Lappeenranta University of Technology

School of Engineering Science

Computational Engineering and Technical Physics

Technomathematics

Charles Ocran

DIMENSION REDUCTION IN KALMAN FILTER

Master's Thesis

Examiners: Professor Heikki Haario

                     Professor Lasse Lensu

Supervisors: Professor Heikki Haario

                     Professor Tuomo Kauranne

ABSTRACT

Lappeenranta University of Technology
School of Engineering Science
Computational Engineering and Technical Physics
Technomathematics

Charles Ocran

Dimension reduction in Kalman filter

Master's Thesis

2017

39 pages, 15 figures, and 2 appendix.

Examiners: Professor Heikki Haario
               Professor Lasse Lensu

Keywords: Kalman filter, state, a priori, dimension-reduction.

In this paper, we studied the well known Kalman filter. We derived the equations of the standard KF and the smoothing versions of it. We demonstrated how the KF algorithms work in estimating the state of a system. The sine curve was engaged as the synthetic data in our experiment for the purpose of illustration. Furthermore, we discussed dimension reduction in KF which forms the focus of this paper. The concept of a *priori* dimension reduction in KF to nonlinear inverse problems was reviewed; in which the goal is to sequentially infer the state of a dynamical system. The Lorenz model II in [5] was employed to obtain smoothed solutions of the unknown. Solutions of distinct cases which could be used to describe the manner in which DR works were obtained using Lorenz model II. We illustrated how dimension reduction is useful in KF based on ideas of a *priori* dimension reduction for a given process estimation.

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto
School of Engineering Science
Laskennallinen tekniikka ja teknillinen fysiikka
Techno-matematiikka

Charles Ocran

Dimension vähennys Kalman-suodatuksella

Diplomityö

2017

39 sivua, 15 kuvaa, ja kaksi liitettä.

Tarkastajat: Professor Heikki Haario
                Professori Lasse Lensu

Hakusanat: Kalman-suodatin, a priori, ulottuvuus vähentäminen.
Keywords: Kalman filter, state, a priori, dimension-reduction.

Tässä tutkimuksessa tutkittiin tunnettua Kalman-suodatinta. Johdimme standardin Kalman-suodattimen ja tasausversion yhtälöt. Osoitimme kuinka KF-algoritmit toimivat järjestelmän tilan arvioinnissa. Sini-käyrä käytettiin menetelmän havainnollistamiseen. Lisäksi keskustelimme KF: n dimension vähennyusestä, joka on tämän työn painopiste. Lisäksi tutkimme KF: ja a *priori* dimensio-alennuksen käsitteitä käsitteitä epälineaarisiin käänteisiin ongelmiin; jossa tavoitteena on päätellä peräkkäin dynaamisen järjestelmän tila. Lorenzin II mallia [5] käytettiin tuntemattoman sileiden ratkaisujen saamiseksi. Esitämme tapausten ratkaisut, joita voitaisiin käyttää kuvaamaan tapaa, jolla DR-teokset saatiin käytäen Lorenzin mallia II. Havainnollistimme, kuinka dimension pienentäminen on hyödyllistä KF: ssä, joka perustuu tietyn prosessin estimointiin *priori* ulottuvuuden pienentämiseen.

PREFACE

First of all, I would like to acknowledge God for His bountiful mercies throughout my stay in Finland. Just like men build houses, God builds all things (Hebrews 3:4) and this He has made evident throughout my studies. I thank Him for making my academic pursuit at Lappeenranta University of Technology a success.

Next, I wish to express my earnest gratitude to my supervisor Professor Heikki Haario without whose immense support and counsel this workpiece would not have been completely successful. I remember whenever I got stuck in any part of the thesis especially, the coding aspects of this work, you were always there to assist me overcome those challenges. I must confess that I am one of the fortunate students to work under your supervision. You have always been approachable for discussions all of which proved fruitful throughout this work. I must say that I have really learnt a lot under your mentorship. I wish someday to come I shall get the opportunity again to work under your supervision. Professor Heikki, I deeply appreciate your encouragement and undeniable contribution to the success of this work.

Not to forget, I also want to acknowledge the role played by Doctor Janne Hakkarainen and Professor Marko Laine from the Finnish Meteorological Institute in making this piece of work a reality. I strongly believe that without your codes for the application part of this thesis, the report might have fallen short in content. To both of you, I say a very big thank you.

Finally, to my lovely family, I say thank you for your support and prayers. The love you have shown me throughout my studies cannot be undermined. I am blessed to have a wonderful and supportive family like you by my side in the pursuit of my academic career. My special thanks go to Mr. Isaac Ocran and Patrick Ocran for your admonishment and assistance throughout my studies in Finland.

Lappeenranta, May 25, 2018

Charles Ocran

CONTENTS

ABBREVIATIONS AND SYMBOLS

DR   Dimension reduction

KF   Kalman filter

GPS   Geographical positioning system

SVD   Singular value decomposition

RTSS   Rauch-Tung-Striebel smoother

Ks   Kalman smoother

rv   random variable

$\Pi(b)$ posterior distribution of b (parameter) given y (data).

PCA   Principal component analysis

LSQ   Least Squares

μ Mean value

std Standard deviation

1 INTRODUCTION

In this chapter, we seek to project the background of this work and further discuss the objectives, motivation and structure for the thesis.

1.1 Background

Empirical data is often characterized by noise and randomness in nature. Statistical estimation algorithms play significant roles in various fields such as target tracking. The Kalman filter (KF) remains one of the most well-known mathematical tools that is used for stochastic estimation [1]. It is used to estimate the state of a system from indirect and uncertain (noisy) measurements. The subject "Dimension reduction (DR)" has in recent times been found relevant in the successful implementation of filtering tools which include KF. "DR" aims to represent higher dimensional data in a lower dimension manifold (subspace) such that the new representation possesses the structure of the original data. When this is achieved, it is much easier to carry out our analysis. Preferably, DR leaves us with the fact that the dimension of the new reduced data should correspond to what is called intrinsic dimensionality of the data [2]. The intrinsic dimension of a data refers to the minimum number of parameters required to account for the observed properties of that data. DR in KF implementation is relevant for the following reasons: *Firstly*, DR reduces computer memory requirements for high dimensional problems that involves KF application. The development of various inexact filtering approaches see [3] and [4] have been motivated by memory restrictions. *Secondly*, DR offers computational speedups and this may be beneficial for real time estimation and control problems on a smaller scale as discussed in [5].

1.2 Motivation

Kalman filter is a unique tool with many possible applications, for instance in weather forecasting, vehicle tracking, GPS and navigation, etc. In the pursuit of knowledge and skills, we seek to fully gain insight on the concepts of Kalman filter and how dimension reduction works with this tool. The motivation for this work is to understand DR in the filtering algorithm Kalman filter with experimentation and some numerical applications.

1.3 Objectives and Restrictions

The goal for this paper is to explore the concepts, acquire insight, experiment and discuss on DR-Kalman filter. The focus is to apply the method in future related works using real world data. We seek to implement the concept which we unravel using MatLab.

1.4 Structure of the Thesis

This thesis is structured as follows. In Chapter 2 of this paper, we discuss DR-Kalman filter in some selected literatures. Next, in Chapter 3, we discuss the derivation of standard KF and the smoothed version of it. In Chapter 4, we present a discussion on DR-Kalman filter. The Chapter 5 consists the experimentation of the unraveled algorithms on KF in MatLab. Chapter 6 illustrates the numerical application on DR-Kalman filter. The Chapter 7 presents the discussion and conclusion of this paper.

2   PREVIOUS WORK

In this chapter, we present a brief review on some existing works in relation to DR in filtering methods. Several approaches of DR have been used in various studies.

The idea of DR has been carried out recently in [5], which we consider as the main focus of discussion in this paper. The concept of a priori dimension reduction to non-stationary inverse problems is discussed and implemented in [5]. It must be noted in this case that the aim is to sequentially deduce the state of a given dynamical system. We are able to decipher efficiently problems of this nature by the use of filtering algorithms. However, inverse problems in high-dimensional structures pose computational difficulties due to numerical discretization of the unknown functions that can lead to high-dimensional problems. This makes it hard to carry out the analysis. To alleviate this challenge, DR methods are employed to achieve a low-dimensional intrinsic structure of the inverse problem. At this stage, we can handle the task efficiently and make our analysis on it. In addition, the DR approach in [5] which we discuss in this paper, other existing methods are discussed as follows.

In [6], DR is achieved by the decomposition of the covariance structure into one large- and one small- scale component with the application of empirical orthogonal functions. The *reduced order KF* method is employed in [7] and [8] to attain dimension reduction by projecting the model dynamics onto a fixed low dimension space. This method and the one we focus on share some similarities however, basic distinctions remain, see [5].

The method of *reduced rank KF* is treated in [9]. Here, the prediction uncertainties are propagated in the state space only along directions where the variance of the states grows most rapidly (the referred Hessian singular vectors). The difference between this approach and the one we review in this paper is discussed in [5].

The idea for a DR in *space-time Kalman filter* by Mardia, in which the state process is written in terms of some class of basis functions, is stated in [10]. The reduced dimension is critical for the successful implementation of the KF on higher dimensional space-time datasets, see [10].

In some instances, DR is sought for particle filtering or a sequential Monte Carlo method. See details in [11] and some more emphasis also in [5].

3 KALMAN FILTER

In this chapter, we discuss the basic KF derivation from the Bayesian estimation point of view using the least squares as key focus. We further move on to derive the smoother version of the standard KF.

3.1 The Least Squares (LSQ) Picture to Kalman filter

According to [12, 13], the KF is the Bayesian solution to the problem of sequentially estimating the states of a dynamical system in which the state evolution and measurement processes are both linear and Gaussian [14]. In other words, KF is the closed form solution to the Bayesian filtering equations for the filtering model, in which the dynamic and observation models are linear Gaussian [12, 15] such that

$$x_t = M_t x_{t-1} + E_t \tag{1}$$

$$y_t = K_t x_t + \epsilon_t \tag{2}$$

where, $x_t \in \mathcal{R}^n$ is the state, $y_t \in \mathcal{R}^m$ is the observation, $E_t \sim N(0, S_{E_t})$ is the evolution noise, $\epsilon_t \sim N(0, S_{\epsilon_t})$ is the observation noise and the prior distribution is Gaussian, $x_0 \sim N(m_0, C_0)$. The term M_t is the transition matrix of the evolution model and K_t is the matrix of observation model.

The LSQ can be very useful in the derivation of the standard KF algorithm. The linear least square problem with general covariance and prior information provide a good ground to derive what is called the Kalman gain which in turn becomes a key term in the known KF-algorithm. We go through some mathematical formulations to discuss on Kalman filter. Let us consider the Bayes formula:

$$\Pi(b) \simeq p(y \mid b)p(b) \tag{3}$$

where,
- $p(y \mid b)$ is the likelihood function,
- $p(b)$ is the prior distribution, and
- $\Pi(b)$ is the posterior distribution of b.

Consider first the standard linear model given by:

$$y = Xb + \varepsilon, \ \ \varepsilon \sim N(0, \sigma^2 I) \tag{4}$$

By going through some manipulations (this is provided in the Appendix 2), the LSQ solution can be written as:

$$\hat{b} = (X'X)^{-1}X'y \tag{5}$$

Such that,

$$cov(\hat{b}) = \sigma^2(X'X)^{-1} \tag{6}$$

Next, we solve a general LSQ problem, with error covariance given by $cov(\varepsilon) = S_\varepsilon$, and a prior constraint represented by the Gaussian distribution

$$p(y|b) \simeq e^{-\frac{1}{2}(y-Xb)'S_\varepsilon^{-1}(y-Xb)} \tag{7}$$

$$p(b) \simeq e^{-\frac{1}{2}(b-b_a)'S_a^{-1}(b-b_a)} \tag{8}$$

Suppose that both measurement error and the prior distribution are Gaussian with covariance matrices S_ε and S_a respectively. If the center point of the prior is denoted by b_a, then from the normal distribution function (in the multi-sense), the expressions represented in (7) and (8) rightly holds. Note that that we have ignored the constant term in the normal distribution function. This is for simplicity in computation. Now, we can rewrite (3) as

$$\Pi(b) = e^{-\frac{1}{2}(y-Xb)'S_\varepsilon^{-1}(y-Xb)} \times e^{-\frac{1}{2}(b-b_a)'S_a^{-1}(b-b_a)} \tag{9}$$

By taking the logarithm on both sides of (9), the equation simplifies to

$$-2log\,\Pi(b) = (y-Xb)'S_\varepsilon^{-1}(y-Xb) + (b-b_a)'S_a^{-1}(b-b_a) \tag{10}$$

We must suppose that the covariance matrices S_ε and S_a are positive definite (so the above expressions are positive for all b, otherwise, the minimization problem is meaningless). In that sense, we can 'take square roots' of the inverses of the covariance matrices. These may be decomposed by (for example) the Cholesky decomposition: Thus,

$$S_\varepsilon^{-1} = K_\varepsilon'K_\varepsilon\,; \quad S_a^{-1} = K_a'K_a \tag{11}$$

Re-substitute (11) into (10) to generate:

$$-2log\ \Pi(b) = (y - Xb)'K'_\varepsilon K_\varepsilon(y - Xb) + (b - b_a)'K'_a K_a(b - b_a)$$
$$= (K_\varepsilon y - K_\varepsilon Xb)'(K_\varepsilon y - K_\varepsilon Xb) + (K_a b - K_a b_a)'(K_a b - K_a b_a)$$
$$= \|K_\varepsilon Xb - K_\varepsilon y\|^2 + \|K_b b - K_a b_a\|^2 \tag{12}$$

This derived expression (12) is the norm for the least square (LSQ) problem $\tilde{y} = \tilde{X}b$ where,

$$\tilde{X} = \begin{pmatrix} K_\varepsilon X \\ K_a \end{pmatrix} ; \quad \tilde{y} = \begin{pmatrix} K_\varepsilon y \\ K_a b_a \end{pmatrix},$$

with $N(0, I)$ as the covariance. Utilizing these new expressions, we can write down the known formulas:

$$\hat{b} = (\tilde{X}'\tilde{X})^{-1}\tilde{X}'\tilde{y}$$

$$S = cov(\hat{b}) = (\tilde{X}'\tilde{X})^{-1},$$

for the solution and the covariance. It then remains to calculate the expressions:

$$(\tilde{X}'\tilde{X}) = \begin{pmatrix} K_\varepsilon X \\ K_a \end{pmatrix}' \cdot \begin{pmatrix} K_\varepsilon X \\ K_a \end{pmatrix} = X'K'_\varepsilon K_\varepsilon X + K'_a K_a ;$$

$$(\tilde{X}'\tilde{X}) = X'S_\varepsilon^{-1}X + S_a^{-1}.$$

Similarly,

$$(\tilde{X}'\tilde{y}) = \begin{pmatrix} K_\varepsilon X \\ K_a \end{pmatrix}' \cdot \begin{pmatrix} K_\varepsilon y \\ K_a b_a \end{pmatrix} = X'K'_\varepsilon K_\varepsilon y + K'_a K_a b_a ;$$

$$(\tilde{X}'\tilde{y}) = X'S_\varepsilon^{-1}y + S_a^{-1}b_a.$$

We can therefore write from the least square solution the following expressions:

$$\hat{b} = (X'S_\varepsilon^{-1}X + S_a^{-1})^{-1}(X'S_\varepsilon^{-1}y + S_a^{-1}b_a) \tag{13}$$

$$cov(\hat{b}) = S = (X'S_\varepsilon^{-1}X + S_a^{-1})^{-1} \tag{14}$$

In other words, the posterior distribution is Gaussian with center point $\hat{b}$ and covariance S such that:

$$-2log\pi(b) = (b - b_a)'S^{-1}(b - b_a) + c ; \tag{15}$$

where c is a constant.

The new form of the LSQ solution in (13) and (14) reduces back to the familiar form

when the following terms are set to the conditions given by:

$$S_a = 0 , \quad \text{and} \quad S_\varepsilon = \sigma^2 I.$$

The mathematical statement (15) serve to solve completely the least square solution problem with general Gaussian noise and Gaussian prior. However, the solution can be written in another manner that further clarifies the roles of the prior and measurement. Moreover, that version is what will yield the KF formula. We formulate the steps as follows:
Let

$$I = Sa^{-1}S_a,$$

and substitute this into (14). This implies that

$$
\begin{aligned}
S &= (X'S_\varepsilon^{-1}X + S_a^{-1})^{-1}Sa^{-1}S_a \\
&= (S_aX'S_\varepsilon^{-1}X + I)^{-1}[(S_aX'S_\varepsilon^{-1}X + I)S_a - S_aX'S_\varepsilon^{-1}XS_a] \\
&= S_a - (S_aX'S_\varepsilon^{-1}X + I)^{-1}S_aX'S_\varepsilon^{-1}XS_a \\
&= S_a - (X'S_\varepsilon^{-1}X + S_a^{-1})^{-1}X'S_\varepsilon^{-1}XS_a
\end{aligned}
$$
(16)

Next, the following equation is needed to proceed:

$$(X'S_\varepsilon^{-1}X + S_a^{-1})^{-1}X'S_\varepsilon^{-1} = S_aX'(XS_aX' + S_\varepsilon)^{-1}$$
(17)

The equation (17) can be rewritten in the form

$$X'S_\varepsilon^{-1}(XS_aX' + S_\varepsilon) = S_aX'(X'S_\varepsilon^{-1}X + S_a^{-1})$$

This new form of the equation can be seen to be an identity expression since the simplification of the expressions on both sides produces the same result. Next, we introduce the notation

$$G = S_aX'(XS_aX' + S_\varepsilon)^{-1}.$$
(18)

We bear in mind this notation defined in (18) and then substitute (17) into (16) to obtain the results that follows:

$$
\begin{aligned}
S &= S_a - (X'S_\varepsilon^{-1}X + S_a^{-1})X'S_\varepsilon^{-1}XS_a \\
&= S_a - GXS_a
\end{aligned}
$$
(19)

This equation (19) represents the solution of the covariance estimate. In a similar way, we

can rewrite the formula for $\hat{b}$ to obtain the new form of the solution by the steps below.

$$
\begin{aligned}
\hat{b} &= (X'S_\varepsilon^{-1}X + S_a^{-1})^{-1}Sa^{-1}S_a(X'S_\varepsilon^{-1}y + S_a^{-1}b_a) \\
&= (S_aX'S_\varepsilon^{-1}X + I)^{-1}(S_aX'S_\varepsilon^{-1}y + b_a) \\
&= (S_aX'S_\varepsilon^{-1}X + I)^{-1}[(S_aX'S_\varepsilon X + I)b_a + S_aX'S_\varepsilon^{-1}(y - Xb_a)] \\
&= b_a + (S_aX'S_\varepsilon^{-1}X + I)^{-1}S_aX'S_\varepsilon^{-1}(y - Xb_a) \\
&= b_a + (X'S_\varepsilon^{-1}X + S_a^{-1})^{-1}X'S_\varepsilon^{-1}(y - Xb_a)
\end{aligned}
\tag{20}
$$

Recall the notation G in (18) which has the corresponding expression in the left hand side (LHS) of (17), the final equation in (20) can be written as:

$$
\hat{b} = b_a + G(y - Xb_a).
\tag{21}
$$

The formulations we have obtained so far would help us to simply derive the basic (linear) Kalman filter. We proceed us follows. Now consider a time-dependent process, the state vector x_t observed at the time point t. Recall our previous notations, where the time evolution and the observation states are given by the expressions:

$$
x_t = M_t x_{t-1} + E_t
\tag{22}
$$

$$
y_t = K_t x_t + \epsilon_t
\tag{23}
$$

In the two equations (22) and (23), M_t denotes the model for the state evolution and and K_t is a matrix which multiply x_t to obtain the observation of the state respectively. The error terms E_t and ϵ_t in both equations respectively give the modelling error and the 'usual' observation error. The model is assumed to contain noise and by the choice of E_t and ϵ_t we can stipulate how much we trust the model or data.

In the basic form of Kalman filter, the interest lies in the prediction of new values of x and then correcting the state values by new measured observations y. This takes place iteratively, as the observations form a time series y_t for $t = 1, 2, 3, ..., N$. The phenomena may be cast in the form of successive applications of the Bayes formula. In this case, the *prior* is represented by the *model prediction* and the *likelihood* by the observation equation in (23). We can detail the case as follows: Suppose the estimate $\hat{x}_{t-1}$ is obtained for the state of the previous time point, and the covariance matrix $\hat{S}_{t-1}$ has been estimated. Suppose further that, the errors are Gaussian, with covariance matrices S_{E_t} and S_{ϵ_t}. The model equation then gives the prediction, that is used as the center point of the *prior*

distribution;

$$x_{at} = M_t \hat{x}_{t-1} \tag{24}$$

The covariance of the prior can be computed (assuming the x_t and E_t are statistically independent):

$$S_{at} = cov(M_t \hat{x}_{t-1} + E_t) = M_t \hat{S}_{t-1} M_t' + S_{E_t} \tag{25}$$

We now have all the required information to employ the formulas of the general Least Squares problem: Thus,

- A given Gaussian prior distribution, and

- A given linear (equation) system for the observations.

We employ the following present notations: K_t in place of X; x_t for b and x_{at} for b_a, and make the right substitutions in (19) and (21) to obtain the new solution. Recall also (22) and (23), these equations serves as the basic Kalman system and the notation G as obtained in (18) is what is referred to as the *Kalman gain*. The basic (linear) KF algorithm can be projected as:

$$x_{at} = M_t \hat{x}_{t-1}, \quad (\textit{State propagation}) \tag{26}$$

$$S_{at} = M_t \hat{S}_{t-1} M_t' + S_{E_t}, \quad (\textit{Error covariance propagation}) \tag{27}$$

$$G_t = S_{at} K_t' (K_t S_{at} K_t' + S_\varepsilon)^{-1}, \quad (\textit{Kalman gain matrix}) \tag{28}$$

$$\hat{x}_t = x_{at} + G_t(y - K_t x_{at}), \quad (\textit{State update estimate}) \tag{29}$$

$$\hat{S}_t = S_a - G_t K_t S_{at}, \quad (\textit{Error covariance update}) \tag{30}$$

We can observe from the algorithm that no estimate of model parameters is involved here but, the state vector itself. The expression x_t simply refers to the minimum set of information (data) sufficient to ubiquitously describe the undisturbed dynamical behaviour of the system [16]. In other words, the state is the least amount of data that is required to estimate the system future behaviour using its information on the past behaviour. Typically, the state is unknown. To estimate it, we use a set of observed data denoted by the vector y_t.

In addition, the uncertainty of the model is taken into account with the modelling error terms. Thus, the *Kalman gain* is a measure of information. Therefore, the bigger the error terms associated in the modelling the lesser information we are able to obtain in the state estimation. For instance, bigger error could mean larger values of S_ε in (28), and we can observe that this case would give a smaller *gain* information.

3.2 Derivation of Kalman smoother

Next, we present the KF smoother version which provides much better and accurate estimate than the basic KF [15]. The version presented is called the *Rauch-Tung-Striebel smoother* (see [15]) or simply the *Kalman smoother*. We derive the smoothing form of KF by adopting the approach in [15, 17]. The preliminaries leading to the derivation of KF is in appendix 1.

By the Gaussian distribution rules, the joint density of x_t and x_{t+1} given $y_{1:t}$ can be denoted by the equation:

$$
\begin{aligned}
p(x_t, x_{t+1}|y_{1:t}) &= p(x_{t+1}|x_t)p(x_t|y_{1:t}) \\
&= N(x_{t+1}|A_t x_t, Q_t)N(x_t|m_t, P_t) \\
&= N\left(\left[\begin{array}{c} x_t \\ x_{t+1} \end{array}\right]\middle|m_1, P_1\right)
\end{aligned}
\tag{31}
$$

Where,

$$
m_1 = \left(\begin{array}{c} m_t \\ A_t m_t \end{array}\right)
$$

$$
P_1 = \left(\begin{array}{cc} P_t & P_t A_t' \\ A_t P_t & A_t P_t A_t' + Q_t \end{array}\right)
$$

By the conditioning rule of Gaussian distribution, we can have:

$$
\begin{aligned}
p(x_t|x_{t+1}, y_{1:T}) &= p(x_t|x_{t+1}, y_{1:t}) \\
&= N(x_t|m_2, P_2)
\end{aligned}
\tag{32}
$$

where,

$$
G_t = P_t A_t'(A_t P_t A_t' + Q_t)^{-1}
$$

$$
m_2 = m_t + G_t(x_{t+1} - A_t m_t)
$$

$$
P_2 = P_t - G_t(A_t P_t A_t' + Q_t)G_t'
$$

The joint probability distribution of x_t and x_{t+1} given all the data $y_{1:T}$ is

$$
\begin{aligned}
p(x_t, x_{t+1}|y_{1:T}) &= p(x_t|y_{1:T})p(x_{t+1}|y_{1:T}) \\
&= N(x_t|m_2, P_2)N(x_{t+1}|m_{t+1}^s, P_{t+1}^s) \\
&= N\left(\left[\begin{array}{c} x_{t+1} \\ x_t \end{array}\right]\Big|m_3, P_3\right)
\end{aligned}
\tag{33}
$$

where,

$$
m_3 = \left(\begin{array}{c} m_{t+1}^s \\ m_k + G_t(m_{t+1}^s - A_t m_t) \end{array}\right)
$$

$$
P_3 = \left(\begin{array}{cc} P_{t+1}^s & P_{t+1}^s G_t' \\ P_{t+1}^s G_t & G_t P_{t+1}^s G_t' + P_2 \end{array}\right)
$$

The marginal mean and covariance are thus given as:

$$
\begin{aligned}
m_t^s &= m_t + G_t(m_{t+1}^s - A_t m_t) \\
P_t^s &= P_t + G_t(P_{t+1}^s - A_t P_t A_t' - Q_t)G_t'
\end{aligned}
\tag{34}
$$

The smoothing distribution is then Gaussian with the above mean and covariance:

$$
p(x_t|y_{1:T}) = N(x_t|m_t^s, P_t^s)
$$

The backward recursion equations for KF smoothing with means m_t^s and covariances P_t^s is given by

$$
m_{t+1}^- = A_t m_t \tag{35}
$$

$$
P_{t+1}^- = A_t P_t A_t' + Q_t \tag{36}
$$

$$
G_t = P_t A_t'[P_{t+1}^-]^{-1} \tag{37}
$$

$$
m_t^s = m_t + G_t(m_{t+1}^s - m_{t+1}^-) \tag{38}
$$

$$
P_t^s = P_t + G_t(P_{t+1}^s - P_{t+1}^-)G_t' \tag{39}
$$

Notice: The terms m_t and P_t are the mean and covariance computed via the KF stage. The recursion is initiated from the last time step T, such that $m_T^s = m_T$ and $P_T^s = P_T$. The distinction between the smoothing version of KF and the basic KF is that the solution computed by the former is conditional on the whole measurement data ($y_{1:T}$), while the solution of the latter is conditional only on the measurements obtained before and at the time step t ($y_{1:t}$).

4   DIMENSION REDUCTION-KALMAN FILTER

The goal here is to discuss dimension reduction in the filtering algorithm Kalman filter. We consider the *prior-based dimension reduction* as in [5].

4.1   Linear dynamical problems and DR

Recall the equations in (22) and (23). These serve as the Gaussian state space model (system). In this case, $M_t \in \mathcal{R}^{d \times d}$ gives the forward model that evolves the state over time and $K_t \in \mathcal{R}^{n \times d}$ is the observation model that connects the state to the observations. Suppose $E_t \sim N(0, Q_t)$ and $\epsilon_t \sim N(0, R_t)$ with known covariance matrices $Q_t \in \mathcal{R}^{d \times d}$ and $R_t \in \mathcal{R}^{n \times n}$. The system as linear Gaussian can be solved using the Kalman filter as follows. Consider that at time interval t, the marginal posterior density is given by

$$x_{t-1}|y_{1:t-1} \sim N(x_{t-1}^a, C_{t-1}^a) \tag{40}$$

The *prediction stage* involves propagating this Gaussian ahead (forward) with the model M_t that produces

$$x_t|y_{1:t-1} \sim N(x_t^f, C_t^f) \tag{41}$$

Here, the superscripts *a* and *f* are defined in [5] as "analysis" (posterior) estimate updated with the most current observations and "forecast" (prior) estimate respectively.

The *update stage* gives the resulting posterior (target) density represented as

$$p(x_t|y_{1:t}) \propto exp\left(-\frac{1}{2}\left(\|x_t^f - x_t\|_{C_t^f}^2 + \|y_t - K_t x_t\|_{R_t}^2\right)\right) \tag{42}$$

Equation (42) is Gaussian with known mean and covariance matrix given by the standard Kalman filter formulas derived above and also seen in [15].

Suppose the unknown state x_t is parametrized such that

$$x_t = x_t^f + P_r \alpha_t \tag{43}$$

where, $P_r \in \mathcal{R}^{d \times r}$ is a fixed reduction operator that is time invariant and x_t^f is the predicted (prior) mean. The matrix P_r serves as the reduced subspace basis of the unknown state x_t and can be constructed by computing the SVD of the prior covariance matrix $\sum = W \Lambda W'$. The basis P_r from the leading r eigenvectors $W_r = (w_1, w_2, ..., w_r)$ is com-

puted as well as the square roots of the correspondent eigenvalues $\Lambda_r = (\lambda_1, \lambda_2, ..., \lambda_r)$ such that $P_r = W_r \Lambda_r^{\frac{1}{2}}$. In this case, the dynamics of states can be encapsulated in a lower dimensional manifold spanned by the leading eigenvectors supposing the Λ values of $\sum$ can be decomposed quickly. The filtering equations for the subspace coordinates α_t can now be obtained assuming that the marginal posterior (target) distribution for α_{t-1} at time $t-1$ is

$$\alpha_{t-1}|y_{1:t-1} \sim N(\alpha_{t-1}^a, \Phi_{t-1}^a) \tag{44}$$

From the parameterized state in (43), by putting $t = t - 1$, we obtain the transform given by

$$x_{t-1} = x_{t-1}^f + P_r \alpha_{t-1} \tag{45}$$

The equation (45) helps to produce the Gaussian distribution in the full state space model such that we have

$$x_{t-1}|y_{1:t-1} \sim N(x_{t-1}^f + P_r \alpha_{t-1}^a, P_r \Phi_{t-1}^a P_r'). \tag{46}$$

By the forward propagation of (46) with M_t, the mean and covariance matrix can be obtained for the predictive distribution $x_t|y_{1:t-1} \sim N(x_t^f, C_t^f)$ in the original coordinates given by

$$x_t^f = M_t(x_{t-1}^f + P_r \alpha_{t-1}^a) \tag{47}$$

$$C_t^f = M_t P_r \Phi_{t-1}^a (M_t P_r)' + Q_t \tag{48}$$

Using this as the prior, and substituting (43) into (42), we obtain the following marginal posterior density of α_t as

$$p(\alpha_t|y_{1:t}) \propto exp\left(-\frac{1}{2}\left(\|P_r \alpha_t\|_{C_t^f}^2 + \|y_t - K_t x_t^f - K_t P_r \alpha_t\|_{R_t}^2\right)\right) \tag{49}$$

The mathematical expression (49) is equivalent to the linear problem with Gaussian likelihood $y_t - K_t x_t^f \sim N(K_t P_r \alpha_t, R_t)$ with zero mean Gaussian prior given by $\alpha_t \sim N(0, (P_r(C_t^f)^{-1}P_r')^{-1})$. The posterior for this case becomes $\alpha_t|y_{1:t} \sim N(\alpha_t^a, \Phi_t^a)$, where

$$\Phi_t^a = \left((K_t P_r)R_t^{-1}(K_t P_r)' + P_r(C_t^f)^{-1}P_r'\right)^{-1} \tag{50}$$

$$\alpha_t^a = \Phi_t^a(K_t P_r)' R_t^{-1} r_t \tag{51}$$

with $r_t = y_t - K_t x_t^f$ given by the prediction residual. The expression $P_r (C_t^f)^{-1} P_r'$ does not equate to identity (I) as in the case of the static problems discussed in section 2.1 of [5]. Therefore, P_r does not whiten the prior. The term $(C_t^f)^{-1} P_r'$ can be conveniently evaluated by substituting $\Phi_{t-1}^a = A_t A_t'$ into (48) to obtain

$$C_t^f = M_t P_r A_t (M_t P_r A_t)' + Q_t \tag{52}$$

where, $A_t \in \mathcal{R}^{r \times r}$ is the component of $\Phi_{t-1}^a = A_t A_t'$. For simplicity, we represent $M_t P_r A_t$ as B_t and by using the Woodbury matrix identity given by $(G + UHV)^{-1} = G^{-1} - G^{-1} H (H^{-1} + V G^{-1} U)^{-1} V G^{-1}$, we obtain the expression of

$$
\begin{aligned}
(C_t^f)^{-1} &= (Q_t + B_t B_t')^{-1} \\
&= Q_t^{-1} - Q_t^{-1} B_t (B_t' Q_t^{-1} B_t + I_r)^{-1} B_t' Q_t^{-1}
\end{aligned} \tag{53}
$$

where, I_r is the $r \times r$ identity matrix. Note that he term $(B_t' Q_t^{-1} B_t + I_r)^{-1}$ also has dimension $r \times r$. As such, a given product $(C_t^f)^{-1} b$ can be conveniently computed provided $Q_t^{-1} b$ remains easily solved. This condition must be met for this technique to work. In practice, Q_t is stated by the user usually as a simple (for instance) diagonal matrix.

In summary, the reduced Kalman filter with a fixed basis in a single stage as in [5] can be computed by the following algorithms. The reduced Kalman filter in one stage with *input* (α_{t-1}^a and Φ_{t-1}^a) and *output* (α_t^a and Φ_t^a) is given as:

   i. The prior mean $x_t^f = M_t(x_{t-1}^f + P_r \alpha_{t-1}^a)$ is computed.

  ii. The decomposition $\Phi_{t-1}^a = A_t A_t'$ is performed.

 iii. The matrix $B_t = M_t P_r A_t$ is also computed.

  iv. The term $(C_t^f)^{-1} P_r'$ is computed with the help of (53).

   v. The expressions α_t^a and Φ_t^a are computed using (51) and (50), respectively.

4.2 Nonlinear dynamical problems

In nonlinear dynamical problems, the state space system can be represented as

$$x_t = \mathbf{M}(x_{t-1}) + E_t \tag{54}$$

$$y_t = \mathcal{K}(x_t) + \epsilon_t \tag{55}$$

where M is the nonlinear forward model and $\mathcal{K}$ is the measurement model. The terms E_t, ϵ_t are the model and observation errors respectively. With nonlinear application problems, the standard KF is unsuitable to resolve the . This introduces us to the extended KF and ensemble KF which are useful in solving and discussing nonlinear problems.

4.2.1 The Extended KF and DR

The model and observation matrices in standard KF system are replaced with their linearized forms. The process and observation noises are assumed to be additive. Therefore, the mean is evolved (propagated) with the nonlinear forward model and the prediction is calculated with the nonlinear measurement model. In some instances, M_t and $\mathcal{K}$ are substituted with $\frac{\partial \mathbf{M}(x_{t-1})}{\partial x_{t-1}}$ and $\frac{\partial \mathcal{K}(x_t)}{\partial x_t}$, respectively. In large scale problems, the explicit computation of these matrices is infeasible. As such, often, we derive analytically the linearized model as an operator and this makes it viable to evolve the state forward with the linear model. These so-called tangent linear codes (gradients) are applied in stages (iii) and (v), see page 20. These codes are used in the weather forecasting setting. This is identical to calculating products $M_t b$ where $b \in \mathcal{R}^{d \times 1}$. The term $B_t = M_t P_r A_t$ in stage (iii) can be computed by applying the linearized forward model to r columns of $P_r A_t$. In stage v, $K_t P_r$ is required and this is computed by applying the linearized observation model to the r columns of P_r. In that sense, we only propagate r state vectors via the linearized models in both scenarios instead of d state vectors in the standard KF.

4.2.2 The Ensemble KF and DR

The concept of Ensemble KF is to represent the state and its uncertainty with samples and then replace the covariances in the filtering equations with their empirical estimates calculated from the samples [5].

Suppose we consider the state space model in which the forward model is nonlinear while the measurement model is linear such that

$$x_t = \mathbf{M}(x_{t-1}) + E_t \tag{56}$$

$$y_t = \mathcal{K}x_t + \epsilon_t \tag{57}$$

DR in Ensemble KF requires the representation of the distribution of α_t with samples or an ensemble of states. Suppose we consider that at time interval $t - 1$, we have N_{ens} posterior samples $\{\alpha_{t-1}^{a,1}, \alpha_{t-1}^{a,2}, ..., \alpha_{t-1}^{a,N_{ens}}\}$ present, sampled from the Gaussian posterior

$N(\alpha_{t-1}^a, \Phi_{t-1}^a)$. In the full state space model, the corresponding samples becomes $x_{t-1}^{a,i} = x_{t-1}^f + P_r \alpha_{t-1}^{a,i}$. The samples from the posterior are moved forward with dynamical model such that $x_t^{f,i} = M(x_{t-1}^{a,i})$ in the prediction stage. The empirical covariance matrix of the predicted samples is computed and this is added to the model error covariance to get the prediction error covariance such that

$$
\begin{aligned}
C_t^f &= Cov(\mathbf{M}(x_{t-1}) + E_t) \\
&\simeq X_t X_t' + Q_t,
\end{aligned}
\tag{58}
$$

where

$$
X_t = \frac{\left[(x_t^{a,1} - x_t^f), (x_t^{a,2} - x_t^f), \dots, (x_t^{a,N_{ens}} - x_t^f) \right]}{\sqrt{N_{ens}}} \in \mathcal{R}^{d \times N_e ns}.
$$

Therefore, $X_t X_t'$ serves as the empirical covariance estimate calculated from the prediction ensemble. The mean x_t^f is considered as the posterior mean from the initial stage propagated through the process (evolution) model, $x_t^f = M(x_{t-1}^f + P_r \alpha_{t-1}^{a,i})$, instead of the empirical mean calculated from the prediction ensemble, which is divided by $\sqrt{N_{ens}}$. See [5] for details.

The reduced dimension ensemble filter algorithm remains the same as that of the reduced KF formulas if the measurement model is linear. The only distinction is that the prior covariance matrix C_t^f is defined via the prediction ensemble. In the case where $N_{ens} \ll d$, handling of $(C_t^f)^{-1}$ as required in step *five* of 4.1, can yet be performed efficiently by using the Woodbury matrix identity such that

$$
\begin{aligned}
(C_t^f)^{-1} &= (X_t X_t' + Q_t)^{-1} \\
&= Q_t^{-1} - Q_t^{-1} X_t (X_t' Q_t^{-1} X_t + I_r)^{-1} X_t' Q_t^{-1}
\end{aligned}
\tag{59}
$$

Now, when applying $(C_t^f)^{-1}$ to a vector, we are to note that the inversion of an $N_{ens} \times N_{ens}$ is employed instead of an $r \times r$ matrix. In summary, one step of the reduced dimension ensemble KF with *input* (α_{t-1}^a and Φ_{t-1}^a) and *output* (α_t^a and Φ_t^a) is as follows:

i. Obtain N_{ens} samples from $N(\alpha_{t-1}^a, \Phi_{t-1}^a)$ and transform samples into the full state space such that
$$
x_t^f = x_{t-1}^f + P_r \alpha_{t-1}^{a,i}.
$$

ii. Estimate the prior mean $x_t^f = \mathbf{M}(x_{t-1}^f + P_r \alpha_{t-1}^{a,i})$ and propagate the samples such that
$$
x_t^{f,i} = M(x_{t-1}^{a,i}).
$$

iii. Generate $X_t = \dfrac{\left[(x_t^{a,1} - x_t^f), (x_t^{a,2} - x_t^f), \ldots, (x_t^{a,N_{ens}} - x_t^f) \right]}{\sqrt{N_{ens}}}$.

iv. Compute $(C_t^f)^{-1} P_r$ using the matrix inversion expression (59).

v. Determine α_t^a and Φ_t^a via equations (51) and (50) respectively.

Note: The computational cost associated with the Ensemble algorithms is determined (influenced) by the number of basis vectors r and the number of ensemble members N_{ens}, see remarks 3 to 7 in [5] for more discussions.

4.2.3   Construction of the Reduced Subspace

In this section, we discuss on the possible approaches how to estimate the covariance matrix $\sum$ and construct the projection matrix P_r.

Principal Component Analysis (PCA): The idea of PCA can be employed on "snapshots" for the formation of low dimension subspaces in high dimensional dynamical systems. The "snapshots" are possible model states derived from existing model simulation or data. See [5] for more discussions on the approach in some fields of application.

The main attraction is dynamical systems without steady states, for instance, chaotic models when non-stationary inverse problems are considered. In this setting, trajectories obtained from either adequately long free model simulation or multiple model simulations with randomized initial constraints are used. With a given sufficient number of snapshots $\{\mathsf{x}^{(i)}\}_{i=1}^N$, the subspace basis can be computed by $P_r = U_r \Lambda_r^{\frac{1}{2}}$ through the eigen decomposition of the empirical state covariance given by

$$\Sigma = \frac{1}{N-1} \sum_{i=1}^{N} (\mathsf{x}^{(i)} - \bar{\mathsf{x}})(\mathsf{x}^{(i)} - \bar{\mathsf{x}})' \tag{60}$$

where, $\bar{\mathsf{x}}$ is the empirical state mean given by $\bar{\mathsf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathsf{x}^{(i)}$.

Considering higher dimensional dynamical problems, it is infeasible to form the empirical covariance directly and employ compact (*dense*) matrix eigen decomposition methods. In such scenario, the Krylov subspace approaches or randomized methods see [5] can be applied in addition with matrix-free operations. The matrix vector product with $\sum$ can be computed by eigen decomposition.

The number of snapshots influences the quality of the basis P_r and a sufficiently large N value should be selected to encapsulate the model essential behaviour. This kind of basis

can be built with un-regularized empirical covariance estimates without the assumption of any specific form for the covariance structure. The regularized covariance estimate method is employed in the case where the number of snapshots available s limited. Here, certain assumptions about, the covariance structure are recommended to regularize the estimation task. See [5] for more discussion on the technique.

Also, the next approach is the **regularized covariance estimation**. Here, inference is made on the state correlation structure from snapshots and a *prior* information such as smoothness assumptions. See [5] for more thorough discussion on this method both for stationary and non-stationary Gaussian Processes.

The main challenge lies in computing the P_r term which can be fixed or time dependent.

5   EXPERIMENTATION

In this chapter, we conduct numerical experiments (simulations) on the concepts discussed earlier in Chapter 3.

The *first* experiment we consider is with the basic Kalman filter. We implement the algorithm for the case where we are interested in tracking the sine curve. We initially create a sine signal as the true measurement and obtain the noisy signal by adding noise to our real measurement. The initial value of noise in the noised data in our experiment is zero-μ Gaussian with std of 0.05. We also consider the spherical covariance matrix of the parameters where signal values do not vary according to the Gaussian random walk structure as in [15]. That is, we allow the error parameters in (27) to have the same magnitude. The result of this experiment is illustrated in Figure 2. We compare the result with the normal signal with addition of noise to the sine data in Figure 1. We can observe that the Kalman filter estimation of the sine signal provides possible options of the state estimate(s) with some amount of better estimation.

In addition, we consider the case where the KF parameters (in Q) are allowed to vary according to the Gaussian random walk structure as in [15]. The output for this case is in Figure 3. We can see that the KF estimate here is sort of comprising random moves unlike the KF estimation in Figure 2. The error structure allowed in (27) has high values to some extent.

Next, we implement the KF smoothing algorithms derived in 3.2. The results can be seen in Figure 4. From this chart, it can be observed that the smoother version of KF gives improved tracking on the sine signal. Recall, this is generated from the same noisy data.

Now, we must note that the KF is an estimation method. For that matter, if the noise involved in the data is somehow allowed to vary, say to 0.15 for the noisy data, we expect some difference in the output of tracking the sine curve. This claim can be illustrated in the Figures 5, 6, 7 and 8. Thus, the choice of noise chosen has a strong influence on the KF estimation of a given phenomena. The smaller the noise involved in the data the better the KF estimation will be and vice versa. This confirms the earlier assertion made on the *Kalman gain* as a measure of information being influenced by the error accompanied in the state estimation modelling.
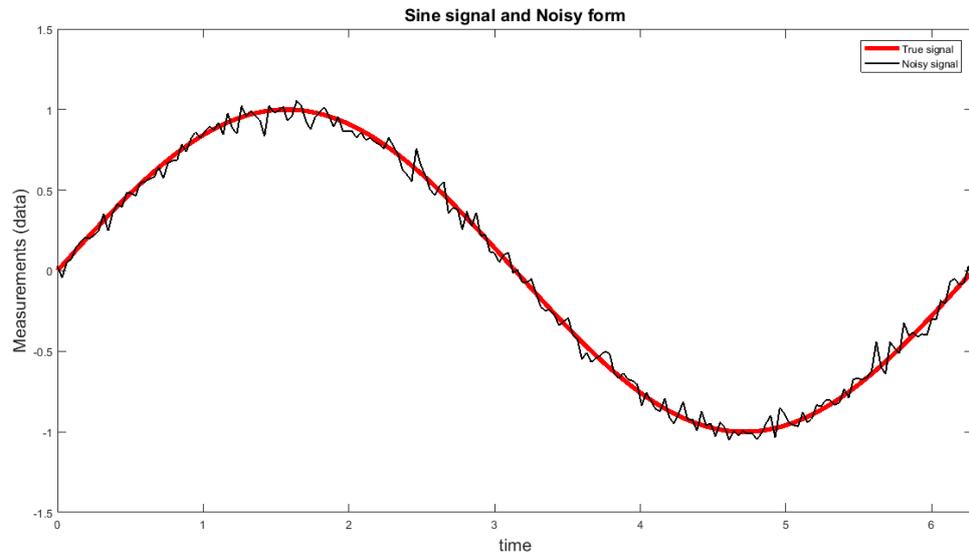
Figure 1. The real sine signal with the noised counterpart created by adding noise to data. Here, the noise is 0.05.
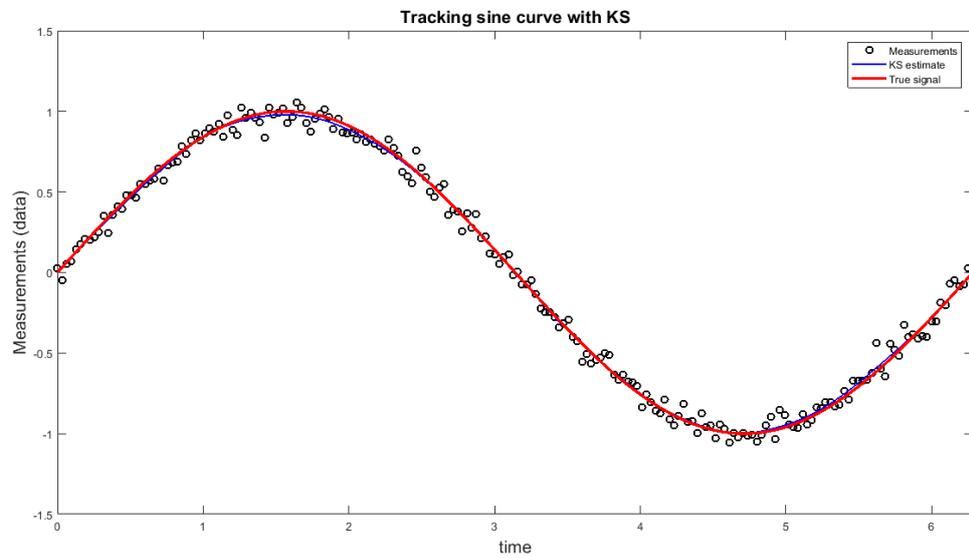

Figure 2. The tracking of sine signal with KF without the involvement of drift. From the data with noise 0.05.
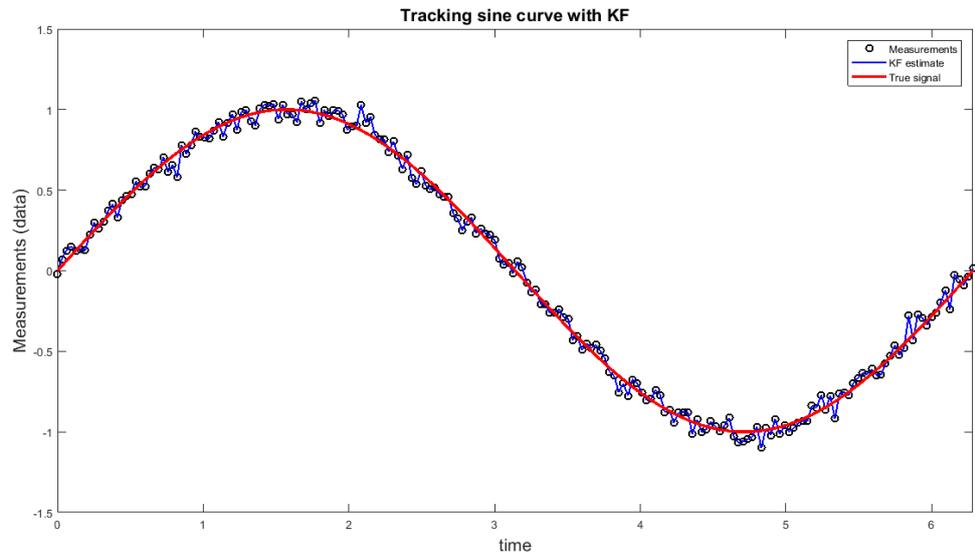
Figure 3. The tracking of sine signal with KF involving drift using data with noise 0.05.
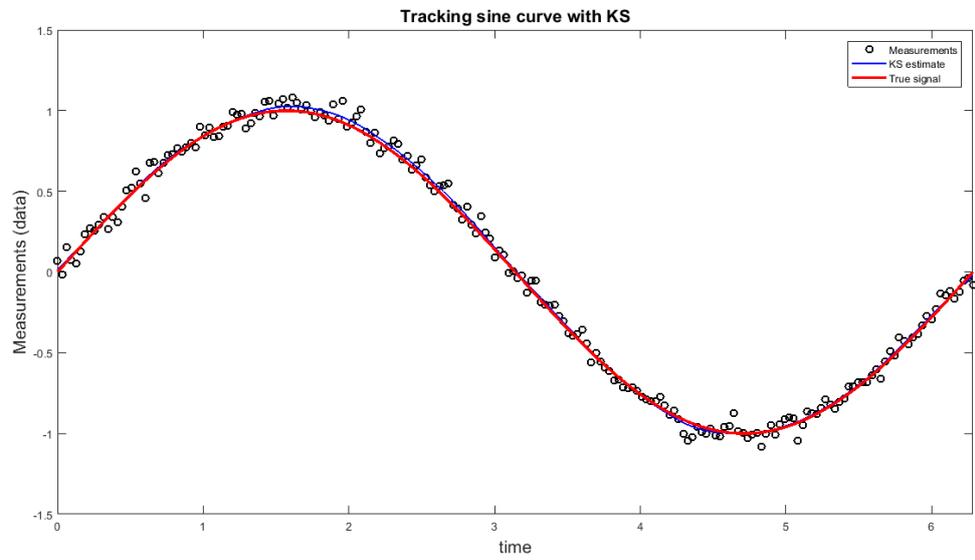

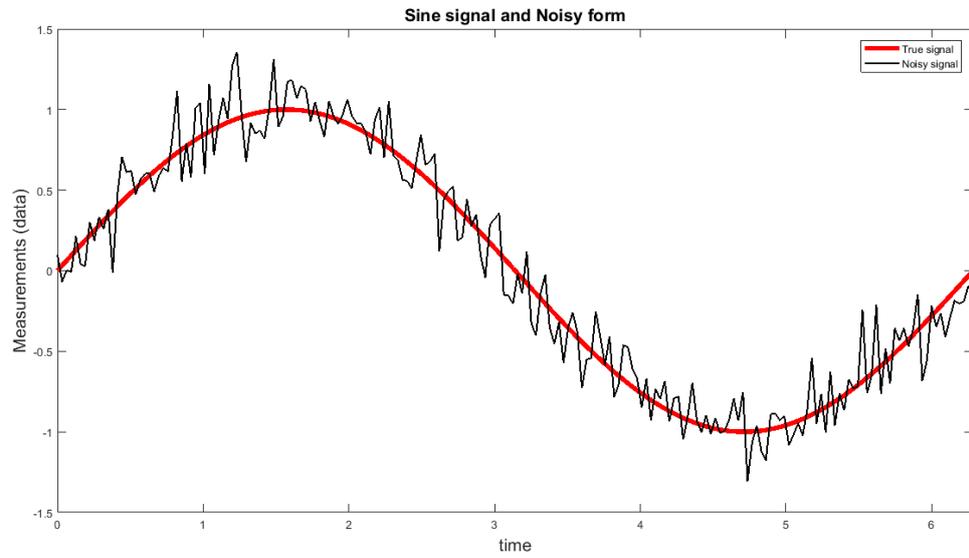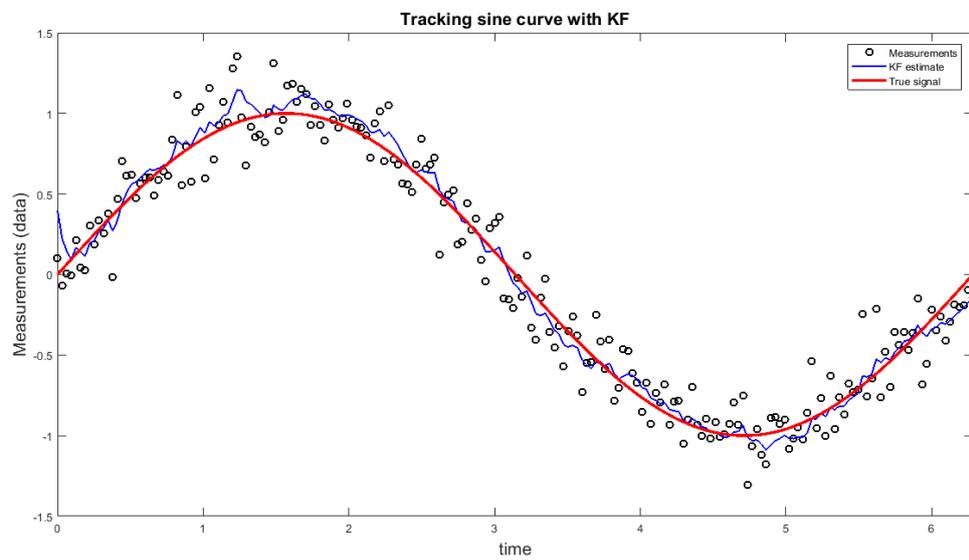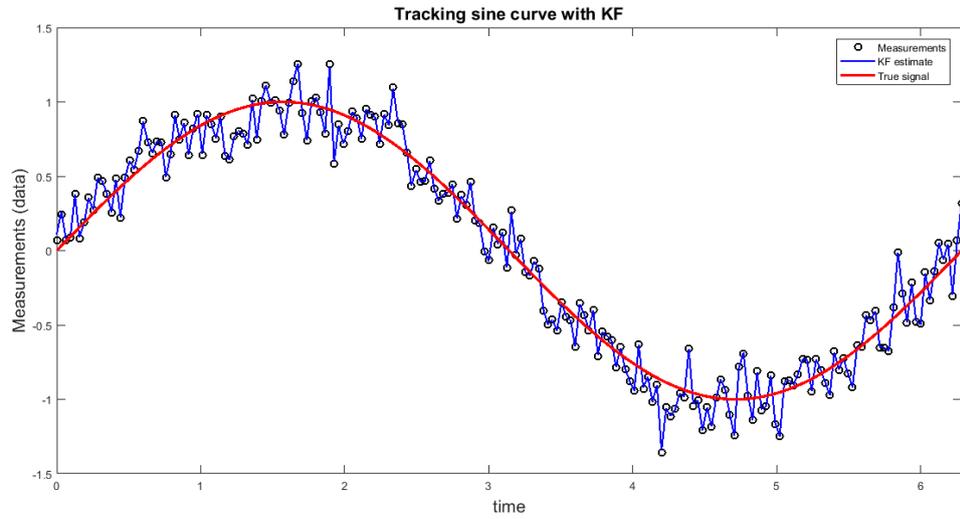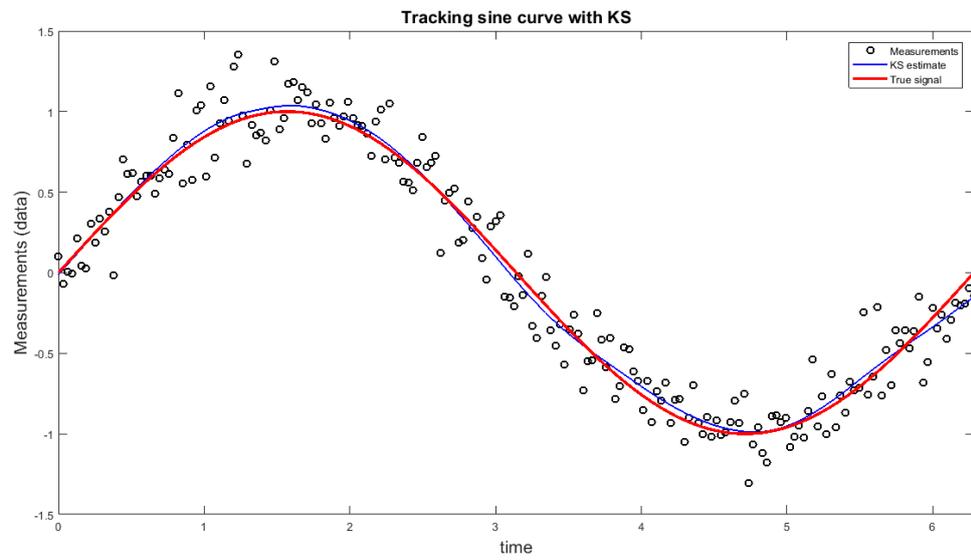**Figure 4.** The tracking of sine signal with Kalman smoother using the data with noise 0.05.

Figure 5. The original sine signal with the noised counterpart created by adding noise to data. Here, the noise is taken to be 0.15.

Figure 6. The tracking of sine signal with KF without the involvement of drift. From the data with noise 0.15.

Figure 7. The tracking of sine signal with KF involving drift using data with noise 0.15.

Figure 8. The tracking of sine signal with Kalman smoother using the data with noise 0.15.

6   APPLICATION

In this chapter, we present some numerical application on dimension-reduction in the filtering algorithm KF. We base our application on the Extended KF by adopting the procedures in [5].

6.1   Lorenz model II

We consider Lorenz 96 model in the general sense for the case of mini scale nonlinear application. The model II explained in [5] is used. The process model can be represented as an ODE system of N formulas:

$$\frac{\mathrm{d}Z_m}{\mathrm{d}t} = \sum_{k=-K}^{K} \sum_{j=-K}^{K} \left(-Z_{m-2J-j}Z_{m-J-i} + Z_{m-J+k-i}Z_{m+J+k} \right) / (J^2 - Z_m) + F_m \quad (61)$$

where $m = 1, ..., N$ is the number of component equations, J is a chosen odd integer and $K = \frac{J-1}{2}$. The variable terms recur at regular intervals, such that $Z_{-i} = Z_{N-i}$ and $Z_{N+i} = Z_i$ for all $i \geq 0$. In the case where $J = 1$, we obtain a reduced version equivalent to the Lorenz 96 model discussed in [18]. The F_n term is called the *forcing constant*. This value is perceived to cause the chaotic behaviour in the model.

We can give an illustration of the Lorenz model II solution by using the following scenarios. The values of J we consider are $7, 11, 19, 35, 67$ with their respective F_m corresponding values as $10, 10, 12, 14, 30$ taken into account for all m values.

From the Figures 9, 10, 11, 12 and 13, we can observe that smoother solutions are obtained as the value of J increases. We can show how dimension reduction performs in various scenarios using these smoother solutions. The fewer the basis vectors required to explain the unknown, the smoother the solution. See [5] for more discussion on this illustrated example.
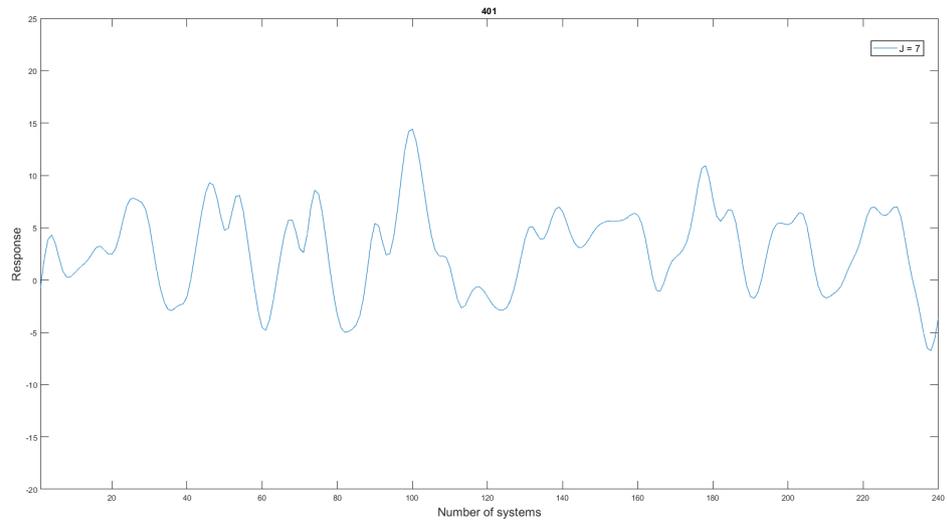
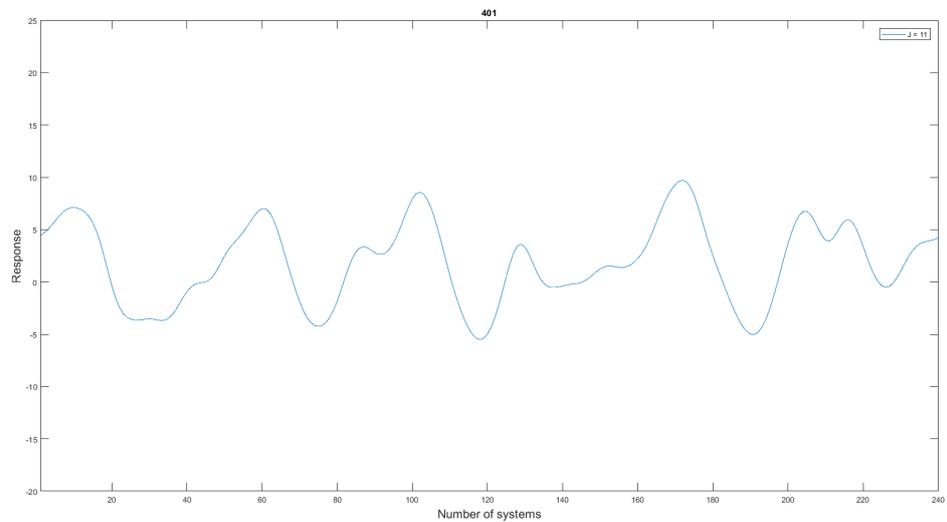**Figure 9.** Solution to Lorenz model II with $J = 7$ and $N = 240$ at one time step.

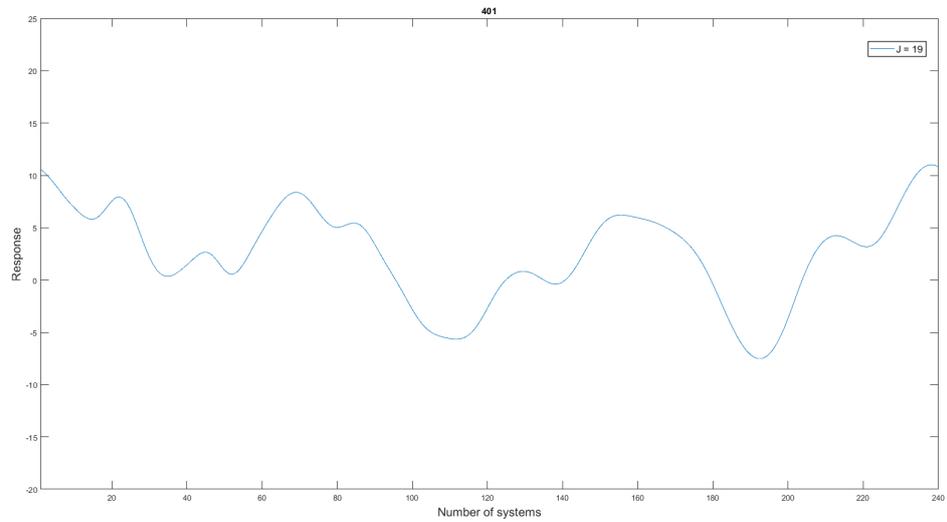**Figure 10.** Solution to Lorenz model II with $J = 11$ and $N = 240$ at one time step.

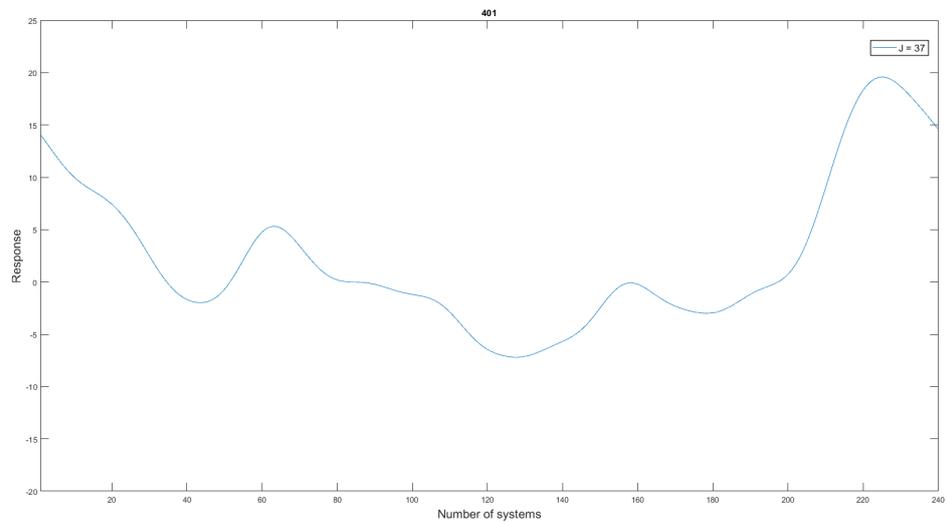**Figure 11.** Solution to Lorenz model II with $J = 19$ and $N = 240$ at one time step.

Figure 12. Lorenz model II solution with $J = 37$ and $N = 240$ at one time step.
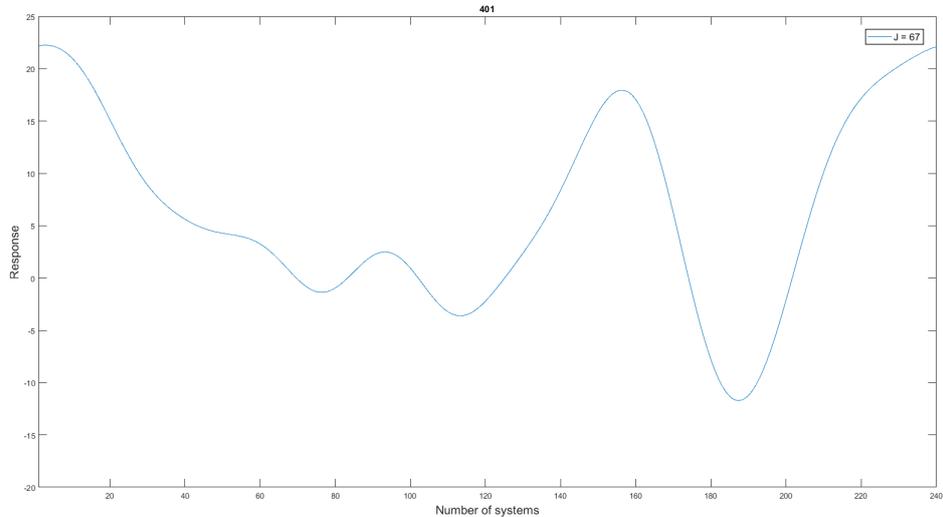
Figure 13. Lorenz model II solution with $J = 67$ and $N = 240$ at one time step.

6.2 Reduced KF for state estimation illustration

The DR in KF here is done on the Extended version. Our earlier discussion on DR KF for nonlinear dynamical problems suggests the construction of reduced subspace. This requires that one computes the covariance and projection matrices ($\sum$ and P_r), respectively.

The Lorenz model II is used to generate new observations assumed to be the true state. This is obtained from using the last column values of initial state observations generated previously by the Lorenz model II. The covariance of the true state is obtained. We apply PCA here to select the principal leading components. We assign the value of 20 principal components here. The Figure 14 backs the choice we make.

The next Figure 15 represents the reduced Extended KF case for state estimation. Here, we consider the Lorenz model II with $J = 37$ and $N = 240$ at single time interval. Data is generated for the prediction by simulating the model. A unit percent of random perturbations from the normal distribution is added to the *forcing constants*. This incurs noise effect into the prediction model. The frequency for the observation is taken to be 2 units and we observe every 10th state vector ($Z_1, Z_{11}, ..., Z_{231}$). This overall produces 21 observations per measurement time interval. The units of time we consider for the data generation here is 20 (in the step of 0.025 with starting point of 1). The 4th order Runge-Kutta method is used to solve the ODE.
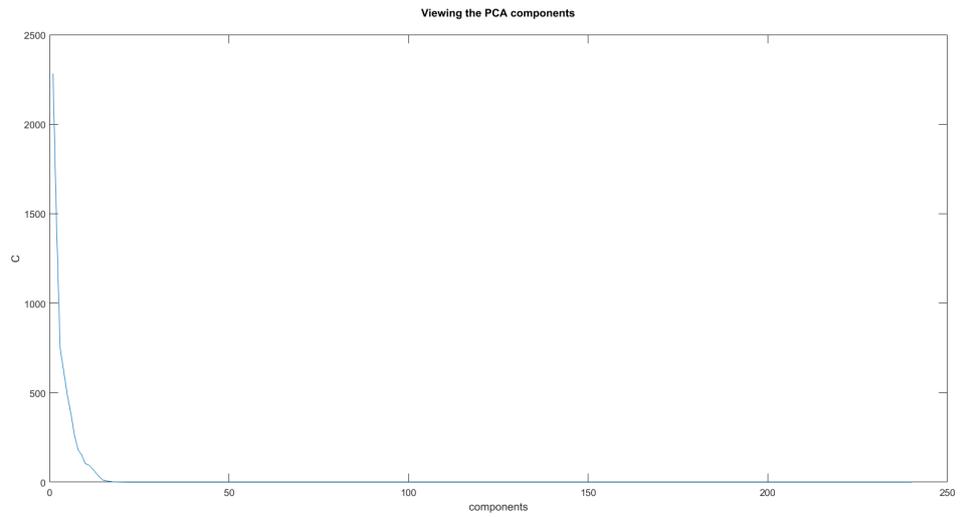
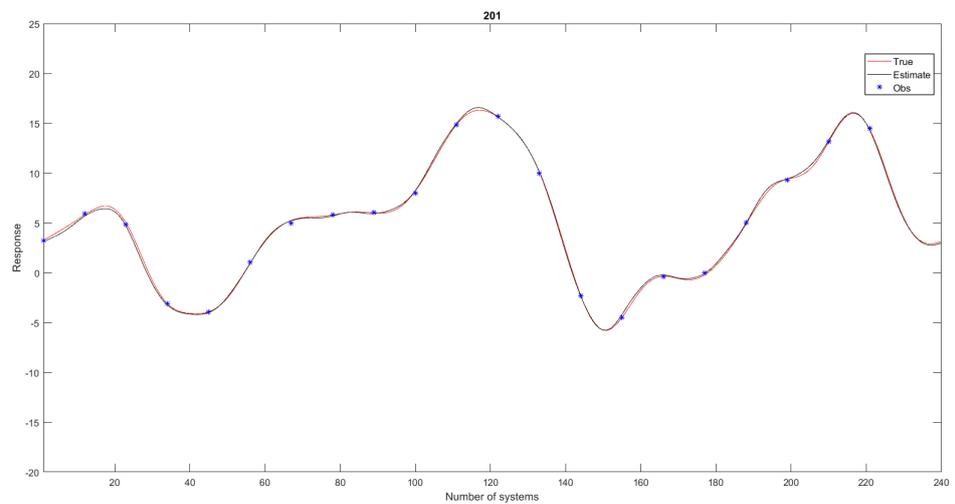**Figure 14.** Visualization of the principal components

Figure 15. Resulting figure from the application of DR with Extended KF

7   DISCUSSION AND CONCLUSION

In this chapter, we present a summary discussion on the derivation of KF, with the experiments and the numerical applications on dimension-reduction KF projected in this paper. Finally, we give a conclusion on the paper.

With the LSQ idea and basics of probability distributions (which include the Bayesian principle), we are able to derive the KF and its smoothing version equations somewhat in a simplified manner. In various other derivations, the approach is not the same as in our case, see [19] for instance.

From the experimentations, we can suggest that KF is a useful tool in the estimation problems of dynamical state systems. We can claim that the KF tools provide a better estimate of states when there is low (minimal) noise structure involved in the estimation problem.

The Lorenz model II solutions demonstrated here provide the grounds for us to show how DR runs in some independent cases. This does not form part of our main thesis discussions but for the purpose of illustration and some emphasis. The model used is assumed to be chaotic and hence, the predictions are obtained from the model's chaotic behaviour. See [5] for more detailed discussion.

We used the Extended KF and implemented the idea of DR to illustrate how the tracking of some phenomena works. We do not seek to make any comparison here between the standard Extended KF and the reduced version of its kind. We leave this for future work due to time constraint.

In conclusion, we are able to present on the basic KF with some experimentation for illustrative purposes. The idea of DR in KF for nonlinear dynamical problems is also discussed in this piece of work. In addition, a numerical application involving DR in KF (specifically, Extended KF) for state estimation purpose is illustrated.

REFERENCES

[1] Greg Welch and Gary Bishop. An introduction to the kalman filter. department of computer science, university of north carolina. *ed: Chapel Hill, NC, unpublished manuscript*, 2006.

[2] Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10:66–71, 2009.

[3] H Auvinen, JM Bardsley, H Haario, and T Kauranne. The variational kalman filter and an efficient implementation using limited memory bfgs. *Int. J. Numer. Meth. Fluids*, 15:1–1, 2008.

[4] Johnathan M Bardsley, Albert Parker, Antti Solonen, and Marylesa Howard. Krylov space approximate kalman filtering. *Numerical Linear Algebra with Applications*, 20(2):171–184, 2013.

[5] Antti Solonen, Tiangang Cui, Janne Hakkarainen, and Youssef Marzouk. On dimension reduction in gaussian filters. *Inverse Problems*, 32(4):045003, 2016.

[6] Anders Malmberg, Ulla Holst, and Jan Holst. Forecasting near-surface ocean winds with kalman filter techniques. *Ocean Engineering*, 32(3-4):273–291, 2005.

[7] Mark A Cane, Alexey Kaplan, Robert N Miller, Benyang Tang, Eric C Hackert, and Anthony J Busalacchi. Mapping tropical pacific sea level: Data assimilation via a reduced state space kalman filter. *Journal of Geophysical Research: Oceans*, 101(C10):22599–22617, 1996.

[8] Ibrahim Hoteit and Dinh-Tuan Pham. An adaptively reduced-order extended kalman filter for data assimilation in the tropical pacific. *Journal of marine systems*, 45(3-4):173–188, 2004.

[9] Michael Fisher. *Development of a simplified Kalman filter*. European Centre for Medium-Range Weather Forecasts, 1998.

[10] Christopher K Wikle and Noel Cressie. A dimension-reduced approach to space-time kalman filtering. *Biometrika*, 86(4):815–829, 1999.

[11] Alexandre J Chorin and Paul Krause. Dimensional reduction for a bayesian filter. *Proceedings of the National Academy of Sciences of the United States of America*, 101(42):15013–15017, 2004.

[12] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[13] Rudolph E Kalman and Richard S Bucy. New results in linear filtering and prediction theory. *Journal of basic engineering*, 83(1):95–108, 1961.

[14] Narayan Kovvali, Mahesh Banavar, and Andreas Spanias. An introduction to kalman filtering with matlab examples. *Synthesis Lectures on Signal Processing*, 6(2):1–81, 2013.

[15] Simo Särkkä. *Bayesian filtering and smoothing*, volume 3. Cambridge University Press, 2013.

[16] Simon S Haykin et al. *Kalman filtering and neural networks*. Wiley Online Library, 2001.

[17] Simo Sarkka. Lecture 7: Optimal Smoothing. `https://users.aalto.fi/~ssarkka/course_k2011/pdf/handout7.pdf`, 2011. Online; accessed 26-March-2018.

[18] Edward N Lorenz. Predictability: A problem partly solved. In *Proc. Seminar on predictability*, volume 1, 1996.

[19] M Yu Byron, Krishna V Shenoy, and Maneesh Sahani. Derivation of kalman filtering and smoothing equations. 2004.

Appendix 1. Preliminaries to Kalman Smoother (Ks) Derivation

- Consider the Gaussian probability distribution of rv x with mean m and given co-variance matrix P: $x \in \mathcal{R}^n$ and $P \in \mathcal{R}^{n \times n}$

$$N(x|m, P) = \frac{1}{(2\pi)^{d/2}|P|^{1/2}} exp\left(-\frac{1}{2}(x-m)'P^{-1}(x-m)\right)$$

Suppose also x and y has the Gaussian densities

$$p(x) = N(x|m, P)$$
$$p(y|x) = N(y|Hx, R)$$

Then, the joint and marginal distributions of x, y can be represented as:

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim N\left(\begin{pmatrix} m \\ Hm \end{pmatrix}, \begin{pmatrix} P & PH' \\ HP & HPH' + R \end{pmatrix}\right)$$
$$y \sim N(Hm, HPH' + R)$$

- Suppose the rv x and y have joint Gaussian probability distribution

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim N\left(\begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} A & C \\ C' & B \end{pmatrix}\right)$$

Then, the marginal and conditional densities of x and y can be given as:

$$x \sim N(a, A)$$
$$y \sim N(b, B)$$
$$x|y \sim N(a + CB^{-1}(y-b), A - CB^{-1}C')$$
$$y|x \sim N(b + C'A^{-1}(x-a), B - C'A^{-1}C)$$

Appendix 2. Deriving the LSQ estimate $\hat{b}$ and its covariance

Recall the equation,

$$y = Xb + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2 I).$$

We can assume that no measurement noise is involved. Such that, $\varepsilon = 0$.
In addition, we suppose X is matrix such that there exist the transpose form. This implies that,

$$X' \cdot y = X' \cdot (Xb + \varepsilon)$$

$$X'y = X'Xb + X' \cdot \varepsilon$$

We can write that,

$$X'y = X'Xb$$

Solving for b estimate implies,

$$X'X\hat{b} = X'y$$

Now, suppose all variables in matrix X are linearly independent, then $(X'X)^{-1}$ exist. So that,

$$(X'X)^{-1}X'X\hat{b} = (X'X)^{-1}X'y$$

And thus,

$$\hat{b} = (X'X)^{-1}X'y.$$

Next, we can obtain the covariance in the following manner:

$$cov(\hat{b}) = cov((X'X)^{-1}X'y)$$

Using the identity that,

$$cov(Ax) = Acov(x)A'$$

And noting that our variable of interest is y, we can write the following.

$$cov(\hat{b}) = [(X'X)^{-1}X'cov(y)X(X'X)^{-1}]$$
$$= cov(y)[(X'X)^{-1}X'X(X'X)^{-1}]$$
$$= \sigma^2(X'X)^{-1}$$

We should note that $cov(y) = \sigma^2 I = \sigma^2$. And thus, we obtain the final result of $cov(\hat{b})$ as above.