boldsymbol{\lambda}),$$

and $\boldsymbol{\lambda}$ is the sensor precision which is the reciprocal of the sensor noise.

Proof. The proof is similar to theorem 16, with the sensor precision as a variable with a diagonal structure. □

Remark 11. Sparse Sensing: The l_1 cost in (4.31a) induces sparseness in the solution, i.e. many entries of the optimal $\boldsymbol{\lambda}$ are expected to be zero. These correspond to zero precision, implying that the corresponding sensor is not required to achieve the required accuracy in the state estimate, and can be eliminated from further consideration. From a system design perspective, this is quite useful. We can formulate the optimization problem with a dictionary of sensors, admitting redundancy in the sensing. The l_1 optimization will result in the optimal (possibly sparse) sensing precisions that will achieve the required accuracy in the state estimate.

Remark 12. It is possible that the user specifies multiple partitions of the augmented state $\boldsymbol{X}$ for utility, i.e. $\boldsymbol{X}_{u_1} := \boldsymbol{M}_{u_1}\boldsymbol{X}, \ldots, \boldsymbol{X}_{u_r} := \boldsymbol{M}_{u_r}\boldsymbol{X}$, with accuracy bounds

$$\mathbf{tr}\left[\boldsymbol{M}_{u_1}\boldsymbol{\Sigma}^+\boldsymbol{M}_{u_1}^T\right] \leq \gamma_{u_1}, \cdots, \mathbf{tr}\left[\boldsymbol{M}_{u_r}\boldsymbol{\Sigma}^+\boldsymbol{M}_{u_r}^T\right] \leq \gamma_{u_r}.$$

In such a case, each upper-bound constraint will add a pair of inequalities to the optimization problem, similar to (4.31b) and (4.31c), but in terms of variable $\boldsymbol{Q}_{u_i}$.

4.3.2   Minimum Noise Satisfying Privacy Constraints

We next present a convex optimization problem, which determines the *minimum* synthetic noise needed to satisfy the lower bound on $\Sigma^+_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}$. It is given by the following theorem.

Theorem 17. *The minimum noise that satisfies* $\mathbf{tr}\left[\Sigma^+_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}\right] \geq \gamma_p$, *is given by the solution of the following optimization problem*

$$\min_{\mathcal{R}^{data}_{k+1}\geq 0,\, \boldsymbol{Q}_p\geq 0} \mathbf{tr}\left[\mathcal{R}^{data}_{k+1}\right], \tag{4.32a}$$

such that,

$$\begin{bmatrix} \boldsymbol{M}_p\Sigma^-_{\boldsymbol{xx},k+1}\boldsymbol{M}_p^T - \boldsymbol{Q}_p & \boldsymbol{M}_p\Sigma^-_{\boldsymbol{xy},k+1} \\ \Sigma^{-T}_{\boldsymbol{xy},k+1}\boldsymbol{M}_p^T & \left(\Sigma^-_{\boldsymbol{yy},k+1} + \mathcal{R}^{sensor}_{k+1} + \mathcal{R}^{data}_{k+1}\right) \end{bmatrix} \geq 0, \tag{4.32b}$$

$$\mathbf{tr}\left[\boldsymbol{Q}_p\right] \geq \gamma_p. \tag{4.32c}$$

Proof. From (4.24), and $\mathbf{tr}\left[\Sigma^+_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}\right] \geq \gamma_p$ we get the following equivalent conditions

$$\boldsymbol{M}_p\Sigma^-_{\boldsymbol{xx},k+1}\boldsymbol{M}_p^T - \boldsymbol{Q}_p - \boldsymbol{M}_p\Sigma^-_{\boldsymbol{xy},k+1}\left(\Sigma^-_{\boldsymbol{yy},k+1} + \mathcal{R}^{sensor}_{k+1} + \mathcal{R}^{data}_{k+1}\right)^{-1}\Sigma^{-T}_{\boldsymbol{xy},k+1}\boldsymbol{M}_p^T \geq 0, \tag{4.33}$$

$$\mathbf{tr}\left[\boldsymbol{Q}_p\right] \geq \gamma_p. \tag{4.34}$$

Using Schur complement, we can write inequality in (4.33) as the LMI in (4.32b)

with respect to $\mathcal{R}_{k+1}^{\mathrm{data}}$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $\square$

Remark 13. Like in the utility case, it is possible that the user specifies multiple partitions of the augmented state $\boldsymbol{X}$ for privacy, i.e. $\boldsymbol{X}_{p_1} := \boldsymbol{M}_{p_1}\boldsymbol{X}, \ldots, \boldsymbol{X}_{p_q} := \boldsymbol{M}_{p_q}\boldsymbol{X}$, with privacy bounds

$$\mathbf{tr}\left[\boldsymbol{M}_{p_1}\boldsymbol{\Sigma}^+\boldsymbol{M}_{p_1}^T\right] \geq \gamma_{p_1}, \ \ldots, \ \mathbf{tr}\left[\boldsymbol{M}_{p_q}\boldsymbol{\Sigma}^+\boldsymbol{M}_{p_q}^T\right] \geq \gamma_{p_q}.$$

In such a case, each lower-bound constraint will add a pair of inequalities to the optimization problem, similar to (4.32b) and (4.32c), but in terms of variable $\boldsymbol{Q}_{p_i}$.

4.3.3 Optimal Privacy-Utility Tradeoff

The optimization problems in the previous two sections have addresses utility and privacy separately. In this section, we present a joint formulation for determining the optimal privacy-utility trade-off. We formulate the optimization problems around *two notions of the trade-off*.

4.3.3.1 Utility-aware privacy

The first notion is *utility-aware privacy*, where the utility is specified via hard constraint γ_u and the privacy is maximized. This results in the following optimization formulation: This results in the following formulation:

$$\max \ \gamma_p, \ \text{subject to} \ \mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}^+\right] \geq \gamma_p, \ \text{and} \ \mathbf{tr}\left[\boldsymbol{\Sigma}_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}^+\right] \leq \gamma_u, \qquad (4.35)$$

for a given γ_u. This can be generalized to multiple partitions of $\boldsymbol{X}$ for privacy and utility.

4.3.3.2   Privacy-aware utility

The second notion is *privacy-aware utility*, where the privacy is specified via hard constraint γ_p and the utility is maximized. This results in the following formulation:

$$\min \ \gamma_u, \text{ subject to } \mathbf{tr}\left[\boldsymbol{\Sigma}^+_{\boldsymbol{x}_p\boldsymbol{x}_p,k+1}\right] \geq \gamma_p, \text{ and } \mathbf{tr}\left[\boldsymbol{\Sigma}^+_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}\right] \leq \gamma_u, \tag{4.36}$$

for a given γ_p. This can also be generalized to multiple partitions of $\boldsymbol{X}$ for both privacy and utility.

The idea is to combine the optimization formulations from theorem 15 (or 16) and theorem 17 into a single formulation. However, theorems 15 (or 16) and 17 involve variables $\boldsymbol{\mathcal{S}}^{\text{data}}_{k+1}$ and $\boldsymbol{\mathcal{R}}^{\text{data}}_{k+1}$ that are constrained by $\boldsymbol{\mathcal{S}}^{\text{data}}_{k+1}\boldsymbol{\mathcal{R}}^{\text{data}}_{k+1} = \boldsymbol{I}_{n_y}$. which is non-convex. In this paper, we linearize this constraint about a known value of $\boldsymbol{\mathcal{S}}^{\text{data}}_{k+1}$ and $\boldsymbol{\mathcal{R}}^{\text{data}}_{k+1}$, and iteratively update it to arrive at a suboptimal solution. That is, initially we assume $\bar{\boldsymbol{\mathcal{S}}}^{\text{data}}_{k+1}$ and $\bar{\boldsymbol{\mathcal{R}}}^{\text{data}}_{k+1}$ are given and we write,

$$\boldsymbol{\mathcal{S}}^{\text{data}}_{k+1} := \bar{\boldsymbol{\mathcal{S}}}^{\text{data}}_{k+1} + \tilde{\boldsymbol{\mathcal{S}}}^{\text{data}}_{k+1} \geq 0, \text{ and} \tag{4.37}$$

$$\boldsymbol{\mathcal{R}}^{\text{data}}_{k+1} := \bar{\boldsymbol{\mathcal{R}}}^{\text{data}}_{k+1} + \tilde{\boldsymbol{\mathcal{R}}}^{\text{data}}_{k+1} \geq 0, \tag{4.38}$$

and linearize $\boldsymbol{\mathcal{S}}^{\text{data}}_{k+1}\boldsymbol{\mathcal{R}}^{\text{data}}_{k+1} = \boldsymbol{I}_{n_y}$ about $\bar{\boldsymbol{\mathcal{S}}}^{\text{data}}_{k+1}$ and $\bar{\boldsymbol{\mathcal{R}}}^{\text{data}}_{k+1}$, to get

$$\tilde{\boldsymbol{\mathcal{S}}}^{\text{data}}_{k+1}\bar{\boldsymbol{\mathcal{R}}}^{\text{data}}_{k+1} + \bar{\boldsymbol{\mathcal{S}}}^{\text{data}}_{k+1}\tilde{\boldsymbol{\mathcal{R}}}^{\text{data}}_{k+1} = 0, \tag{4.39}$$

and solve for $\tilde{\boldsymbol{\mathcal{S}}}^{\text{data}}_{k+1}$ and $\tilde{\boldsymbol{\mathcal{R}}}^{\text{data}}_{k+1}$.

We next present the complete optimization formulation for *utility-aware privacy*, formulated assuming linear sensing model. It can be generalized to nonlinear sensing model by formulating it around theorems 15 and 17.

Theorem 18. *The data noise that satisfies the utility-constraint* $\mathbf{tr}\left[\Sigma^+_{\boldsymbol{x}_u\boldsymbol{x}_u,k+1}\right] \leq \gamma_u$ *for a given* γ_u, *and maximizes privacy, is given by the solution of the following optimization problem:*

$$\max_{\boldsymbol{Q}_p, \boldsymbol{Q}_u, \tilde{\boldsymbol{S}}^{data}_{k+1}, \tilde{\boldsymbol{\mathcal{R}}}^{data}_{k+1}, \boldsymbol{K}, \gamma_p} \gamma_p \tag{4.40a}$$

subject to

$$\begin{bmatrix} \boldsymbol{Q}_u & \boldsymbol{M}_u(\boldsymbol{I}-\boldsymbol{K}\boldsymbol{C})\sqrt{\Sigma^-_{\boldsymbol{xx},k+1}} & \boldsymbol{M}_u\boldsymbol{K} & \boldsymbol{M}_u\boldsymbol{K} \\ \sqrt{\Sigma^-_{\boldsymbol{xx},k+1}}(\boldsymbol{I}-\boldsymbol{K}\boldsymbol{C})^T\boldsymbol{M}_u^T & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{K}^T\boldsymbol{M}_u^T & \boldsymbol{0} & \left(\boldsymbol{\mathcal{R}}^{sensor}_{k+1}\right)^{-1} & \boldsymbol{0} \\ \boldsymbol{K}^T\boldsymbol{M}_u^T & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{S}^{data}_{k+1} \end{bmatrix} \geq 0, \tag{4.40b}$$

$$\begin{bmatrix} \boldsymbol{M}_p\Sigma^-_{\boldsymbol{xx},k+1}\boldsymbol{M}_p^T - \boldsymbol{Q}_p & \boldsymbol{M}_p\Sigma^-_{\boldsymbol{xx},k+1}\boldsymbol{C}^T \\ \boldsymbol{C}\Sigma^-_{\boldsymbol{xx},k+1}\boldsymbol{M}_p^T & \left(\boldsymbol{C}\Sigma^-_{\boldsymbol{xx},k+1}\boldsymbol{C}^T + \boldsymbol{\mathcal{R}}^{sensor}_{k+1} + \boldsymbol{\mathcal{R}}^{data}_{k+1}\right) \end{bmatrix} \geq 0, \tag{4.40c}$$

$$\mathbf{tr}\left[\boldsymbol{Q}_u\right] \leq \gamma_u, \tag{4.40d}$$

$$\mathbf{tr}\left[\boldsymbol{Q}_p\right] \geq \gamma_p, \tag{4.40e}$$

$$\boldsymbol{S}^{data}_{k+1} := \bar{\boldsymbol{S}}^{data}_{k+1} + \tilde{\boldsymbol{S}}^{data}_{k+1} \geq 0, \tag{4.40f}$$

$$\boldsymbol{\mathcal{R}}^{data}_{k+1} := \bar{\boldsymbol{\mathcal{R}}}^{data}_{k+1} + \tilde{\boldsymbol{\mathcal{R}}}^{data}_{k+1} \geq 0, \tag{4.40g}$$

$$\tilde{\boldsymbol{S}}^{data}_{k+1}\bar{\boldsymbol{\mathcal{R}}}^{data}_{k+1} + \bar{\boldsymbol{S}}^{data}_{k+1}\tilde{\boldsymbol{\mathcal{R}}}^{data}_{k+1} = 0. \tag{4.40h}$$

Proof. Builds on the proof for theorems 16 and 17, along with linearization of the nonconvex constraint $\boldsymbol{\mathcal{S}}_{k+1}^{\text{data}} \boldsymbol{\mathcal{R}}_{k+1}^{\text{data}} = \boldsymbol{I}_{n_y}$.

$\square$

Remark 14. The optimization for privacy-aware utility can be formulated by replacing (4.40a) with

$$\min_{\boldsymbol{Q}_p, \boldsymbol{Q}_u, \tilde{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}}, \tilde{\boldsymbol{\mathcal{R}}}_{k+1}^{\text{data}}, \boldsymbol{K}, \gamma_u} \gamma_u \tag{4.41}$$

where γ_p is user specified.

Remark 15. Optimization problem (4.40) can also be generalized to multiple partitions of $\boldsymbol{X}$ for both privacy and utility, by introducing new variables $\{\boldsymbol{Q}_{p_i}\}$, $\{\boldsymbol{Q}_{u_j}\}$, and $\{\gamma_{p_i}\}$ or $\{\gamma_{u_j}\}$.

Optimization in (4.40), is solved repeatedly with updated values of $\bar{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}}$ and $\bar{\boldsymbol{\mathcal{R}}}_{k+1}^{\text{data}}$, until there is no significant change in the cost function. For the next iteration, $\bar{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}}$ and $\bar{\boldsymbol{\mathcal{R}}}_{k+1}^{\text{data}}$ are updated with the optimal $\tilde{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}}$ and $\tilde{\boldsymbol{\mathcal{R}}}_{k+1}^{\text{data}}$. The update however, is slightly different for utility-aware privacy and privacy-aware utility. For utility-aware privacy, we must ensure utility constraints are satisfied. Since the optimization in (4.40) satisfies this constraint with $\boldsymbol{\mathcal{S}}_{k+1}^{\text{data}} := \bar{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}} + \tilde{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}}$, we update $\bar{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}}$ and $\bar{\boldsymbol{\mathcal{R}}}_{k+1}^{\text{data}}$ as

$$\bar{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}} := \bar{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}} + \tilde{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}}, \text{ and} \tag{4.42a}$$

$$\bar{\boldsymbol{\mathcal{R}}}_{k+1}^{\text{data}} := \left(\bar{\boldsymbol{\mathcal{S}}}_{k+1}^{\text{data}} \right)^{-1} . \tag{4.42b}$$

For privacy-aware utility, we must ensure privacy constraints are satisfied. Since the optimization (see remark 14) satisfies this constraint with $\mathcal{R}_{k+1}^{\text{data}} := \bar{\mathcal{R}}_{k+1}^{\text{data}} + \tilde{\mathcal{R}}_{k+1}^{\text{data}}$, we update $\bar{\mathcal{S}}_{k+1}^{\text{data}}$ and $\bar{\mathcal{R}}_{k+1}^{\text{data}}$ for this case, as

$$\bar{\mathcal{R}}_{k+1}^{\text{data}} := \bar{\mathcal{R}}_{k+1}^{\text{data}} + \tilde{\mathcal{R}}_{k+1}^{\text{data}}, \text{ and} \qquad (4.43a)$$

$$\bar{\mathcal{S}}_{k+1}^{\text{data}} := \left(\bar{\mathcal{R}}_{k+1}^{\text{data}}\right)^{-1}. \qquad (4.43b)$$

Pseudocode for implementing the above iterative algorithm for utility-aware privacy is presented in Algorithm 3. The algorithm for privacy-aware utility can be similarly developed by incorporating the changes from remark 14 and (4.43), in Algorithm 3.

Algorithm 3 Algorithm for **utility-aware** privacy

Initialize: $\bar{\mathcal{S}}_{k+1}^{\text{data}} := \boldsymbol{I}_{n_y}$, $\bar{\mathcal{R}}_{k+1}^{\text{data}} := \boldsymbol{I}_{n_y}$ # Any other initialization may work

Initialize: $\gamma_{p_{\text{old}}} := 1e^{10}$ # Something large

Initialize: $\epsilon := 10^{-3}$ # Tolerance

Initialize: `done := false`

`while done == false`

    Solve optimization problem in (4.40) to get γ_p, $\tilde{\mathcal{S}}_{k+1}^{\text{data}}$ and $\tilde{\mathcal{R}}_{k+1}^{\text{data}}$

    `if` $|\gamma_p - \gamma_{p_{\text{old}}}| \leq \epsilon$

        `done := true`

    `else`

        $\gamma_{p_{\text{old}}} := \gamma_p$

        Update $\bar{\mathcal{S}}_{k+1}^{\text{data}}$ and $\bar{\mathcal{R}}_{k+1}^{\text{data}}$ using (4.42)

    `end`

`end`

4.4 Numerical Simulation

In this section, we apply the proposed algorithms for tracking the International Space Station (ISS), with its orbit defined by the following TLE set:

```
ISS (ZARYA)
1 25544U 98067A 19248.67387091 0.00001921 00000-0 41082-4 0 9997
2 25544 51.6464 322.0340 0007976 9.5374 121.4565 15.50435809187740
```

The orbit is propagated in Keplerian coordinates $(a, e, i, \Omega, \omega, f)$, with uncertain initial conditions and $J4$ perturbation. Initial condition uncertainty is assumed to be only in the semi-major axis (a). It is modeled as a Gaussian random variable with mean defined by the TLE set and standard-deviation equal to 1% of the mean. This uncertainty is represented with 100 samples.

Since the algorithms proposed here assume Gaussian uncertainty (EnKF and UKF), we transform the 6 dimensional orbit data from Keplerian coordinates $(a, e, i, \Omega, \omega, f)$ to Cartesian coordinates (x, y, z, u, v, w), and investigate the privacy-utility tradeoff in this representation. From fig.(4.1), we see that 1% uncertainty in a causes significant increase in the state uncertainty after only one orbit, and therefore it provides a rich enough data set for investigating privacy-utility tradeoff.

We also assume that we can sense (x, y, z), which gives us a *linear measurement model*. Consequently, we apply the corresponding algorithms to demonstrate the privacy-utility tradeoff. The orbit data is generated for 6000 seconds, at 1 second intervals. This captures one orbit of the ISS. We extract (x, y, z) at arbitrarily chosen times $0, 1600, 1900, 3400, 5100$ seconds, and treat them as measurements. These times are also expressed as $0, 0.27\, T_{\text{orb}}, 0.32\, T_{\text{orb}}, 0.57\, T_{\text{orb}}, 0.85\, T_{\text{orb}}$, where $T_{\text{orb}} := 6000$ sec-
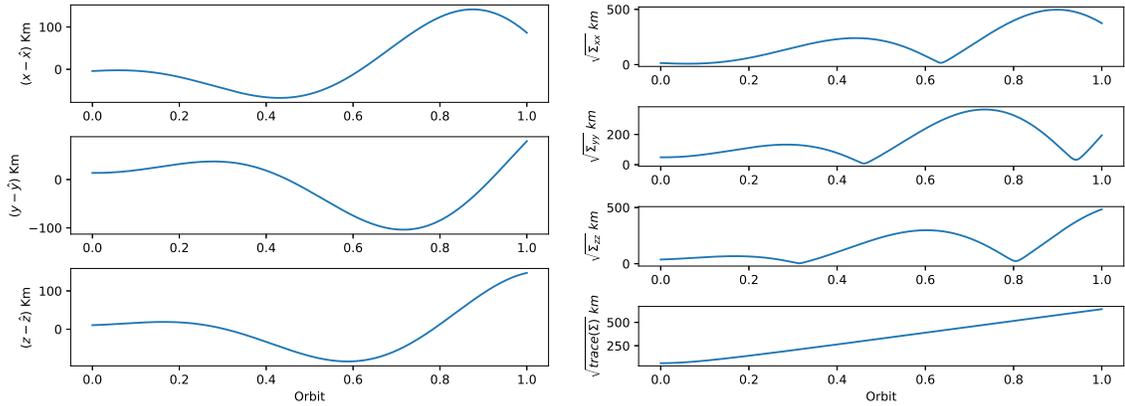
Figure 4.1: Growth in positional uncertainty due to 1% uncertainty in the initial semi-major axis. Reprinted with permission from [17].

onds is the time for one orbit. The locations of these measurements, or the sensing sites, are shown in fig.(4.2a). Using these five sensing sites, the objective of this example is to demonstrate:

1. the optimal sensor precisions that achieve the given utility in the state estimate,

2. minimum synthetic noise in the sensed data that achieves the prescribed privacy,

3. the optimal sensor precisions that achieve utility-aware privacy, i.e. maximize privacy with given utility constraints.

These studies are performed with respect to the position of the ISS, and are presented next.

(a) Sensing sites and orbit locations where accurate state estimates are required. Reprinted with permission from [17].

(b) Optimal sensor precisions: $0, 0, 0, 0, 0.94$. Reprinted with permission from [17].

(c) Posterior variance satisfying the required estimation accuracy. Reprinted with permission from [17].
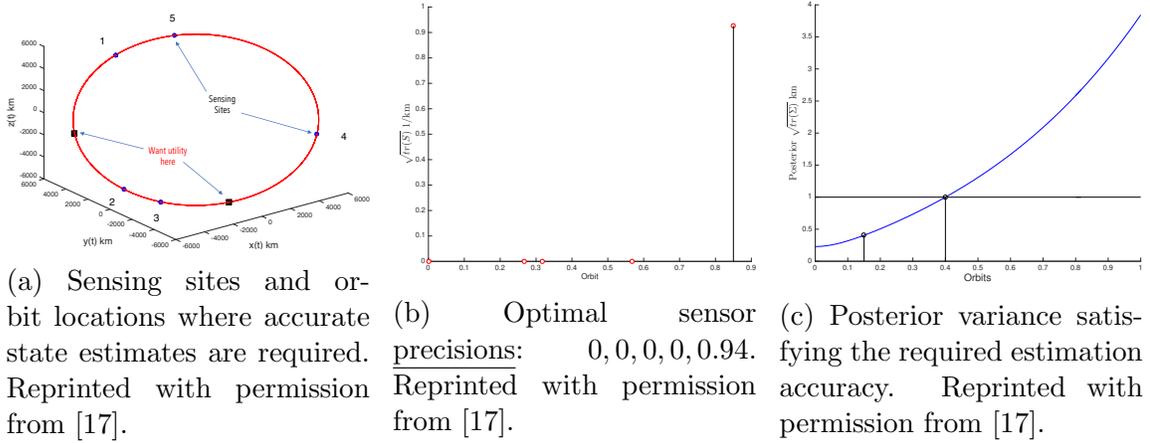
Figure 4.2: Optimal sensing precision satisfying utility constraints only. Reprinted with permission from [17].

4.4.1 Minimum Precision Guaranteeing Required Utility

Here we determine the precisions of sensors for which the utility constraints are satisfied. This is achieved by applying the optimization problem formulation from corollary 1. Fig.(4.2a) shows the location of the sensing sites, along with the locations where the utility constraints are to be satisfied. These are imposed at times $0.15\,T_{\mathrm{orb}}$, and $0.4\,T_{\mathrm{orb}}$. Recall that utility constraints are defined by (4.27), and for this example we choose $\gamma_{u_i} := 1$ km, i.e. the utility constraints are $\mathbf{tr}\left[\boldsymbol{M}_{u_i}\boldsymbol{\Sigma}^+\boldsymbol{M}_{u_i}^T\right] \leq 1$. Matrices $\boldsymbol{M}_{u_i}$ are mask matrices, which extract the variance of the error in (x, y, z) estimates, at times $0.15\,T_{\mathrm{orb}}$ and $0.4\,T_{\mathrm{orb}}$ respectively. Fig.(4.2b) shows the summation of the (x, y, z) sensor precisions, at each of the five sensing sites, for which the utility constraints are satisfied. This is confirmed by fig.(4.2c).

From fig.(4.2b), we see that minimization of the l_1 norm in (4.31) results in a sparse sensing architecture, i.e. many of the sensor precisions are zero. Therefore, only data from the sensing site #5, with the indicated precision, is required to satisfy the

150

utility constraint. There are three implications of this result. Firstly, from a sensing perspective, only sensors at site #5 are sufficient to track the object at the specified locations with the required accuracy. This helps in sensor allocations for space object tracking. Secondly, the optimal sensing precision can be used to optimize the energy used in the radar/laser based sensing. Finally, from a data sharing perspective, if data is available from all the sensing sites – only data from location #5 needs to be shared. If the accuracy of the sensed data is more than required, it can be corrupted by noise defined by the reciprocal of the precision value. This will protect the economic value of the data [31].

We also observe that data from the future is used to satisfy the utility constraints in the past, resulting in optimal smoothing. Since the optimization is formulated as a batch estimation, this is possible. The results are consistent with the fact that optimal smoothers generally achieve lower mean-square error than optimal filters [5]. This allows utility constraints in the past to be satisfied with least sensor precisions.

4.4.2   Minimum Noise Guaranteeing Required Privacy

In this section, we apply the optimization problem from theorem 17 to determine the minimal noise in the sensor data for which the errors in the state estimates are above a prescribed value, at the specified location in the orbit. The location where privacy is required is shown in fig.(4.3a), which corresponds to time $0.82\,T_{\mathrm{orb}}$. The privacy constraint is defined by (4.27) with $\gamma_p := 5.17^2$. This is determined by requiring $\mathbf{tr}\left[\boldsymbol{M}_p\boldsymbol{\Sigma}^+\boldsymbol{M}_p^T\right] \geq 10^{-4}\mathbf{tr}\left[\boldsymbol{M}_p\boldsymbol{\Sigma}^-\boldsymbol{M}_p^T\right]$, where $\boldsymbol{M}_p$ extracts the variance of the error in (x, y, z) estimates at time $0.82\,T_{\mathrm{orb}}$. The sensing sites are the same as the above problem.

Fig.(4.3b) shows the summation of the (x, y, z) sensor noise variances, at each of the five sensing sites, for which the privacy constraints are satisfied. We see that the minimum noise, for which privacy is guaranteed, has an increasing trend, with higher noise in the vicinity of the location where privacy is required. Fig.(4.3c) shows $\sqrt{\mathbf{tr}\left[\boldsymbol{\Sigma}^+\right]}$, which satisfies the privacy constraints at $t = 0.82\,T_{\mathrm{orb}}$, i.e. $\sqrt{\mathbf{tr}\left[\boldsymbol{M}_p\boldsymbol{\Sigma}^+\boldsymbol{M}_p^T\right]} \geq 5.17$. Therefore, from a data-sharing perspective, sensor data from the 5 sites should be corrupted with synthetic noise – defined by the optimal noise variances in fig.(4.3b), to ensure the required privacy.
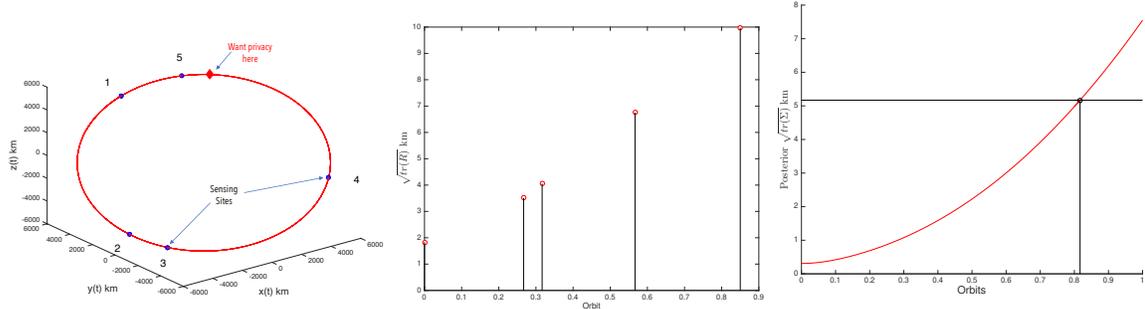
(a) Orbit location where privacy is required. Reprinted with permission from [17].

(b) Optimal sensor noise with l_1 minimization. Reprinted with permission from [17].

(c) Posterior variance satisfying the required privacy constraint. Reprinted with permission from [17].

Figure 4.3: Optimal sensor noise for only privacy. Reprinted with permission from [17].

4.4.3 Utility-Aware Maximum Privacy

We next present results obtained by applying Algorithm 3 to the ISS data set, where privacy is maximized with utility constraints. The locations of sensors, the utility constraints, and the location where privacy is required, are the same as in the above studies. Fig.(4.4b) shows the sensor precisions for which the utility constraints (same

as those in §4.4.1) are satisfied and privacy is maximized. The resulting posterior is shown in fig.(4.4c). We observe that utility constraints are satisfied at times $0.15\,T_{\mathrm{orb}}$, and $0.4\,T_{\mathrm{orb}}$, while privacy is maximized at time $0.82\,T_{\mathrm{orb}}$. Fig.(4.4c) also shows the case when privacy is not maximized, but only utility is satisfied with minimum precision, i.e. the case discussed in §4.4.1. We see that utility-aware maximum privacy is able to achieve about 1.63 times more privacy. The value of $\sqrt{\mathbf{tr}\left[\boldsymbol{M}_p\boldsymbol{\Sigma}^+\boldsymbol{M}_p^T\right]}$ in §4.4.1 is 2.26, whereas $\sqrt{\mathbf{tr}\left[\boldsymbol{M}_p\boldsymbol{\Sigma}^+\boldsymbol{M}_p^T\right]}$ in this case is 4.35. Thus, we can see the benefit Algorithm 3.

(a) Orbit locations where utility is constrained, and location where privacy is maximized. Reprinted with permission from [17].

(b) Sensor precisions for maximum utility-aware privacy: $2.30, 1.82, 1.51, 31.62, 0.29$. Reprinted with permission from [17].

(c) Red: Posterior variance for maximum utility-aware privacy. Blue: fig.(4.2c) shown again. Reprinted with permission from [17].
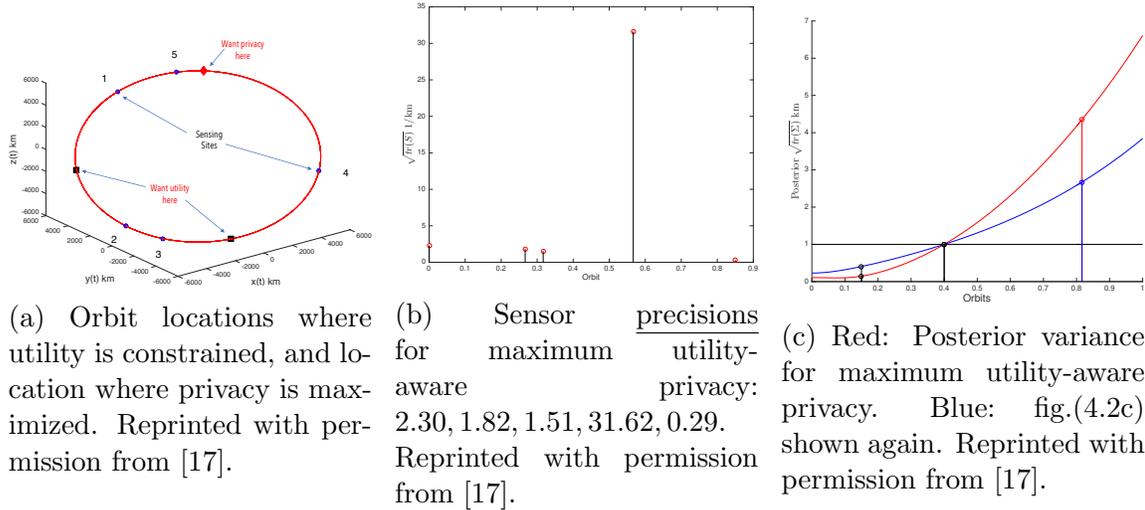
Figure 4.4: Optimal sensor precision for utility-aware privacy over one orbit of the ISS. Reprinted with permission from [17].

Fig.(4.4b) shows that the utility constraints are satisfied using most precise data from the future (i.e. site #4) , which implies that the optimization primarily applies smoothing to satisfy the utility constraints. This is similar to the results from fig.(4.2b), where smoothing was used to satisfy the utility constraints.

We also see that privacy is maximized by reducing the precision at site #5. Data from site #5 is from the future, and incorporating this data would reduce uncertainty at all past locations due to smoothing. This would not maximize privacy. Low precision data from site #5 essentially means that no update is made to the prior beyond the data from the sensing site #4, and this maximizes the privacy. Consequently, data from site #4 plays an important role. It is used to satisfy the uncertainty constraints at the utility sites using smoothing, and maximize uncertainty at the privacy site using prediction.

Remark 16. The examples above, are shown with data from one orbit, where privacy is maximized after the utility. This is a simpler scenario, because state uncertainty grows without update and hence it is easier to achieve higher uncertainty, and hence privacy, at later times.

In the next example, we consider an interesting scenario where privacy is maximized in between two locations where utility is constrained. For this scenario, we consider data from *five* orbits of the ISS, with data saved every 100 seconds and measurements available at arbitrarily chosen times 0, $0.15\,T_{\mathrm{orb}}$, $0.82\,T_{\mathrm{orb}}$, $1.65\,T_{\mathrm{orb}}$, $3.32\,T_{\mathrm{orb}}$, $4.15\,T_{\mathrm{orb}}$, and $4.98\,T_{\mathrm{orb}}$. Therefore, there are 7 sensing sites, measuring (x, y, z). The utility constraints are imposed at times $0.48\,T_{\mathrm{orb}}$ and $4.82\,T_{\mathrm{orb}}$, with $\mathbf{tr}\left[\boldsymbol{M}_{u_i}\boldsymbol{\Sigma}^+\boldsymbol{M}_{u_i}^T\right] \leq 1$. Privacy is maximized at the time $2.48\,T_{\mathrm{orb}}$. Therefore, privacy is maximized in between the two times where utility is constrained. The privacy and utility time points are chosen arbitrarily.

Fig.(4.5) shows the results from the utility-aware privacy optimization for this case. In fig.(4.5a), the optimal sensor precisions that satisfy the utility constraints are shown. We observe that data from sites #1 to #5 are not required to satisfy the

(a) Minimum precision satisfying only utility constraints: 0, 0, 0, 0, 0, 1.29, 2.14. Reprinted with permission from [17].

(b) Sensor precisions for maximum utility-aware privacy: 0.93, 0.61, 0.33, 0.13, 0.17, 0.6, 44.23. Reprinted with permission from [17].

(c) Red: Posterior variance for maximum utility-aware privacy. Blue: Posterior variance satisfying utility constraints only. Reprinted with permission from [17].

Figure 4.5: Optimal sensor precision for utility-aware privacy over <u>five</u> orbits of the ISS. Reprinted with permission from [17].

utility constraints. Only data from sites #6 and #7 are required, with the given precisions. The corresponding $\sqrt{\mathbf{tr}\left[\boldsymbol{\Sigma}^+\right]}$ is shown in fig.(4.5c) in blue, and we can see that the utility constraints are satisfied at times $0.48\,T_{\mathrm{orb}}$ and $4.82\,T_{\mathrm{orb}}$. Achieving the utility with minimum precision implicitly achieves a certain level of privacy at time $2.48\,T$, as we can see $\sqrt{\mathbf{tr}\left[\boldsymbol{\Sigma}^+\right]}$ increases between times $0.48\,T$ and $4.82\,T$. Fig.(4.5c) also shows in red, $\sqrt{\mathbf{tr}\left[\boldsymbol{\Sigma}^+\right]}$ from utility-aware privacy maximization. We can see that it achieves significantly higher privacy at time $2.48\,T$. The value of $\sqrt{\mathbf{tr}\left[\boldsymbol{M}_p\boldsymbol{\Sigma}^+\boldsymbol{M}_p^T\right]}$ from utility-only optimization is 2.67, and the value of $\sqrt{\mathbf{tr}\left[\boldsymbol{M}_p\boldsymbol{\Sigma}^+\boldsymbol{M}_p^T\right]}$ from utility-aware privacy maximization is 4.45, resulting in an improvement by a factor of 1.67. The corresponding sensor precisions are shown in fig.(4.5b). Therefore, for this example too, we see that Algorithm 3 is able to achieve higher privacy, while satisfying the utility constraints.

We next discuss the convergence property of Algorithm 3. For the results shown in fig.(4.4) and fig.(4.5), the respective convergences are shown in fig.(4.6), where

155

$$\Delta J := |\gamma_p - \gamma_{p_{\text{old}}}|.$$
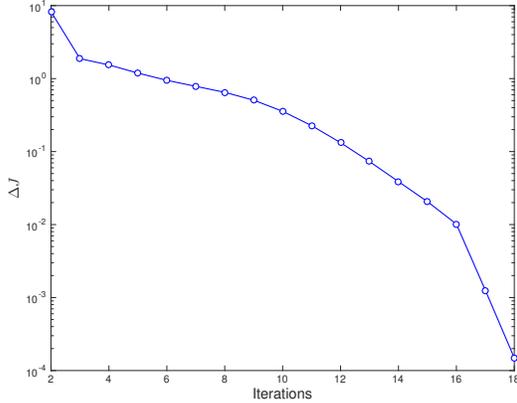
(a) Convergence of Algorithm 3 for the 1-Orbit problem shown in fig.(4.4). Reprinted with permission from [17].
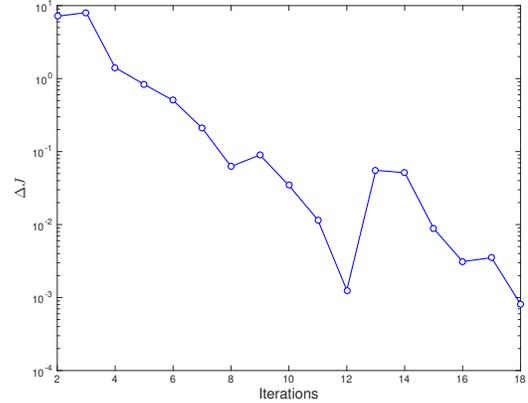
(b) Convergence of Algorithm 3 for the 5-Orbit problem shown in fig.(4.5). Reprinted with permission from [17].

Figure 4.6: Convergence of Algorithm 3. Reprinted with permission from [17].

Empirically, we observe that in both the cases the algorithm convergences to the optimal solution within 18 iterations, for $\epsilon := 10^{-3}$. The associated computational times are summarized in table 4.1. For every iteration, the optimization problem for the 1-orbit problem takes an average of 3.5 seconds. Whereas, for the 5-orbit problem, optimizations take an average of 235.93 seconds, which is higher due to larger problem size. The computational times for the utility-only optimization are also shown in the table. We observe that for 1 and 5 orbit problems, the computational times are 0.5 and 28.26 seconds respectively. These times are obtained in MATLAB, with YALMIP [52] as the parser and MOSEK [3] as the solver, executing in a Mac Book Pro with 2.3 GHz Quad-Core Intel Core i7 processor and 16 GB (1600 MHz DDR3) memory.

	1-Orbit Problem	5-Orbit Problem
Utility-aware Privacy	3.5	235.93
Utility-Only	0.5	28.26

Table 4.1: Average computational times (sec) for 1 and 5 orbit problems. These times are obtained in MATLAB, with YALMIP [52] as the parser and MOSEK [3] as the solver. Reprinted with permission from [17].

Therefore, the utility-only optimization results in significantly faster solution, but the privacy is not maximized. Whereas, with the utility-aware privacy maximization, we are able to achieve higher privacy but with significantly higher computational time. Since both the formulations solve a semi-definite program, the computational times are expected to grow polynomially using interior-point method. Therefore, for large-scale problems, specialized methods (such as first order methods) can be applied to solve the optimization problem.

4.5 Conclusion & Summary

In this paper we presented a new formulation that addresses optimal privacy vs utility tradeoff in the Ensemble/Unscented Kalman filtering framework. Privacy and utility are defined in terms of the estimation error variances. The formulation achieves this tradeoff by injecting synthetic noise to the sensor data, which is used to regulate the posterior error covariance.

We demonstrated how this formulation can be applied to achieve the tradeoff in the context of space-object tracking problem. We were able to show that it is possible to satisfy utility (defined by upper bounding the estimation error), while achieving a certain level of privacy (defined by lower-bounding the estimation error).

In particular, the presented results demonstrated that:

1. Utility upper-bounds can be satisfied with sparse sensing, which also indirectly helps in sensor scheduling. This is achieved by satisfying the utility constraints with maximum sensor noise or minimum sensor precision, where the precision is defined to be the inverse of noise. Sparseness in sensing is achieved by l_1 minimization.

2. Privacy lower-bounds can be satisfied by maximizing the noise in the sensor.

3. A joint optimization problem for utility-aware privacy is able to maximize privacy (i.e. maximize the lower bound), while satisfying utility upper bound. Similarly, a joint-optimization can maximize utility (i.e. minimize the upper bound), while satisfying the privacy lower bound.

These results have significant implications from a data sharing perspective. The above optimization problems determine the level of synthetic noise that should be added to the raw sensed data, such that the desired privacy-utility tradeoff is achieved. This is important, because privacy concerns can result in conservative data obfuscation and severely impede utility. On the other hand, utility concerns can result in excessive sensing accuracy, which can violate privacy concerns and perhaps be uneconomical from a system design and operations perspective. Therefore, we expect our framework to enable better data collection and sharing policy, particularly for the SSA community. In addition, since SSA data is being commoditized, the proposed algorithms will enable data-products with different levels of accuracy, corresponding to various stratified pricing models for future value-added services in space-data analytics.

Acknowledgment

5. SUMMARY AND FUTURE WORKS

In this work we presented a novel formulation that addresses optimal privacy vs utility tradeoff in the Kalman & Ensemble/Unscented Kalman filtering framework. We defined privacy and utility in terms of the estimation error variances. The formulation that we presented satisfies this tradeoff by injecting synthetic noise to the sensor data, which regulates the posterior error covariance. Mainly in the context of space-object tracking problem, we demonstrated how this formulation can be applied to achieve the privacy-utility tradeoff . We were able to show that it is possible to satisfy utility (defined by upper bounding the estimation error), while achieving a certain level of privacy (defined by lower-bounding the estimation error). The problem of sensor precision selection can be extended to selecting the optimal locations of sensors, sensor types, and sensor noise characteristics, for making measurements of some dynamic spatio-temporal process. When designing sensing strategies, the aim is usually to find the design which minimizes expected post-experimental or data-collection uncertainties on the model parameters or system states or function of system states. Classical design methods are inefficient in non-linear problems because they assume linear design criteria. These inaccurate assumptions include linear mapping from parameter or states to measurement space along with Gaussian assumptions on the priors and measurement noises. Bayesian design of experiments (BDE) [68] or data collection is a framework developed to incorporate some form of available prior information in the design stage of measurement process. Although the design stage of the measurement process or sensing involves various parameters, we are particularly interested in that subspace that is spanned by the parameters related to the measurement model, such as sensor type, sensor noise characteristics,

sensor location and sensor set cardinality. If we consider the problem of sensor design for multiple time steps together, we might need to add sensing schedule to the set of unknown design parameters. Algorithms for generic sensing architecture design can be addressed along the lines of Bayesian design of experiments.

REFERENCES

[1] Meteosat-8 [MSG-1]. `http://sat-nd.com/failures/msg1.html`.

[2] Russian Telecom Satellite Fails After Sudden Impact. `https://tinyurl.com/qcpd8`.

[3] Erling D Andersen and Knud D Andersen. The mosek interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In *High performance optimization*, pages 197–232. Springer, 2000.

[4] Brian DO Anderson and John B Moore. Optimal filtering. *Englewood Cliffs*, 21:22–95, 1979.

[5] Brian DO Anderson and John B Moore. *Optimal filtering*. Courier Corporation, 2012.

[6] Bhargav Appasani and Dusmanta Kumar Mohanta. Optimal placement of synchrophasor sensors for risk hedging in a smart grid. *IEEE Sensors Journal*, 17(23):7857–7865, dec 2017.

[7] Robert R Bitmead, Michel R Gevers, Ian R Petersen, and R John Kaye. Monotonicity and stabilizability-properties of solutions of the riccati difference equation: Propositions, lemmas, theorems, fallacious conjectures and counterexamples. *Systems & Control Letters*, 5(5):309–315, 1985.

[8] Claudio J Bordin and Marcelo GS Bruno. Particle filtering on the complex stiefel manifold with application to subspace tracking. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5485–5489. IEEE, 2020.

[9] R. Brockett. Stabilization of motor networks. In *Proceedings of 1995 34th IEEE Conference on Decision and Control*. IEEE, 1995.

[10] Arthur Earl Bryson. *Applied optimal control: optimization, estimation and control*. Routledge, 2018.

[11] Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted l_1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905, 2008.

[12] Xudong Chen, Mohamed-Ali Belabbas, and Tamer Başar. Optimal capacity allocation for sampled networked systems. *Automatica*, 85:100–112, nov 2017.

[13] Sundeep Prabhakar Chepuri and Geert Leus. Sparsity-promoting adaptive sensor selection for non-linear filtering. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, may 2014.

[14] Sundeep Prabhakar Chepuri and Geert Leus. Sparsity-promoting sensor selection for non-linear measurement models. *IEEE Transactions on Signal Processing*, 63(3):684–698, feb 2015.

[15] Jorge Cortés, Geir E Dullerud, Shuo Han, Jerome Le Ny, Sayan Mitra, and George J Pappas. Differential privacy in control and network systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 4252–4272. IEEE, 2016.

[16] Andre F. C. da Silva and Tim Colonius. Ensemble-based state estimator for aerodynamic flows. *AIAA Journal*, 56(7):2568–2578, jul 2018.

[17] Niladri Das and Raktim Bhattacharya. Privacy and utility aware data sharing for space situational awareness from ensemble and unscented kalman filtering perspective, 2019.

[18] Niladri Das and Raktim Bhattacharya. Eigen value analysis in lower bounding uncertainty of kalman filter estimates, 2020.

[19] Niladri Das and Raktim Bhattacharya. Optimal sensing precision in ensemble and unscented kalman filtering, 2020.

[20] Niladri Das, Vedang Deshpande, and Raktim Bhattacharya. Optimal-transport-based tracking of space objects using range data from a single ranging station. *Journal of Guidance, Control, and Dynamics*, pages 1–13, feb 2019.

[21] Neil K. Dhingra, Mihailo R. Jovanovic, and Zhi-Quan Luo. An ADMM algorithm for optimal sensor and actuator selection. In *53rd IEEE Conference on Decision and Control*. IEEE, dec 2014.

[22] Cynthia Dwork. Differential privacy. *Encyclopedia of Cryptography and Security*, pages 338–340, 2011.

[23] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[24] Yonina C Eldar and Gitta Kutyniok. *Compressed sensing: theory and applications*. Cambridge university press, 2012.

[25] Geir Evensen. The ensemble kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53(4):343–367, nov 2003.

[26] Geir Evensen. The ensemble kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics*, 53(4):343–367, 2003.

[27] Geir Evensen and Peter Jan Van Leeuwen. Assimilation of geosat altimeter data for the agulhas current using the ensemble kalman filter with a quasigeostrophic model. *Monthly Weather Review*, 124(1):85–96, 1996.

[28] Farhad Farokhi and Henrik Sandberg. Ensuring privacy with constrained additive noise by minimizing fisher information. *Automatica*, 99:275–288, jan 2019.

[29] Jeff Foust. Spacex's space-internet woes: Despite technical glitches, the company plans to launch the first of nearly 12,000 satellites in 2019. *IEEE Spectrum*, 56(1):50–51, 2018.

[30] Jeff Foust. Esa spacecraft dodges potential collision with starlink satellite.

SPACENEWS, Sep 2019.

[31] Nate Frierson. An economic goods analysis of us space situational awareness (ssa) policy.

[32] Arpita Ghosh and Robert Kleinberg. Inferential privacy guarantees for differentially private mechanisms. *arXiv preprint arXiv:1603.01508*, 2016.

[33] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, March 2014.

[34] Carlos Guestrin, Andreas Krause, and Ajit Paul Singh. Near-optimal sensor placements in gaussian processes. In *Proceedings of the 22nd international conference on Machine learning - ICML 05*. ACM Press, 2005.

[35] Vijay Gupta, Timothy H. Chung, Babak Hassibi, and Richard M. Murray. On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42(2):251–260, feb 2006.

[36] Abhishek Halder and Raktim Bhattacharya. Dispersion analysis in hypersonic flight during planetary entry using stochastic liouville equation. *Journal of Guidance, Control, and Dynamics*, 34(2):459–474, 2011.

[37] Duo Han, Junfeng Wu, Huanshui Zhang, and Ling Shi. Optimal sensor scheduling for multiple linear dynamical systems. *Automatica*, 75:260–270, jan 2017.

[38] Todd Harrison, Kaitlyn Johnson, and Thomas G Roberts. *Space Threat Assessment 2018*. Center for Strategic & International Studies, 2018.

[39] Ying He and Edwin K.P. Chong. Sensor scheduling for target tracking: A monte carlo sampling approach. *Digital Signal Processing*, 16(5):533–545, sep 2006.

[40] Jeff Hecht. A "star wars" sequel? the allure of directed energy for space weapons. *Bulletin of the Atomic Scientists*, 75(4):162–170, 2019.

[41] White House. Space Policy Directive-3, National Space Traffic Management Policy . `https://www.whitehouse.gov/presidential-actions/`

```
space-policy-directive-3-national-space-traffic-management-policy/.
```

[42] Syed Talha Jawaid and Stephen L. Smith. Submodularity and greedy algorithms in sensor scheduling for linear dynamical systems. *Automatica*, 61:282–288, nov 2015.

[43] Nicholas L Johnson, Eugene Stansbery, David O Whitlock, Kira J Abercromby, and Debra Shoots. History of on-orbit satellite fragmentations. *NASA/TM 2008 214779*, 2008.

[44] S. Joshi and S. Boyd. Sensor selection via convex optimization. *IEEE Transactions on Signal Processing*, 57(2):451–462, feb 2009.

[45] Simon J Julier and Jeffrey K Uhlmann. New extension of the kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–193. International Society for Optics and Photonics, 1997.

[46] Yu Kawano and Ming Cao. Differential privacy and qualitative privacy analysis for nonlinear dynamical systems. *IFAC-PapersOnLine*, 51(23):52–57, 2018.

[47] Fragkiskos Koufogiannis and George J Pappas. Differential privacy for dynamical sensitive data. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 1118–1125. IEEE, 2017.

[48] Bhavya Lal, Asha Balakrishnan, Becaja M Caldwell, Reina S Buenconsejo, and Sara A Carioscia. Global trends in space situational awareness (ssa) and space traffic management (stm). *IDA Science & Technology Policy Institute*, 2018.

[49] Chien-Hua Lee. Matrix bounds of the solutions of the continuous and discrete riccati equations–a unified approach. *International Journal of Control*, 76(6):635–642, 2003.

[50] Faming Li, Maurcio C. De Oliveira, and Robert E. Skelton. Integrating information architecture and control or estimation design. *SICE Journal of Control*,

Measurement, and System Integration, 1(2):120–128, 2008.

[51] Fu Lin, Makan Fardad, and Mihailo R. Jovanovic. Design of optimal sparse feedback gains via the alternating direction method of multipliers. *IEEE Transactions on Automatic Control*, 58(9):2426–2431, sep 2013.

[52] Johan Löfberg. Yalmip: A toolbox for modeling and optimization in matlab. In *Proceedings of the CACSD Conference*, volume 3. Taipei, Taiwan, 2004.

[53] Nick Lord and Ali R. Amir-Moez. Extreme properties of linear transformations. *The Mathematical Gazette*, 76(477):436, nov 1992.

[54] Edward N Lorenz. Predictability: A problem partly solved. In *Proc. Seminar on predictability*, volume 1, 1996.

[55] Edward N Lorenz and Kerry A Emanuel. Optimal sites for supplementary weather observations: Simulation with a small model. *Journal of the Atmospheric Sciences*, 55(3):399–414, 1998.

[56] Reza Madankan, Puneet Singla, and Tarunraj Singh. Optimal information collection for source parameter estimation of atmospheric release phenomenon. In *2014 American Control Conference*. IEEE, jun 2014.

[57] Veronica Magan. GOES 13 Weather Satellite Returns to Service After Hit from Micrometeoroid. `https://tinyurl.com/y5su7zs5`.

[58] James S. McCabe and Kyle J. DeMars. Particle filter methods for space object tracking. In *AIAA/AAS Astrodynamics Specialist Conference*. American Institute of Aeronautics and Astronautics, aug 2014.

[59] Darren McKnight, Mark Matney, Kris Walbert, Sophie Behrend, Patrick Casey, and Seth Speaks. Preliminary analysis of two years of the massive collision monitoring activity. *NASA Technical Report*, 2017.

[60] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *FOCS*, volume 7, pages 94–103, 2007.

[61] Richard H Middleton and Graham C Goodwin. *Digital Control and Estimation: A Unified Approach (Prentice Hall Information and System Sciences Series)*. Prentice Hall Englewood Cliffs, NJ, 1990.

[62] Alessandro Nordio, Alberto Tarable, Fabrizio Dabbene, and Roberto Tempo. Sensor selection and precoding strategies for wireless sensor networks. *IEEE Transactions on Signal Processing*, 63(16):4411–4421, aug 2015.

[63] Daniel L Oltrogge and Salvatore Alfano. The technical challenges of better space situational awareness and space traffic management. *Journal of Space Safety Engineering*, 2019.

[64] Joseph N Pelton. Space weapons, the threat of war in space and planetary defense. In *Space 2.0*, pages 115–128. Springer, 2019.

[65] Sergio Pequito, Soummya Kar, and A. Pedro Aguiar. A structured systems approach for optimal actuator-sensor placement in linear time-invariant systems. In *2013 American Control Conference*. IEEE, jun 2013.

[66] Jonas Radtke, Christopher Kebschull, and Enrico Stoll. Interactions of the space debris environment with mega constellations?using the example of the oneweb constellation. *Acta Astronautica*, 131:55–68, 2017.

[67] Bhaskar D Rao and Kenneth Kreutz-Delgado. An affine scaling methodology for best basis selection. *IEEE Transactions on signal processing*, 47(1):187–200, 1999.

[68] Elizabeth G. Ryan, Christopher C. Drovandi, James M. McGree, and Anthony N. Pettitt. A review of modern computational algorithms for bayesian optimal design. *International Statistical Review*, 84(1):128–154, jun 2015.

[69] Radhika Saraf, Raktim Bhattacharya, and Robert Skelton. $\mathcal{H}_2$ optimal sensing architecture with model uncertainty. In *2017 American Control Conference (ACC)*, pages 2429–2434. IEEE, 2017.

[70] Dawei Shi and Tongwen Chen. Optimal periodic scheduling of sensor networks: A branch and bound approach. *Systems & Control Letters*, 62(9):732–738, sep 2013.

[71] Vasile Sima and Peter Benner. Solving linear matrix equations with slicot. *European Control Conference, ECC 2003*, 04 2015.

[72] Shuang Song and Kamalika Chaudhuri. Composition properties of inferential privacy for time-series data. In *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 814–821. IEEE, 2017.

[73] Yang Song, Chong Xiao Wang, and Wee Peng Tay. Privacy-aware kalman filtering. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, apr 2018.

[74] Jos F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999.

[75] Tyler H. Summers, Fabrizio L. Cortesi, and John Lygeros. On submodularity and controllability in complex dynamical networks. *IEEE Transactions on Control of Network Systems*, 3(1):91–101, mar 2016.

[76] Meng Sun and Wee Peng Tay. Inference and data privacy in iot networks. In *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5. IEEE, 2017.

[77] Youmin Tang, Jaison Ambandan, and Dake Chen. Nonlinear measurement function in the ensemble kalman filter. *Advances in Atmospheric Sciences*, 31(3):551–558, apr 2014.

[78] V. Tzoumas, A. Jadbabaie, and G. J. Pappas. Sensor placement for optimal kalman filtering: Fundamental limits, submodularity, and algorithms. In *2016 American Control Conference (ACC)*. IEEE, jul 2016.

[79] David A Vallado. *Fundamentals of astrodynamics and applications*, volume 12.

Springer Science & Business Media, 2001.

[80] Eric A Wan and Rudolph Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*, pages 153–158. Ieee, 2000.

[81] Jun Wang, Rongbo Zhu, and Shubo Liu. A differentially private unscented kalman filter for streaming data in IoT. *IEEE Access*, 6:6487–6495, 2018.

[82] EC Warner and JT Scruggs. Iterative convex overbounding algorithms for bmi optimization problems. *IFAC-PapersOnLine*, 50(1):10449–10455, 2017.

[83] Brian Weeden. Time for the U.S. military to let go of the civil space situational awareness mission. `https://tinyurl.com/y4886mqf`.

[84] Joel Williamsen. Satellite Vulnerability to Direct Ascent KE ASAT. *SurvivabilityTime to Get Serious 10*, page 23, 2008.

[85] Guocan Wu and Xiaogu Zheng. The error covariance matrix inflation in ensemble kalman filter. In *Kalman Filters - Theory for Advanced Applications*. InTech, feb 2018.

[86] Armin Zare and Mihailo R. Jovanovic. Optimal sensor selection via proximal optimization algorithms. In *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, dec 2018.

[87] Armin Zare, Hesameddin Mohammadi, Neil K. Dhingra, Mihailo R. Jovanovi, and Tryphon T. Georgiou. Proximal algorithms for large-scale statistical modeling and optimal sensor/actuator selection, 2018.

[88] Haotian Zhang, Raid Ayoub, and Shreyas Sundaram. Sensor selection for kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms. *Automatica*, 78:202–210, apr 2017.

[89] Haotian Zhang, Raid Ayoub, and Shreyas Sundaram. Sensor selection for

kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms. *Automatica*, 78:202–210, 2017.

[90] Lei Zhang and Dimitrios Hristu-Varsakelis. Communication and control co-design for networked control systems. *Automatica*, 42(6):953–958, jun 2006.