HIGHER ORDER KALMAN FILTERING FOR NONLINEAR SYSTEMS

by

Deanna Colonna Easley
A Dissertation
Submitted to the
Graduate Faculty
of
George Mason University
In Partial fulfillment of
The Requirements for the Degree
of
Doctor of Philosophy
Mathematics

Committee:

_____	Dr. Tyrus Berry, Dissertation Director
_____	Dr. Thomas Wanner, Committee Member
_____	Dr. Daniel Anderson, Committee Member
_____	Dr. John Cressman, Committee Member
_____	Dr. Maria Emelianenko, Department Chair
_____	Dr. Donna M. Fox, Associate Dean Office of Student Affairs & Special Programs, College of Science
_____	Dr. Fernando R. Miralles-Wilhelm, Dean, College of Science
Date: _____	Spring Semester 2022 George Mason University Fairfax, VA

Higher Order Kalman Filtering for Nonlinear Systems

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

By

Deanna Colonna Easley
Master of Science
George Mason University, 2017
Bachelor of Science
George Mason University, 2015

Director: Dr. Tyrus Berry, Professor
Department of Mathematical Sciences

Spring Semester 2022
George Mason University
Fairfax, VA

Dedication

I dedicate my dissertation work to my family and loved ones who have meant and continue to mean so much to me. A special feeling of gratitude to my loving parents, Flavia Colonna and Glenn Easley, and my brother, Julian Easley, who have never left my side and have always been supportive. I am also immensely grateful for my loving nonna, Mima Maranò, who made sure to pray for me for every exam I took in my life despite being an ocean away.

I also dedicate this dissertation to my many friends who have supported me throughout the entire doctorate program. I will never forget how my friend Sayomi Kamimoto encouraged me to pursue a PhD at a time in my life when I had my doubts that I could make it. I give special thanks to my friends, Taylor Stevens, Wafa Mahzari, and Cigole Thomas for always being there for me through hard times. Finally, I dedicate this work to Calvin Stanley, whose words of encouragement and confidence in my abilities have stuck with me and continue to push me forward. I am so grateful to all of them for believing in me.

Acknowledgments

I would like to thank the following people who made this possible: Dr. Tyrus Berry, Dr. Thomas Wanner, and Dr. Anderson whose stimulating lectures had sparked my interest in applied mathematics. Special thanks to my advisor, Dr. Tyrus Berry, for all his valuable advice and profound belief in my work.

Table of Contents

List of Figures

Abstract

HIGHER ORDER KALMAN FILTERING FOR NONLINEAR SYSTEMS

Deanna Colonna Easley, PhD

George Mason University, 2022

Dissertation Director: Dr. Tyrus Berry

We seek to improve upon and generalize the Ensemble Kalman Filter (EnKF) by defining a Higher Order Kalman Filter. The Kalman filter consists of two steps: forecast and assimilation. In this thesis we develop the forecast step of our desired Higher Order Kalman Filter with the higher order unscented transform (HOUT). The HOUT is a quadrature rule that estimates the expected value of the first four moments of a distribution, i.e. the mean, covariance, skewness and kurtosis. We then discuss how to generalize the assimilation step. The original Kalman Filter can be derived in three ways: the Bayesian approach, the Minimum Mean-Square Estimate (MMSE) approach and the Closure approach. Each derivation provides a different avenue for us to derive the Higher Order Kalman Filter. In order to generalize the Bayesian approach to the first four moments, instead of using a Gaussian likelihood and prior, we use exponentials with a quartic polynomial as the exponent. In order to generalize the MMSE approach we consider deriving optimal quadratic filters. Finally we may generalize the closure approach by deriving the ordinary differential equations for the skewness and kurtosis and instead of assuming that the skewness is zero, we seek new closures for the first four moments rather than just the first two.

Chapter 1: Introduction

The Kalman Filter is an algorithm that produces estimates of unknown variables using a series of measurements observed over time, which contain statistical noise and other inaccuracies. The Kalman Filter consists of two steps: forecast and assimilation. The forecast step is our prediction step in which the Kalman filter produces estimates of the current state variables, including their uncertainties. After the next measurement is observed, comes the assimilation step or our update step in which the estimates are updated using a weighted average.

Given the linear dynamical system

$$\begin{aligned} x_k &= Fx_{k-1} + \omega_k \\ y_k &= Hx_k + \nu_k \end{aligned}$$

where $\omega_k \sim \mathcal{N}(0, Q)$ and $\nu_k \sim \mathcal{N}(0, R)$, the forecast step of the Kalman Filter consists of the equations

$$\begin{aligned} x_k^- &= Fx_{k-1}^+ \\ P_k^- &= FP_{k-1}^+F^\top + Q \end{aligned}$$

where x_k^- is the mean of the forecast distribution and P_k^- is the covariance of the forecast distribution. The assimilation step of the Kalman Filter consists of the equations

$$\begin{aligned} x_k^+ &= x_k^- + K(y_k - Hx_k^-) \\ P_k^+ &= (I - KH)P_k^- \end{aligned} \tag{1.1}$$

where x_k^+ is the mean of the posterior distribution and P_k^+ is the posterior of the forecast distribution and $K = P^- H^\top (R + HP^- H^\top)^{-1}$ is the Kalman gain. The latter are known as the Kalman Equations.

Once we consider a nonlinear dynamical system, however, a more sophisticated tool is needed to try to forecast the uncertainty. Hence, the Ensemble Kalman Filter (EnKF) was developed to account for nonlinear systems. For the EnKF, in place of the linear forecast step as previously mentioned, the unscented transform is used. Specifically, the EnKF represents the distribution of the system state with an ensemble, a collection of state vectors, and replaces the covariance matrix with the empirical covariance generated from the ensemble.

Both the Kalman Filter and the EnKF have the assumption that all probability distributions involved are Gaussian. However, if the system is nonlinear, the true distributions are no longer Gaussian, so instead for the EnKF we are just making a Gaussian approximation. This means that technically the assimilation step for the Kalman Filter is no longer optimal as it is based on linear-Gaussian assumptions, but the EnKF continues to use the same assimilation step.

This manuscript aims to generalize the EnKF to higher order moments. In other words, rather than assuming a particular distribution, we assume that we are only given the first four moments of the distribution. For the generalized forecast step, we developed a new approach for estimating the expected values of nonlinear functions applied to multivariate random variables with arbitrary distributions. In particular, we efficiently represent the distribution using a small number of quadrature nodes which are called σ-points. The classical scaled unscented transform (SUT) matches the mean and covariance of a distribution. In this manuscript, we introduce the higher order unscented transform (HOUT) which also matches any given skewness and kurtosis tensors. This work has been completed and accepted for publication [1].

Recall that in order to have a complete Kalman Filter, we need the generalization of the assimilation step. The original Kalman Filter can be derived in multiple ways, so several approaches to generalize the Kalman equations are possible. In the remainder of this introduction, we will first overview the motivation and methods of generalizing the unscented transform (forecast step) in Section 1.1 and then we will overview the methods of generalizing the Kalman equations (assimilation/update step) in Section 1.2.

1.1 Generalizing the Unscented Transform

A fundamental problem in uncertainty quantification is to approximate the expectation of a function $f : \mathbb{R}^d \to \mathbb{R}$ applied to a random variable X sampled from a probability measure dp on $\mathbb{R}^d$, namely

$$\mathbb{E}[f(X)] = \int f(x)\, dp. \tag{1.2}$$

Even when the distribution is known, this can be a challenging computation in high dimensions, and the problem is often compounded by uncertain or incomplete knowledge of f and dp. Moreover, in most problems of interest f has an extremely complex form. For example, f may encapsulate the solution of a differential equation and the computation of some feature of interest on the solution. So we may not be able to assume that f is known in an explicit form, but instead that f or an approximation to f is available only as a black-box computational scheme which can take inputs x and produce outputs $f(x)$. Similarly, the type of partial knowledge of the probability measure can vary widely. We may have an explicit expression for a density function $p(x) = dp/dx$ (if it even exists), or we may only have some samples of dp, or estimates of some of the moments.

The method developed in this manuscript will assume that the first four moments of the probability measure, dp, exist and can be accurately estimated. Our method will not use any additional knowledge of the probability beyond these moments. Moreover, we will not require any explicit knowledge of f, so our method is applicable if f is a black-box. In order

to derive error bounds we will require some regularity assumptions on f and additional decay assumptions on the probability measure at infinity. While our error bounds depend on the error of approximating f by a polynomial, our method does not require us to actually find such an approximation, and will only require evaluating f on a small number of test points.

The problem of approximating (1.2) can be approached as a problem of numerical quadrature (also known as cubature when x has dimensionality greater than one – we will use the term quadrature for both). A quadrature is an approximation of the form

$$\mathbb{E}[f(X)] \approx \sum_{i=1}^{N} w_i f(x_i) \tag{1.3}$$

where x_i are called nodes and w_i are called weights. The goal is to find a small number of nodes and weights that accurately represent the probability measure for a large space of functions $f \in \mathcal{C}$ (e.g. Section 4.2, we will consider $\mathcal{C}$ to be the space of n times continuously differentiable functions). A common strategy in quadrature methods is to choose nodes and weights so that the above approximation is actually an equality for all f in some finite dimensional subspace $\tilde{\mathcal{C}} \subset \mathcal{C}$ (such as a space of polynomials up to a fixed degree). For f outside of $\tilde{\mathcal{C}}$ we can then attempt to bound the error in the approximation (1.3) if we can control the error between f and its projection into $\tilde{\mathcal{C}}$. When f is sufficiently smooth and dp is concentrated in a small region, then it is reasonable to approximate f using the space of polynomials up to a fixed degree. Under these assumptions, we can bound the error between f and a low degree polynomial via interpolation error bounds.

Ensuring that (1.3) holds with equality for all polynomials up to degree k is equivalent to satisfying the so-called moment equations,

$$m_{j_1,\ldots,j_n} = \mathbb{E}\left[X_1^{j_1} X_2^{j_2} \cdots X_n^{j_n}\right] = \sum_{i=1}^{N} w_i x_1^{j_1} x_2^{j_2} \cdots x_n^{j_n} \tag{1.4}$$

for all $j_1 + j_2 + \cdots j_n \leq k$, since polynomials of the form $x_1^{j_1}, \ldots x_n^{j_n}$ form a basis of the space of all polynomials of degree less than or equal to k. In other words, we are asking that the empirical moments of the nodes x_i, weighted by discrete probabilities w_i, exactly agree with the true moments $m_{j_1,\ldots,j_n}$ of the distribution. When $k = 2$, the moment equations specify that weighted nodes must match the mean vector and covariance matrix of the true distribution, and this is achieved with the so-called *Scaled Unscented Ensemble* (SUT) [2] (see section 2.1 for an overview).

The quadrature approach is an alternative to stochastic quadrature methods such as Monte-Carlo quadrature, which is commonly used in particle filtering. Stochastic quadratures use random variables X_i to build quadrature rules such that

$$\mathbb{E}[f(X)] \approx \mathbb{E}\left[\sum_{i=1}^{N} w_i f(X_i)\right]. \tag{1.5}$$

However, the computed value $\sum_{i=1}^{N} w_i f(X_i)$ will be stochastic. This means that in addition to a possible approximation error in (1.5), we also have an error due to the variance of the random variable $\sum_{i=1}^{N} w_i f(X_i)$. While it is often easier to design stochastic quadrature methods where the approximation error in (1.5) is small or even zero, for many problems controlling the variance error requires a large number of random variables X_i and hence a large number of function evaluations. When f is very expensive to compute, it may be more efficient to use a small deterministic ensemble and accept the quadrature error in (1.3) in order to avoid the large ensemble size that would be required to control the variance in a stochastic quadrature.

The problem (1.2) is often part of a larger problem such as filtering [3], particle filtering [4], adaptive filtering [5], smoothing [6], parameter estimation [7–9] and even model-free filtering [10]. In all these applications it can be beneficial to have a deterministic approximation of (1.2) to improve the stability of the overall algorithm. For example, filters built on random ensembles can fail catastrophically since they can generate realizations that

would normally have very low probability but lead to perverse behavior [11, 12]. Similarly, gradient based optimization method for parameter estimation will need to carefully account for any stochasticity in the objective function, so replacing a stochastic quadrature with a deterministic quadrature can be desirable in certain applications.

The highly successful Unscented Kalman Filter (UKF) [3] is based on the SUT, as are many of the other methods mentioned above. A closely related technique, called Cubature Kalman Filters (CKF) [13], follows a similar strategy and is typically designed to achieve a high degree of exactness under a Gaussian assumption on the distribution. Another potentially deterministic method would be quadrature based on sparse grids [14, 15]; however designing such a quadrature typically requires detailed knowledge of the probability distribution. Similarly, polynomial chaos expansions [16, 17] require explicit knowledge of the function f and the distribution. Our method is an alternative quadrature that only requires us to know the first four moments of the distribution. Moreover, the nodes of our quadrature will be adapted to the moments of the distribution. A potential future application of the method developed here would be to a higher order UKF which tracks four moments, and this was one of the inspirations behind this work. However, the current work only generalizes the forecast step of the UKF to four moments, and generalizing the assimilation step of the UKF is a significant remaining challenge.

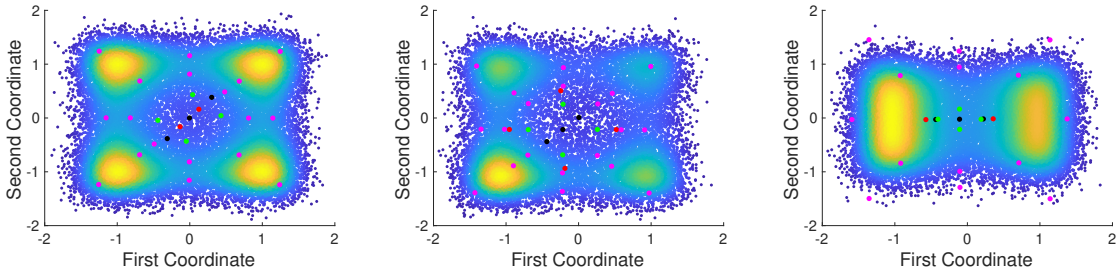
In this manuscript, we develop a Higher Order Unscented Transform (HOUT) based on tensor decomposition of the first four moments of a distribution. Whereas the UKF (and implicitly most CKFs) only require the rank decomposition of the covariance matrix, the HOUT requires the CANDECOMP/PARAFAC (CP) decomposition of higher order tensors such as the skewness and kurtosis. The CP decomposition of a k-tensor is defined by vectors v_i such that,

$$T = \sum_{i=1}^{p} v_i^{\otimes k}, \tag{1.6}$$

i.e. the CP decomposition decomposes a tensor as the summation of rank-1 tensors, $v_i^{\otimes k}$ for

$i = 1, \ldots, p$. The minimal value of p such that the above decomposition exists is called the rank of T. For detailed definitions of tensors, tensor product ($\otimes$) and tensor decomposition, see Section 2.2. For the sake of giving an overview, we assume these definitions for now. For details on tensor product and CP decomposition see Definitions 4 and 6. For a more detailed introduction to tensors we suggest [18, 19].

Ideally, we would like an exact CP decomposition (1.6) with the minimum possible number of vectors; however this turns out to be an NP-complete problem [20, 21]. Instead, we will use an effective algorithm for obtaining an approximate CP decomposition up to an arbitrary tolerance. The algorithm was originally suggested by [22], and it works by repeatedly subtracting the best rank-1 approximation to a tensor until the norm of the residual is less than any desired tolerance. Many methods have been developed based on this idea (see [23] and citations therein) and in [24] it was proven to converge but without any convergence rate. In Chapter 3, we give the first proof that this algorithm converges linearly and we derive an upper bound on the convergence rate. While the approximate decomposition typically requires many more vectors than the minimal CP decomposition, it avoids the NP-completeness of that problem and gives us an effective algorithm.



(a) (b) (c)

Figure 1.1: The 4-moment σ-points of the HOUT we developed of (a) a quadramodal distribution, (b) a skewed quadramodal distribution, and (c) a skewed bimodal distribution. The black points are the σ-points that correspond to the mean, the green points are the σ-points corresponding to the covariance, the points in red are the σ-points corresponding to the skewness, and the magenta points are the σ-points that correspond to the kurtosis.

In Chapter 2 we briefly review the SUT and some tensor facts and notation, including

the Higher Order Power Method (HOPM) [25] that we will use for finding tensor eigenvectors. Based on the HOPM, we prove the convergence of the approximate CP decomposition algorithm in Chapter 3. This proof also requires some new inequalities relating the maximum eigenvalue of a tensor to the entries of the tensor, and these inequalities are likely to be of independent interest. In Chapter 4 we introduce the Higher Order Unscented Transform (HOUT) which generalizes the SUT in order to match arbitrary skewness and kurtosis tensors. The HOUT gives a quadrature rule with degree of exactness four that is applicable to arbitrary distributions. For a preview of the nodes of the HOUT, see Figure 1.1 where we consider data sampled from two dimensional distributions with nontrivial skewness and kurtosis tensors. For each distribution we show the HOUT nodes that are designed so that the first four moments computed with this small number of nodes match the true moments of the distribution up to the specified tolerance. In Section 4.2 we derive error bounds under appropriate regularity assumptions on f and decay assumptions on the probability. In Section 4.3 we demonstrate the HOUT on various non-Gaussian multivariate random variables on complex nonlinear transformations.

1.2 Generalizing the Kalman Equations

Let us focus on the assimilation step of our desired Higher Order Kalman Filter. The original Kalman Equations were derived in three distinct ways: either with the Bayesian approach, or with the Minimum Mean-Square Estimate (MMSE) approach, or with the Closure approach. Each approach provides a different method for us to derive the Higher Order Kalman Filter. Our main contribution in this manuscript is with the generalized Bayesian approach. However, we will also outline how the other approaches can be generalized.

The original Kalman Filter's assimilation step was derived with the Bayesian approach in the following way. Assume we have the prior $p(x)$ (the probability of x) and the likelihood $p(y|x)$ (the probability of y given x). We want to find the posterior $p(x|y)$ (probability of

x given y). By Bayes' law,

$$p(x|y) \propto p(y|x)p(x).$$

Assuming both the likelihood and prior are Gaussian, the posterior is also Gaussian. We then write out the likelihood and the prior as exponential functions where

$$p(x) = e^{-\frac{1}{2}(x-x^-)^\top (C^+)^{-1}(x-x^-)} \quad \text{and} \quad p(y|x) = e^{-\frac{1}{2}(y-Hx)^\top R^{-1}(y-Hx)}$$

to get

$$p(x|y) \propto e^{-\frac{1}{2}(y-Hx)^\top R^{-1}(y-Hx)} e^{-\frac{1}{2}(x-x^-)^\top (C^+)^{-1}(x-x^-)}.$$

After some algebraic calculations and completing the square inside the exponent, we get

$$p(x|y) \propto e^{-\frac{1}{2}\left[x^\top \left(H^\top R^{-1} H + (C^-)^{-1}\right)x - 2\left(y^\top R^{-1} H + (x^-)^\top (C^-)^{-1}\right)x\right]}.$$

Since the posterior is Gaussian, we know it will have the following form

$$p(x|y) \propto e^{-\frac{1}{2}(x-x^+)^\top (C^+)^{-1}(x-x^+)} \propto e^{-\frac{1}{2}(x^\top (C^+)^{-1}x - 2(x^+)^\top (C^+)^{-1}x)}$$

Thus we derive the mean and covariance of the posterior, getting the formulas for x^+ and C^+, where we define

$$x^+ = C^+\left(H^\top R^{-1} y + (C^-)^{-1}x^-\right) \quad \text{and} \quad C^+ = \left(H^\top R^{-1} H + (C^-)\right)^{-1}.$$

Reformulating x^+ and C^+, we get that

$$x^+ = x^- + K(y - Hx^-) \quad \text{and} \quad C^+ = (I - KH)C^-$$

where $K = C^- H^\top (R + HC^- H^\top)^{-1}$ is the Kalman gain. This is the general idea for how the original Kalman equations were derived using this approach.

A way we can see how to generalize the Bayesian approach is: instead of using a Gaussian likelihood and prior, i.e. exponentials with a degree two polynomial as the exponent, we use exponentials with a degree four polynomial as an exponent. This way we have an order 4-moment filter. The Bayesian update will simply be adding the polynomials in the exponent of the prior distribution and likelihood function. However this method does present some challenges. By replacing the quadratic with higher degree polynomial, we lose the nice property that Gaussians possess, which is that the mean and covariance can be easily determined from the polynomial. So there is an algebraic problem of how we can determine the first four moments of a distribution purely from the polynomial. In fact, it is also a challenge when using a cubic. The real challenge here will be figuring out the connection between the moments and the coefficients of the polynomial.

In Chapter 5 we introduce novel methods of extending the Kalman equations (1.1) to higher order moments. We will mainly cover the Bayesian approach in Section 5.1 but we will also discuss the other approaches to derivation in Sections 5.2 and 5.3. Combining the higher order unscented transform developed in Section 4.1 and the higher order Kalman filter proposed in Chapter 5 will lead to a higher order ensemble Kalman filter for nonlinear processses with more general noise statistics.

Chapter 2: Background

We start by reviewing the Scaled Unscented Transform (SUT) in Section 2.1 which has degree of exactness two. We then briefly introduce our tensor notation in Section 2.2 and tensor-vector products and tensor norms in Section 2.3. Finally, in Section 2.4 we review tensor eigenvectors and eigenvalues and the Higher Order Power Method (HOPM) [25] for finding them.

2.1 Scaled Unscented Transform

The Scaled Unscented Transform (SUT) was introduced by Julier and Uhlmann in [2] and further developed in [3, 26–29]. The fundamental goal of this paper is to generalize their method to higher order moments. This work was started in [28] which worked on matching the skewness, and below we show that CP decompositions are the key to generalizing their approach.

The SUT uses the mean and covariance of a distribution to choose quadrature nodes and weights such that the quadrature rule has degree of exactness 2. Degree of exactness k means that a quadrature rule is exact for computing the expectation of polynomials up to degree k. The fundamental insight of Julier and Uhlmann is that achieving degree of exactness 2 is equivalent to matching the first two moments of the distribution. Moreover, they showed that this can be efficiently accomplished using a matrix square root of the covariance matrix.

Definition 1 (ith column of the symmetric matrix square root of A)**.** *Let A be a $d \times d$ matrix. We define the* **ith column of the symmetric matrix square root of $\boldsymbol{A}$***, denoted*

$\sqrt{A}_i$, *by*

$$\sum_{i=1}^{d} \sqrt{A}_i^{\otimes 2} = \sum_{i=1}^{d} \sqrt{A}_i \sqrt{A}_i^{\top} = A.$$

The notation $v^{\otimes k}$ will be defined below. Note that the following definition can use any matrix square root but we have found empirically that the unique symmetric matrix square root has the best performance. The negative of any matrix square root is also a matrix square root. The following definition perturbs the mean μ by both a matrix square root and its negative to create an ensemble of $2d + 1$ points.

Definition 2 (The Scaled Unscented Transform (SUT) [2])**.** *Let dp be a probability measure with mean $\mu \in \mathbb{R}^d$ and the covariance $C \in \mathbb{R}^{d \times d}$. Then for some $\beta \in \mathbb{R}$ the* $\boldsymbol{\sigma}$**–points** *are defined by*

$$\sigma_i = \begin{cases} \mu & \text{if } \ i = 0 \\ \mu + \beta\sqrt{C}_i & \text{if } \ i = 1, \ldots, d \\ \mu - \beta\sqrt{C}_{i-d} & \text{if } \ i = d+1, \ldots, 2d \end{cases}$$

and the corresponding **weights** *are defined by*

$$w_i = \begin{cases} 1 - \frac{d}{\beta^2} & \text{if } \ i = 0 \\ \frac{1}{2\beta^2} & \text{if } \ i = 1, \ldots, 2d \end{cases}$$

We note that the choice of β can have significant impact on the effectiveness of the transform.

Remark 1. *The absolute condition number of the Scaled Unscented Transform is bounded above by*

$$\sum_{i=0}^{2d} |w_i| = \left| 1 - \frac{d}{\beta^2} \right| + \frac{d}{\beta^2}.$$

If $\beta \geq \sqrt{d}$, *then* $\sum_{i=0}^{2d} |w_i| = 1$. *If* $\beta < \sqrt{d}$, *then* $\sum_{i=0}^{2d} |w_i| = \frac{2d}{\beta^2} - 1$.

The following theorem says that the SUT matches the first two moments, μ and C.

Theorem 1 (Empirical mean and Empirical covariance [2])**.** *For an arbitrary* β, *we have*

$$\mu = \mathbb{E}[X] = \sum_{i=0}^{2d} w_i \sigma_i \qquad \text{and} \qquad C = \mathbb{E}[(X-\mu)(X-\mu)^\top] = \sum_{i=0}^{2d} w_i (\sigma_i - \mu)(\sigma_i - \mu)^\top$$

and if $q : \mathbb{R}^d \to \mathbb{R}$ *is a polynomial of degree at most* 2, *we have,* $\mathbb{E}[q(X)] = \sum_{i=0}^{2d} w_i q(\sigma_i)$.

We should note that if the distribution has zero skewness, such as a Gaussian distribution, then the symmetry of the nodes yields degree of exactness 3, and, in the specific case of a Gaussian distribution the choice $\beta = \sqrt{3}$ achieves degree of exactness 4 [2, 3, 29]. The choice $\beta = \sqrt{d}$ is often called the *unscented transform* and sets $w_0 = 0$ so that only $2d$ of the σ-points are required. The ability of the SUT to match the first four moments of the Gaussian distribution has led some to associate the SUT with a Gaussian assumption, however this is not required and degree of exactness 2 is achieved for arbitrary distributions. Our goal is to generalize the unscented transform to higher moments, which are tensors.

2.2 Tensors

Tensors are essentially multidimensional matrices, which will be used to conveniently express the notions of covariance, skewness and kurtosis in a similar fashion.

Definition 3 (k-order tensor)**.** *For positive integers* d *and* k, *a tensor* T *belonging to* $\mathbb{R}^{d^k}$ *is called a* ***k*-order tensor** *or simply a* ***k*-tensor**.

In particular, a vector in $\mathbb{R}^d$ can be viewed as a first order tensor and a $d \times d$ matrix as a second order tensor. Let $x \in \mathbb{R}^d$. We note that the outer product $xx^\top$ yields a $d \times d$

matrix whose ij-entry can be represented as

$$(xx^\top)_{ij} = x_i x_j = (x \otimes x)_{ij} = (x^{\otimes 2})_{ij}.$$

We generalize this process of forming higher order tensors from vectors with following definition.

Definition 4 (kth-order tensor product)**.** *Let $v \in \mathbb{R}^d$ and k be a positive integer then the* ***k*th-order tensor product** *is a k-tensor denoted*

$$v^{\otimes k} = \underbrace{v \otimes v \otimes \cdots \otimes v}_{k \text{ times}},$$

the elements are given by $(v^{\otimes k})_{i_1,\ldots,i_k} = v_{i_1} \ldots v_{i_k}$.

Definition 4 immediately connects tensor products to the moments of a distribution since we can represent the covariance as $C = \mathbb{E}[(X-\mu)^{\otimes 2}] = \int (x-\mu)^{\otimes 2}\, dp(x)$ so that the skewness S and kurtosis K can be defined as

$$S = \int (x-\mu)^{\otimes 3}\, dp(x) \qquad K = \int (x-\mu)^{\otimes 4}\, dp(x),$$

so that, for example,

$$S_{ijk} = \int \left((x-\mu)^{\otimes 3}\right)_{ijk} dp(x) = \int (x-\mu)_i (x-\mu)_j (x-\mu)_k\, dp(x).$$

The following definition generalizes the notion of a rank-1 matrix to tensors.

Definition 5 (Rank-1 Tensor)**.** *Let $T \in \mathbb{R}^{d^k}$ then T is called a* **rank-1 tensor** *if there exists a $v \in \mathbb{R}^d$ such that*

$$v^{\otimes k} = T.$$

For tensors that are not rank-1, one may seek to decompose the tensor as a sum of rank-1 tensors.

Definition 6 (CP decomposition)**.** *The vectors* $v_1, ..., v_p$ *form a* **CP decomposition** *of a tensor* T *if,*

$$T = \sum_{\ell=1}^{p} v_\ell^{\otimes k}$$

and the minimum p *for which such a decomposition exists is called the* rank *of the tensor* T*.*

This notion of rank agrees with the classical notion of matrix rank in the case of second order tensors but many of the properties of matrix rank do not generalize to higher order tensors [19–21, 30, 31].

2.3 Tensor Multiplication and Tensor Norms

In this section we introduce the necessary definitions and notation along with some preliminary results that will be needed below. To discuss how tensor multiplication works, let us first look at the simplest case where we multiply a 2-tensor with a 1-tensor. Recall that for a matrix $A \in \mathbb{R}^{d\times d}$ and $v \in \mathbb{R}^d$, the matrix-vector multiplication Av is given by $(Av)_i = \sum_{j=1}^{d} A_{ij}v_j$ so we define two natural tensor-vector products

$$(A \times_1 v)_i \;=\; \sum_{j=1}^{d} A_{ji}v_j = (A^\top v)_i \qquad \text{and} \qquad (A \times_2 v)_i = \sum_{j=1}^{d} A_{ij}v_j = (Av)_i$$

Analogously, for a 3-tensor $S \in \mathbb{R}^{d\times d\times d}$ and a vector $v \in \mathbb{R}^d$, the tensor-vector multiplication is carried out as follows

$$(S \times_1 v)_{ik} \;=\; \sum_{j=1}^{d} S_{jik}v_j, \qquad (S \times_2 v)_{ik} = \sum_{j=1}^{d} S_{ijk}v_j, \qquad (S \times_3 v)_{ik} = \sum_{j=1}^{d} S_{ikj}v_j,$$

each case resulting in a $d \times d$ matrix. For example, if $S \in \mathbb{R}^{3\times3\times3}$ and $v \in \mathbb{R}^3$ such that

$$S = \left[\begin{array}{ccc} S_{111} & S_{121} & S_{131} \\ S_{211} & S_{221} & S_{231} \\ S_{311} & S_{321} & S_{331} \end{array}\right] \left[\begin{array}{ccc} S_{112} & S_{122} & S_{132} \\ S_{212} & S_{222} & S_{232} \\ S_{312} & S_{322} & S_{332} \end{array}\right] \left[\begin{array}{ccc} S_{113} & S_{123} & S_{133} \\ S_{213} & S_{223} & S_{233} \\ S_{313} & S_{323} & S_{333} \end{array}\right]$$

then

$$S \times_1 v = \begin{bmatrix} S_{111}v_1 + S_{211}v_2 + S_{311}v_3 & S_{112}v_1 + S_{212}v_2 + S_{312}v_3 & S_{113}v_1 + S_{213}v_2 + S_{313}v_3 \\ S_{121}v_1 + S_{221}v_2 + S_{321}v_3 & S_{122}v_1 + S_{222}v_2 + S_{322}v_3 & S_{123}v_1 + S_{223}v_2 + S_{323}v_3 \\ S_{131}v_1 + S_{231}v_2 + S_{331}v_3 & S_{132}v_1 + S_{232}v_2 + S_{332}v_3 & S_{133}v_1 + S_{233}v_2 + S_{333}v_3 \end{bmatrix}.$$

Generalizing this to arbitrary order tensors yields the following definition.

Definition 7 (n-mode product of a tensor)**.** *The* ***n*-mode product** *of a k-order tensor $T \in \mathbb{R}^{d^k}$ with a vector $v \in \mathbb{R}^d$, denoted by $T \times_n v$, is defined elementwise as*

$$(T \times_n v)_{i_1,\ldots,i_{n-1},i_{n+1},\ldots,i_k} = \sum_{j=1}^{d} T_{i_1,\ldots,i_{n-1},j,i_{n+1},\ldots,i_k} v_j.$$

Note that $T \times_n v \in \mathbb{R}^{d^{k-1}}$, so the order of the resulting tensor is decreased by 1.

The above definition can also be generalized for tensor-matrix multiplication [19]. Finally we note that the Frobenius norm for matrices can be generalized to tensors in the following way.

Definition 8 (Tensor Frobenius Norm [19])**.** *The* **Frobenius norm of a tensor** $T \in \mathbb{R}^{d^k}$

is the square root of the sum of the squares of all its elements

$$\|T\|_F = \sqrt{\sum_{i_1=1}^{d} \cdots \sum_{i_k=1}^{d} {T_{i_1,\ldots,i_k}}^2}.$$

Moments of a distribution have the special property in that they are symmetric in the following sense.

Definition 9 (Symmetric Tensor)**.** *A tensor $T \in \mathbb{R}^{d^k}$ is* **symmetric***, if the tensor is invariant to permutations of the indices, i.e.*

$$T_{i_1 \cdots i_k} = T_{p(i_1 \cdots i_k)}$$

for any permutation p.

Notice that if a tensor is symmetric then the n-mode product is independent of the mode, i.e. if $T \in \mathbb{R}^{d^k}$ is symmetric then

$$T \times_n v = T \times_m v$$

for any $1 \leq n, m \leq k$. The next lemma shows that the tensor Frobenius norm has a particularly simple formula for rank-1 tensors.

Lemma 1. *Let $v \in \mathbb{R}^d$ and k be a positive integer then the tensor Frobenius norm of the kth-order tensor product is the same as the Euclidean norm of v raised to the k, i.e.*

$$\|v^{\otimes k}\|_F = \|v\|^k.$$

Proof. By the definition of the tensor Frobenius norm,

$$\|v^{\otimes k}\|_F^2 = \sum_{i_1=1}^{d} \cdots \sum_{i_k=1}^{d} [(v^{\otimes k})_{i_1,\ldots,i_k}]^2$$

and since $(v^{\otimes k})_{i_1 \ldots i_k} = v_{i_1} v_{i_2} \cdots v_{i_k}$, we have $\|v^{\otimes k}\|_F^2 = \sum_{i_1=1}^{d} \cdots \sum_{i_k=1}^{d} v_{i_1}^2 \cdots v_{i_k}^2$, so

$$\|v^{\otimes k}\|_F^2 = \sum_{i_1=1}^{d} v_{i_1}^2 \sum_{i_2=1}^{d} v_{i_2}^2 \cdots \sum_{i_k=1}^{d} v_{i_k}^2 = \underbrace{\|v\|^2 \|v\|^2 \cdots \|v\|^2}_{k \text{ times}}$$

by definition of $\|v\|$ so

$$\|v^{\otimes k}\|_F = \|v\|^k.$$

□

2.4 Tensor Eigenvectors and Normalized Power Iteration

The key to our approximate CP decomposition is rank-1 approximation which is based on tensor eigenvectors which can be found with the Higher Order Power Method (HOPM) which we review in this section.

Definition 10 (Tensor Eigenvectors and Eigenvalues)**.** *Let $T \in \mathbb{R}^{d^k}$ be a symmetric tensor then $v \in \mathbb{R}^d$ is an* **eigenvector** *and $\lambda \in \mathbb{R}$ is the corresponding* **eigenvalue** *of T if*

$$(((T \times_1 v) \times_1 v) \cdots \times_1 v) = \lambda v.$$

Note that since T is symmetric, the choice of n-mode product does not affect the definition of a tensor eigenvector. The next lemma shows that an eigenvalue-eigenvector pair provides a rank-1 approximation of a tensor in the Frobenius norm.

Lemma 2. *Let T be a k-order symmetric tensor with dimension d, i.e. $T \in \mathbb{R}^{d^k}$ and $v \in \mathbb{R}^d$ be a unit length eigenvector of T with eigenvalue $\lambda \neq 0$. Then*

$$\|T - \lambda v^{\otimes k}\|_F^2 = \|T\|_F^2 - \lambda^2$$

and $\|T\|_F \geq \lambda$.

Proof. We first wish to show that $\|T - \lambda v^{\otimes k}\|_F^2 = \|T\|_F^2 - \lambda^2$.

$$\begin{aligned}
\|T - \lambda v^{\otimes k}\|_F^2 &= \sum_{i_1=1}^{d} \cdots \sum_{i_k=1}^{d} [(T - \lambda v^{\otimes k})_{i_1,\ldots,i_k}]^2 = \sum_{i_1=1}^{d} \cdots \sum_{i_k=1}^{d} [T_{i_1,\ldots,i_k} - \lambda (v^{\otimes k})_{i_1,\ldots,i_k}]^2 \\
&= \sum_{i_1=1}^{d} \cdots \sum_{i_k=1}^{d} (T^2_{i_1,\ldots,i_k} - 2\lambda T_{i_1,\ldots,i_k} v_{i_1} v_{i_2} \cdots v_{i_k} + \lambda^2 v_{i_1}^2 v_{i_2}^2 \cdots v_{i_k}^2) \\
&= \|T\|_F^2 - 2\lambda \sum_{i=1}^{d} v_i (T \times_2 v \times_3 v \times_4 \cdots \times_k v)_i + \lambda^2 \|v^{\otimes k}\|_F
\end{aligned}$$

Since $\|v\| = 1$ and by Lemma 1, $\|v^{\otimes k}\|_F = 1$, hence

$$\|T - \lambda v^{\otimes k}\|_F^2 = \|T\|_F^2 - 2\lambda \langle v, \lambda v \rangle + \lambda^2 = \|T\|_F^2 - 2\lambda^2 \|v\|_2^2 + \lambda^2 = \|T\|_F^2 - \lambda^2$$

Since $\|T - \lambda v^{\otimes k}\|_F \geq 0$, $\|T\|_F^2 - \lambda^2 \geq 0$ so $\|T\|_F^2 \geq \lambda^2$ and taking square roots, $\|T\|_F \geq |\lambda|$. □

It immediately follows from Lemma 2 that the eigenvector with the largest eigenvalue will achieve the best rank-1 approximation among the eigenpairs. In fact, it has been shown that the eigenpair with the largest eigenvalue achieves the best possible rank-1 approximation of the tensor [22, 32]. This fact will form the basis for an effective algorithm for finding an approximate CP decomposition in the next section.

Finally, an effective algorithm for finding the eigenvector associated to the largest eigenvalue in absolute value is the Higher Order Power Method (HOPM) originally developed in [25] and further analyzed in [32,33]. In the case of symmetric tensors the Symmetric-HOPM (S-HOPM) has a simpler form that is very similar to Normalized Power Iteration (NPI) but is not guaranteed to converge [22]. The HOPM algorithm for a symmetric order-k tensor $T \in \mathbb{R}^{d^k}$ requires initialization with the left singular vector, u, corresponding to the largest singular value of the unfolding (reshaping) of the tensor into a $d \times d^{k-1}$ matrix. The HOPM then defines k sequences of vectors, $v_0^{(1)}, \ldots, v_0^{(k)}$, by initializing them all to be equal to u, $v_0^{(1)} = \cdots = v_0^{(k)} = u$, and inductively updating

$$w = T \times_1 v_{j+1}^{(1)} \times_1 \cdots \times_1 v_{j+1}^{(i-1)} \times_1 v_j^{(i+1)} \times_1 \cdots \times_1 v_j^{(k)} \tag{2.1}$$

$$v_{j+1}^{(i)} = \frac{w}{||w||}$$

for each $i = 1, ..., k$ and then increments j. Note that in formula (2.1), the subscripts do not represent the indices of the vector, they refer to the iteration whereas in Algorithm 2.1 subscripts indicate vector indices.

Notice that the product that updates $v_{j+1}^{(i)}$ is the tensor T multiplied by the $k-1$ other vectors, leaving out $v_j^{(i)}$. Also note that we use the already updated $(j+1)$-step vectors for the first $i-1$ products and the j-step vectors for the last $k-i$ products. The HOPM is guaranteed to converge to an eigenvector of T [33], and when T is symmetric all $v_j^{(1)}, ..., v_j^{(k)}$ converge to the same eigenvector but may differ in sign for even order tensors. For completeness we summarize the HOPM algorithm of [25] in Algorithm 1.

Unlike the case of matrices, for tensors of order greater than two the basins of attraction for multiple distinct eigenvalues can have non-zero measure. It has been observed [22,32,33] that initialization with the left singular vector, u, of the tensor unfolding typically leads to convergence to the eigenvector with the largest eigenvalue. The next chapter will rely on

Algorithm 1 Higher Order Power Method (HOPM) [25]

Inputs: A k-tensor $T \in \mathbb{R}^{d^k}$
Outputs: Eigenvector $v \in \mathbb{R}^d$ and eigenvalue λ such that $T \times_1 v \times_1 \cdots \times_1 v = \lambda v$

Reshape T into a $d \times d^{k-1}$ matrix and compute the leading left singular vector, v_0
Initialize $v^{(1)} = v^{(2)} = \cdots = v^{(k)} = u$, $\lambda = \text{Inf}$ and $\lambda_{\text{prev}} = 0$
while $|\lambda - \lambda_{\text{prev}}| > \text{tol}$ **do**
  for $\ell = 1, ..., k$ **do**
    Set $v_s^{(\ell)} = \sum_{i_1,...,i_{\ell-1},i_{\ell+1},...,i_k=1}^{d} T_{i_1,...,i_{\ell-1},s,i_{\ell+1},...,i_k} v_{i_1}^{(1)} \cdots v_{i_{\ell-1}}^{(\ell-1)} v_{i_{\ell+1}}^{(\ell+1)} \cdots v_{i_k}^{(k)}$
    Set $v_s = \frac{v_s}{\|v_s\|}$
  end for
  Set $\lambda_{\text{prev}} = \lambda$
  Set $\lambda = \sum_{i_1,...,i_k=1}^{d} T_{i_1,...,i_k} v_{i_1}^{(1)} \cdots v_{i_k}^{(1)}$
end while
Set $v = v^{(1)}$
Return v, λ.

the ability to find the eigenpair associated to the largest eigenvalue (in absolute value) so a guaranteed way to find an initial condition in the basin of the largest eigenvalue is still an important problem for future research.

Chapter 3: Approximate CP Decomposition

In this chapter, we show how tensors eigenvectors can be used to form an approximate CP decomposition up to an arbitrary level of precision. Of course, this is not a method of finding the minimal CP decomposition, the computation of which is NP-complete [20, 21]. Moreover, we do not even see an exact CP decomposition. Instead, given an order-k tensor T, we seek a sequence of vectors v_ℓ and constants λ_ℓ such that $\sum_{\ell=1}^{p} \lambda_\ell v_\ell^{\otimes k}$ approximates T in the Frobenius norm up to an error that can be made arbitrarily small by increasing p. In the next section we will show that this approximate CP decomposition is a key component for generalizing the unscented ensemble to higher moments.

Our approach is motivated by a theorem of [22] which states that if v is the unit length eigenvector of an order-k tensor T associated to the largest eigenvalue λ (in absolute value), then $\lambda v^{\otimes k}$ is the best rank-1 approximation of T, namely

$$\|T - \lambda v^{\otimes k}\|$$

is minimized over all possible λ, $\|v\| = 1$. It is well known that subtracting the best rank-1 approximation does not produce an *exact* CP decomposition, and in fact may increase tensor rank [30, 31]. However, it was suggested in [22] that repeatedly subtracting the rank-1 approximations may result in an *approximate* CP decomposition. Theorem 2 below will show that this process converges subject to a certain tensor eigenvalue inequality that will be shown in Lemma 3 below.

Originally ([1]) our proof for Theorem 2 required an inequality of the form

$$\lambda_{maxabs} \geq |T_{i_1,\ldots,i_k}|. \tag{3.1}$$

In the case of symmetric matrices, the inequality (3.1) holds since if $T \in \mathbb{R}^{d^2}$ is symmetric, it has an orthogonal eigendecomposition, $T = U^\top \Lambda U$, so by the Cauchy-Schwarz inequality,

$$|T_{ij}| = |\langle u_i, \lambda_j u_j \rangle| \leq ||u_i||\,||\lambda_j u_j|| = |\lambda_j| \leq \lambda_{maxabs} \tag{3.2}$$

where u_k is an eigenvector of T and λ_k is the associated eigenvalue for $1 \leq k \leq d$, and the identity matrix shows that $c = 1$ is the best possible constant for matrices. Naturally, this method of proof cannot be generalized to arbitrary tensors due to the lack of a similar rank-1 eigendecomposition. However, the following result of Banach [34] will allow us to prove the inequality (3.1) for all tensors.

Corollary 1 (Banach, [34])**.** *If $T(v_1, ..., v_k)$ is a symmetric k-linear tensor, then*

$$\sup_{\|v_1\| \leq 1, ..., \|v_k\| \leq 1} |T(v_1, ..., v_k)| = \sup_{\|v\| \leq 1} |T(v, ..., v)|.$$

In fact, Corollary 1 is a special case of Banach's Satz I in [34] when applied to scalar valued tensors (the full result applies to L^2-function valued tensors). This allows us to prove the following lemma for all symmetric tensors.

Lemma 3. *For all symmetric k-order tensors T with largest eigenvalue in absolute value λ_{maxabs}, then*

$$\lambda_{maxabs} \geq |T_{i_1, ..., i_k}|. \tag{3.3}$$

Proof. By Corollary 1, we have:

$$\lambda_{maxabs} = \sup_{||v||=1} |T(v, ..., v)| = \sup_{||v_1||=...=||v_k||=1} |T(v_1, ..., v_k)| \geq |T(e_{i_1}, ..., e_{i_k})| = |T_{i_1, ..., i_k}|$$

□

We can now prove that repeatedly subtracting rank-1 yields an approximate CP decomposition to any desired precision.

Theorem 2. *Let T be a k-order symmetric tensor with size d, i.e. $T \in \mathbb{R}^{d^k}$. Consider the process of finding an approximate CP decomposition of T by starting from $T_0 = T$ and setting $T_{\ell+1} = T_\ell - \lambda_\ell v_\ell^{\otimes k}$ where λ_ℓ is the largest eigenvalue in absolute value of T_ℓ and v_ℓ is the associated eigenvector. Then $\|T_\ell\|_F \to 0$ and for $r = \sqrt{1 - \frac{1}{d^k}} \in [0, 1)$*

$$\frac{\|T_{\ell+1}\|_F}{\|T_\ell\|_F} \leq r \quad \textit{and} \quad T = \sum_{\ell=1}^{p} \lambda_\ell v_\ell^{\otimes k} + \mathcal{O}(r^L)$$

for all $L \in \mathbb{N}$.

Proof. First let λ_{maxabs} be the largest eigenvalue in absolute value of a tensor T and recall $\lambda_{maxabs} \geq |T_{i_1 \ldots i_k}|$ for all $i_1, \ldots, i_k$. We will show that there exists a constant $c = \frac{1}{d^{k/2}} \in (0, 1]$ such that $\lambda_{maxabs} \geq c\|T\|_F$. Since $\lambda_{maxabs} \geq |T_{i_1 \ldots i_k}|$, we have

$$\lambda_{maxabs}^2 \geq T_{i_1 \ldots i_k}^2$$

which implies that

$$d^k \lambda_{maxabs}^2 \geq \sum_{i_1, \ldots, i_k} T_{i_1 \ldots i_k}^2$$

so we have $d^{k/2}\lambda_{maxabs} \geq \sqrt{\sum_{i_1, \ldots, i_k} T_{i_1 \ldots i_k}^2}$ and

$$\lambda_{maxabs} \geq \frac{1}{d^{k/2}} \|T\|_F, \tag{3.4}$$

where we take $c = \dfrac{1}{d^{k/2}} \in (0, 1]$, since $d \geq 1$.

By Lemma 2 applied to T_ℓ, we have

$$\|T_{\ell+1}\|_F{}^2 = \|T_\ell - \lambda_\ell v_\ell^{\otimes k}\|_F{}^2 = \|T_\ell\|_F{}^2 - \lambda_\ell{}^2.$$

Since λ_ℓ is defined to be the largest eigenvalue of T_ℓ, (3.4) says that $\lambda_\ell \geq c\|T_\ell\|_F$ where $c = \frac{1}{d^{k/2}}$ so

$$\begin{aligned} \|T_{\ell+1}\|_F{}^2 &\leq \|T_\ell\|_F{}^2 - c^2\|T_\ell\|_F{}^2 \\ &\leq (1-c^2)\|T_\ell\|_F{}^2. \end{aligned}$$

Thus, setting $r = \sqrt{1-c^2} \in [0,1)$ we have $\|T_{\ell+1}\|_F \leq r\|T_\ell\|_F$ and $\|T_{\ell+1}\|_F \leq r^2\|T_{\ell-1}\|_F$ and so forth and proceeding inductively we find,

$$\|T_{\ell+1}\|_F \leq r^{\ell+1}\|T_0\|_F = r^{\ell+1}\|T\|_F.$$

Since $0 < r < 1$, $\lim_{\ell\to\infty} r^{\ell+1} = 0$, so $0 \leq \|T_{\ell+1}\|_F \leq r^{\ell+1}\|T\|_F \to 0$ implies $\|T_{\ell+1}\| \to 0$ as $\ell \to \infty$. Since this limit is 0, an upper bound on the rate of convergence of $\|T_\ell\|_F$ is found by considering

$$\frac{\|T_{\ell+1}\|_F}{\|T_\ell\|_F} \leq r = \sqrt{1 - \frac{1}{d^k}}.$$

□

Theorem 2 gives an effective algorithm for finding approximate CP decompositions of tensors. Note that this theorem can be improved using a result of [35] in which we define

$$\lambda_{maxabs} = ||T||_2 \geq \mathrm{App}_k(\mathbb{R}, d, ..., d)||T||_F$$

where $\mathrm{App}_k(\mathbb{R}, d, ..., d)$ is the best rank-one approximation ratio. When $d = 1, 2, 4, 8$, we

have $\text{App}_k(\mathbb{R}, d, ..., d) = d^{-\frac{k-1}{2}}$ but for all other d we have $\text{App}_k(\mathbb{R}, d, ..., d) > d^{-\frac{k-1}{2}}$, thus

$$\lambda_{maxabs} \geq d^{-\frac{k-1}{2}} \|T\|_F \tag{3.5}$$

which is the best inequality that holds for all d, although better inequalities may hold for $d \neq 1, 2, 4, 8$. Thus, by replacing (3.4) with (3.5), the Theorem 2 can be rewritten as follows with the best possible bound for r.

Theorem 3. *Let T be a k-order symmetric tensor with size d, i.e. $T \in \mathbb{R}^{d^k}$. Consider the process of finding an approximate CP decomposition of T by starting from $T_0 = T$ and setting $T_{\ell+1} = T_\ell - \lambda_\ell v_\ell^{\otimes k}$ where λ_ℓ is the largest eigenvalue in absolute value of T_ℓ and v_ℓ is the associated eigenvector. Then $\|T_\ell\|_F \to 0$ and for $r = \sqrt{1 - d^{1-k}} \in (0, 1)$*

$$\frac{\|T_{\ell+1}\|_F}{\|T_\ell\|_F} \leq r \quad \textit{and} \quad T = \sum_{\ell=1}^{p} \lambda_\ell v_\ell^{\otimes k} + \mathcal{O}(r^L)$$

for all $L \in \mathbb{N}$.

In the next chapter, we will show how to use the approximate CP decomposition to build an ensemble that simultaneously matches the mean, covariance, skewness and kurtosis.

We summarize the approximate CP decomposition algorithm below.

Algorithm 2 Approximate CP Decomposition

Inputs: A k-tensor $T \in \mathbb{R}^{d^k}$ and a tolerance τ.

Outputs: Vectors, v_ℓ, and signs, $s_\ell \in \{-1, 1\}$ such that $\left|\left|\sum_{\ell=1}^{p} s_\ell v_\ell^{\otimes k} - T\right|\right|_F \leq \tau$.

Set $\ell = 1$

while $||T||_F > \tau$ **do**

Apply the HOPM (Algorithm 1) to find an eigenpair (v, λ) of T.

Set $s_\ell = \text{sign}(\lambda)$ (note that if k is odd we can always choose $s_\ell = 1$)

Set $v_\ell = |\lambda|^{1/k} v$

Set $T = T - s_\ell v_\ell^{\otimes k}$

Set $\ell = \ell + 1$

end while

Return the set of all s_ℓ, v_ℓ.

Finally, we demonstrate this algorithm on a random 3-tensor and 4-tensor with $d = 2$ and $d = 10$ in Figure 3.1. We note that in all cases the convergence is much faster than our theoretical upper bound, however for $d = 10$ we see that the ratio of residual norms approaches much closer to our upper bound. Moreover, high dimensional tensors require a much larger number of vectors to achieve a given tolerance with the approximate CP decomposition. So while our approach provides an effective solution, it is likely that there is room for improvement, and the higher order unscented transform (HOUT) introduced in the next section can use any method of CP decomposition.

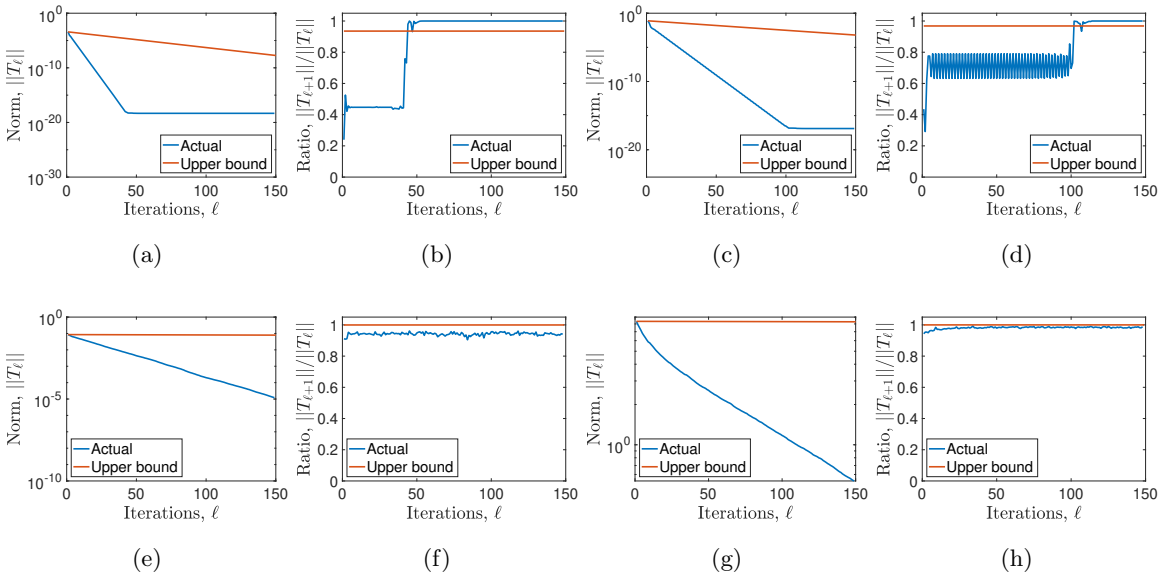


Figure 3.1: Top (a-d): With $d = 2$ we demonstrate the convergence rate of the approximate CP decomposition of a random symmetric 3-tensor (a,b) and 4-tensor (c,d). The norm of the residual in blue (a,c) decays to numerical zero faster than the upper bound, r^ℓ (red). The ratio of successive Frobenius norms shown in blue (b,d) respect the derived upper bound r (red) as long as the difference between iterations exceeds numerical precision. However, it has not really converged to the true value, it has just converged up to 16 digits of precision. Notice how the blue line in (b) violates the upper bound right before iteration 50, and at the same iteration in (a), the error levels off near numerical precision (10^{-16}). Bottom (e-h): We repeat the experiment with $d = 10$.

3.1 Sharpness

Now we wish to show the sharpness of Lemma 3, i.e. demonstrate that the inequality (3.3) is optimal. We first show that without loss of generality we only need to show sharpness for 2^k tensors.

Lemma 4. *If there is an example of a symmetric tensor $T \in \mathbb{R}^{2^k}$ such that the inequality is shown to be sharp, then the inequality is also sharp for $\mathbb{R}^{n^k}$.*

Proof. Suppose there is a symmetric tensor $T \in \mathbb{R}^{2^k}$ that demonstrates sharpness of the Lemma 3. We can extend this to the n^k case where entries with indices that consist of ones and twos are the same as the 2^k case and all other entries 0. This yields the following

equations:

$$\begin{aligned}
\lambda u_1 &= (T \times_2 u \times_3 u)_1 = \sum_{j,k=1}^{n} T_{1jk\ell} u_j u_k u_\ell \\
\lambda u_2 &= (T \times_2 u \times_3 u)_2 = \sum_{j,k=1}^{n} T_{2jk\ell} u_j u_k u_\ell \\
\lambda u_3 &= (T \times_2 u \times_3 u)_3 = \sum_{j,k=1}^{n} T_{3jk\ell} u_j u_k u_\ell = 0 \\
\lambda u_4 &= (T \times_2 u \times_3 u)_4 = \sum_{j,k=1}^{n} T_{4jk\ell} u_j u_k u_\ell = 0 \\
&\vdots \\
\lambda u_n &= (T \times_2 u \times_3 u)_3 = \sum_{j,k=1}^{n} T_{njk\ell} u_j u_k u_\ell = 0 \\
1 &= u_1^1 + u_2^2 + ... + u_n^2
\end{aligned}$$

Obviously $u_3 = u_4 = ... = u_n = 0$ and thus $u_1^2 + u_2^2 = 1$. Hence we have the same equations as in the 2^k case. Therefore, we have proved the inequality for any symmetric k-tensor. $\square$

We first consider the case of 3-tensors. By the eigenvalue equation, for some 3-tensor $T \in \mathbb{R}^{d^3}$ and eigenvector $\vec{u} \in \mathbb{R}^d$ of length 1 with associated eigenvalue λ we have

$$T \times_2 u \times_3 u = \lambda u \tag{3.6}$$

with the component-wise definition

$$(T \times_2 u \times_3 u)_i = \sum_{j,k=1}^{d} T_{ijk} u_j u_k.$$

By Lemma 4, we only need to consider tensors that are in $\mathbb{R}^{2^3}$. Consider the $2 \times 2 \times 2$ symmetric tensor T with entries $T_{111} = 0, T_{211} = 1, T_{221} = 0, T_{222} = -1$ so

$$T = \left[\begin{array}{cc} T_{111} & T_{121} \\ T_{211} & T_{221} \end{array}\right] \diagdown \left[\begin{array}{cc} T_{112} & T_{122} \\ T_{212} & T_{222} \end{array}\right] = \left[\begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array}\right] \diagdown \left[\begin{array}{cc} 1 & 0 \\ 0 & -1 \end{array}\right]$$

and eigenvector of length 1 $\vec{u} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$. By plugging these values into the eigenvalue equation and taking the first component of both sides of (3.6) we have the following equality

$$\lambda u_1 = (T \times_2 u \times_3 u)_1 = \sum_{j,k=1}^{2} T_{1jk} u_j u_k = T_{111} u_1^2 + T_{112} u_1 u_2 + T_{121} u_2 u_1 + T_{122} u_2^2 = 2u_1 u_2$$

Now, taking the second component we have

$$\lambda u_2 = (T \times_2 u \times_3 u)_2 = \sum_{j,k=1}^{2} T_{2jk} u_j u_k = T_{211} u_1^2 + T_{212} u_1 u_2 + T_{221} u_2 u_1 + T_{222} u_2^2 = u_1^2 - u_2^2$$

Therefore, we have the system of equations

$$\lambda u_1 = 2u_1 u_2 \tag{3.7}$$

$$\lambda u_2 = u_1^2 - u_2^2 \tag{3.8}$$

$$u_1^2 + u_2^2 = 1 \tag{3.9}$$

Rearranging (3.7) gives $\lambda = 2u_2$. Plugging this value of λ into (3.8) then gives $u_1^2 = 3u_2^2$. Plugging this into (3.9) and solving the equation yields $\lambda = \pm 1$.

Therefore, since $\lambda_{maxabs} = 1$ and is greater than or equal to the absolute value of each

entry in the tensor, this proves that in the inequality $\lambda_{maxabs} \geq c \cdot T_{stu}$, c cannot have a value greater than 1, proving the sharpness of our inequality for the 3-tensor.

We next consider the case of 4-tensors. By Lemma 4, we only need to consider tensors that are in $\mathbb{R}^{2^4}$. Note that by the eigenvalue equation, for some $T \in \mathbb{R}^{d^4}$ and eigenvector $\vec{u} \in \mathbb{R}^d$ of length 1 with associated eigenvalue λ we have

$$T \times_2 u \times_3 u \times_4 u = \lambda u \tag{3.10}$$

with the component-wise definition

$$(T \times_2 u \times_3 \times_4 u)_i = \sum_{j,k,\ell=1}^{d} T_{ijk} u_j u_k u_\ell.$$

Now, consider the $2 \times 2 \times 2 \times 2$ symmetric tensor T with entries $T_{1111} = T_{2222} = 3$, $T_{1122} = 1$, and $T_{1112} = T_{1222} = 0$. We then take the first component of both sides of (3.10) to get the following equality

$$\begin{aligned}
\lambda u_1 &= (T \times_2 u \times_3 u \times_4 u)_1 \\
&= \sum_{j,k=1}^{2} T_{1jk\ell} u_j u_k u_\ell \\
&= T_{1111}u_1^3 + T_{1211}u_1^2u_2 + T_{1121}u_1^2u_2 + T_{1112}u_1^2u_2 + T_{1221}u_1u_2^2 + T_{1212}u_1u_2^2 \\
&\quad + T_{1122}u_1u_2^2 + T_{1222}u_2^3 \\
&= T_{1111}u_1^3 + (T_{1211} + T_{1121} + T_{1112})u_1^2u_2 + (T_{1221} + T_{1212} + T_{1122})u_1u_2^2 + T_{1222}u_2^3 \\
&= 3u_1^3 + 3u_1u_2^2 \\
&= 3(u_1^3 + u_1u_2^2).
\end{aligned}$$

Now, taking the second component we have

$$\begin{aligned}
\lambda u_2 &= (T \times_2 u \times_3 u \times_4 u)_2 \\
&= \sum_{j,k=1}^{2} T_{2jk\ell} u_j u_k u_\ell \\
&= T_{2111}u_1^3 + T_{2211}u_1^2u_2 + T_{2112}u_1^2u_2 + T_{2121}u_1^2u_2 + T_{2212}u_1u_2^2 + T_{2221}u_1u_2^2 \\
&\quad + T_{2122}u_1u_2^2 + T_{2222}u_2^3 \\
&= T_{2111}u_1^3 + (T_{2211} + T_{2112} + T_{2121})u_1^2u_2 + (T_{2212} + T_{2221} + T_{2122})u_1u_2^2 + T_{2222}u_2^3 \\
&= 3u_1^2u_2 + 3u_2^3 \\
&= 3(u_1^2u_2 + u_2^3).
\end{aligned}$$

Therefore, we have the system of equations

$$\lambda u_1 = 3(u_1^3 + u_1u_2^2) \tag{3.11}$$

$$\lambda u_2 = 3(u_1^2u_2 + u_2^3) \tag{3.12}$$

$$u_1^2 + u_2^2 = 1 \tag{3.13}$$

Rearranging (3.13) as $u_2^2 = 1 - u_1^2$ and plugging this into (3.11) gives us $\lambda u_1 = 3u_1$ hence $\lambda = 3$. Then we rearrange (3.13) as $u_1^2 = 1 - u_2^2$ and plug this into (3.12) to give us $\lambda u_2 = 3u_2$ thus once again $\lambda = 3$. Therefore $\lambda_{maxabs} = 3$ and satisfies the Lemma 3

$$\lambda_{maxabs} \geq |T_{ijk\ell}|$$

since the largest entry in T is also 3, proving the sharpness of the lemma for the 4-tensor.

Chapter 4: Higher Order Unscented Transform

4.1 Higher Order Unscented Transform

The goal of the scaled unscented transform is to generate a small ensemble that exactly matches the mean and covariance of a distribution, thus forming a quadrature rule that can be to estimate the expected value of nonlinear functions. In this section we define the higher order unscented transform which matches the first four moments of a distribution, thus providing a quadrature rule with a higher degree of exactness. While we only describe the process explicitly for up to four moments, our method is based on the approximate tensor decomposition from the previous section and should allow generalization to an arbitrary number of moments.

Suppose we are given the following moments of the distribution of a random variable: the mean $\mu \in \mathbb{R}^d$, the covariance matrix $C \in \mathbb{R}^{d\times d}$, the skewness tensor $S \in \mathbb{R}^{d\times d\times d}$, and kurtosis tensor $K \in \mathbb{R}^{d\times d\times d\times d}$. Let τ be a parameter that specifies the tolerance of the approximate CP decompositions and let S and K have the approximate CP decompositions

$$\left|\left|S - \sum_{i=1}^{J} \tilde{v_i}^{\otimes 3}\right|\right|_F \leq \tau/2 \qquad \left|\left|K - \sum_{i=1}^{L} s_i \tilde{u_i}^{\otimes 4}\right|\right|_F \leq \tau/2$$

where $s_i \in \{-1, 1\}$ denote signs. Note that these approximate decompositions can be constructed by the algorithm described in Theorem 2 and then moving the eigenvalues inside the tensor power by the rule $(cv)^{\otimes k} = c^k v^{\otimes k}$. Note that the signs s_i are required for the kurtosis since constants come out of even order tensor powers as absolute values.

The key to forming an ensemble that matches all four moments simultaneously is carefully balancing the interactions between the moments. For example, if we add new quadrature nodes of the form $\mu + \gamma\tilde{v}_i$ in order to try to match the skewness, these nodes will influence the mean of the ensemble. In order to balance these interactions we make the following definitions based on the approximate CP decompositions of the skewness and kurtosis,

$$\tilde{\mu} = \sum_{i=1}^{J} \tilde{v}_i, \qquad \hat{\mu} = -\gamma^{-2}\tilde{\mu}, \qquad \tilde{C} = \sum_{i=1}^{L} s_i \tilde{u_i}^{\otimes 2}, \qquad \hat{C} = C - \frac{1}{\delta^2}\tilde{C}$$

where $\hat{L} = \sum_{i=1}^{L} s_i$ and β, γ, δ are arbitrary positive constants that will define the 4 moment σ-points below. We note that C is assumed symmetric and positive definite since it is a covariance matrix and $\tilde{C}$ is symmetric by definition. In order to insure that $\hat{C}$ is also positive definite, let $\lambda_{\max}^{\tilde{C}}$ be the largest eigenvalue of $\tilde{C}$ and let $\lambda_{\min}^{C}$ be the smallest eigenvalue of C, then we require that $\delta > \sqrt{\frac{\lambda_{\max}^{\tilde{C}}}{\lambda_{\min}^{C}}}$ which guarantees that $\hat{C}$ is positive definite. We note that this choice can be overly conservative especially when C is close to rank deficient. In these cases, it can be helpful to iterative divide δ by 2 as long as $\hat{C}$ remains positive definite. These choices balance out the interactions between the moments and are the key to proving Theorem 4 below. We are now ready to define the 4-moment σ-points.

Definition 11 (The 4-moment σ–points of the higher order unscented transform)**.** *Let* α*,*

β, γ, δ be positive real numbers, we define the **4 moment $\boldsymbol{\sigma}$-points** *by*

$$\sigma_i = \begin{cases} \mu & \text{if } i = -2 \\ \mu + \alpha\hat{\mu} & \text{if } i = -1 \\ \mu - \alpha\hat{\mu} & \text{if } i = 0 \\ \mu + \beta\sqrt{\hat{C}}_i & \text{if } i = 1, \ldots, d \\ \mu - \beta\sqrt{\hat{C}}_{i-d} & \text{if } i = d+1, \ldots, 2d \\ \mu + \gamma\tilde{v}_{i-2d} & \text{if } i = 2d+1, \ldots, 2d+J \\ \mu - \gamma\tilde{v}_{i-2d-J} & \text{if } i = 2d+J+1, \ldots, 2d+2J \\ \mu + \delta\tilde{u}_{i-2d-2J} & \text{if } i = 2d+2J+1, \ldots, 2d+2J+L \\ \mu - \delta\tilde{u}_{i-2d-2J-L} & \text{if } i = 2d+2J+L+1, \ldots, N \end{cases}$$

and the corresponding **weights** *by*

$$w_i = \begin{cases} 1 - d\beta^{-2} - \hat{L}\delta^{-4} & \text{if } i = -2 \\ \frac{1}{2}\alpha^{-1} & \text{if } i = -1 \\ -\frac{1}{2}\alpha^{-1} & \text{if } i = 0 \\ \frac{1}{2}\beta^{-2} & \text{if } i = 1, \ldots, 2d \\ \frac{1}{2}\gamma^{-3} & \text{if } i = 2d+1, \ldots, 2d+J \\ -\frac{1}{2}\gamma^{-3} & \text{if } i = 2d+J+1, \ldots, 2d+2J \\ \frac{1}{2}\delta^{-4}s_{i-2d-2J} & \text{if } i = 2d+2J+1, \ldots, 2d+2J+L \\ \frac{1}{2}\delta^{-4}s_{i-2d-2J-L} & \text{if } i = 2d+2J+L+1, \ldots, N \end{cases}$$

For convenience, denote $N = 2(d + J + L)$.

The next theorem shows that the 4-moment σ-points match the first two moments exactly and match the skewness and kurtosis up to an error term that can be controlled below.

Theorem 4. *Given the 4-moment σ-points associated with μ, C, S, and K we have $\sum_{i=-2}^{N} w_i = 1$ and*

$$\begin{aligned}
\sum_{i=-2}^{N} w_i \sigma_i &= \mu \\
\sum_{i=-2}^{N} w_i (\sigma_i - \mu)^{\otimes 2} &= C \\
\left|\left| \sum_{i=-2}^{N} w_i (\sigma_i - \mu)^{\otimes 3} - S \right|\right|_F &\leq \tau/2 + \alpha^2 \left|\left| \hat{\mu}^{\otimes 3} \right|\right|_F \\
\left|\left| \sum_{i=-2}^{N} w_i (\sigma_i - \mu)^{\otimes 4} - K \right|\right|_F &\leq \tau/2 + \beta^2 \left|\left| \bar{C} \right|\right|_F .
\end{aligned}$$

where $\bar{C} = \sum_{i=1}^{d} \sqrt{\hat{C}_i}^{\otimes 4}$.

Proof. We first we wish to show that the first moment equation matches our mean. We begin by splitting the sum

$$\sum_{i=-2}^{N} w_i \sigma_i = \sum_{i=-2}^{0} w_i \sigma_i + \sum_{i=1}^{2d} w_i \sigma_i + \sum_{i=2d+1}^{2d+2J} w_i \sigma_i + \sum_{i=2d+2J+1}^{N} w_i \sigma_i$$

Using the expressions defining the 4 moment σ-points σ_i and the corresponding weights w_i,

we have

$$\begin{aligned}
\sum_{i=-2}^{N} w_i\sigma_i &= (1-d\beta^{-2}-\hat{L}\delta^{-4})\mu + \frac{1}{2\alpha}(\mu+\alpha\hat{\mu}) - \frac{1}{2\alpha}(\mu-\alpha\hat{\mu}) \\
&+ \sum_{i=1}^{d}\frac{1}{2\beta^2}\big(\mu+\beta\sqrt{\hat{C}_i}\big) + \sum_{j=d+1}^{2d}\frac{1}{2\beta^2}\big(\mu-\beta\sqrt{\hat{C}}_{i-d}\big) \\
&+ \sum_{i=2d+1}^{2d+J}\frac{1}{2\gamma^3}\big(\mu+\gamma\tilde{v}_{i-2d}\big) + \sum_{j=2d+J+1}^{2d+2J}\frac{-1}{2\gamma^3}\big(\mu-\gamma\tilde{v}_{i-2d-J}\big) \\
&+ \sum_{i=2d+2J+1}^{2d+2J+L}\frac{1}{2\delta^4}s_{i-2d-2J}\big(\mu+\delta\tilde{u}_{i-2d-2J}\big) \\
&+ \sum_{j=2d+2J+L+1}^{N}\frac{1}{2\delta^4}s_{i-2d-2J-L}\big(\mu-\delta\tilde{u}_{i-2d-2J-L}\big)
\end{aligned}$$

and regrouping like terms, we obtain

$$\begin{aligned}
\sum_{i=-2}^{N} w_i\sigma_i &= (1-d\beta^{-2}-\hat{L}\delta^{-4})\mu + \hat{\mu} + \sum_{i=1}^{d}\frac{1}{2\beta^2}\big(2\mu+\beta\sqrt{\hat{C}_i}-\beta\sqrt{\hat{C}_i}\big) \\
&\quad + \sum_{i=1}^{J}\left(\frac{1}{2\gamma^3}\big(\mu+\gamma\tilde{v}_i\big) - \frac{1}{2\gamma^3}\big(\mu-\gamma\tilde{v}_i\big)\right) + \sum_{i=1}^{L}\frac{1}{2\delta^4}s_i\big(2\mu+\delta\tilde{u}_i-\delta\tilde{u}_i\big) \\
&= (1-d\beta^{-2}-\hat{L}\delta^{-4})\mu + \hat{\mu} + \sum_{i=1}^{d}\frac{\mu}{\beta^2} + \sum_{i=1}^{J}\frac{\tilde{v}_i}{\gamma^2} + \sum_{i=1}^{L}\frac{s_i\mu}{\delta^4} \\
&= (1-d\beta^{-2}-\hat{L}\delta^{-4})\mu + \hat{\mu} + d\beta^{-2}\mu + \gamma^{-2}\sum_{i=1}^{J}\tilde{v}_i + \delta^{-4}\mu\sum_{i=1}^{L}s_i \\
&= \mu + \hat{\mu} + \gamma^{-2}\tilde{\mu} \\
&= \mu
\end{aligned}$$

using the definition $\hat{\mu} = -\gamma^{-2}\tilde{\mu}$ for the last equality.

To look at the other moment equations, let's first observe that for $n = 2, 3, 4$,

$$\begin{aligned}\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes n} &= \sum_{i=-1}^{0} w_i(\sigma_i - \mu)^{\otimes n} + \sum_{i=1}^{2d} w_i(\sigma_i - \mu)^{\otimes n} \\ &\quad + \sum_{i=2d+1}^{2d+2J} w_i(\sigma_i - \mu)^{\otimes n} + \sum_{i=2d+2J+1}^{N} w_i(\sigma_i - \mu)^{\otimes n}\end{aligned}$$

Notice that since the first σ-point σ_{-2} is μ, the term $w_{-2}(\sigma_{-2} - \mu)^{\otimes n} = 0$. By the definition of σ-points and corresponding weights,

$$\begin{aligned}\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes n} &= \frac{\alpha^{n-1}}{2}\left(\hat{\mu}^{\otimes n} - (-\hat{\mu})^{\otimes n}\right) \\ &\quad + \frac{\beta^{n-2}}{2}\left(\sum_{i=1}^{d}\big(\sqrt{\hat{C}}_i\big)^{\otimes n} + \sum_{j=d+1}^{2d}\big(-\sqrt{\hat{C}}_{i-d}\big)^{\otimes n}\right) \\ &\quad + \frac{\gamma^{n-3}}{2}\left(\sum_{i=2d+1}^{2d+J}\big(\tilde{v}_{i-2d}\big)^{\otimes n} - \sum_{j=2d+J+1}^{2d+2J}\big(-\tilde{v}_{i-2d-J}\big)^{\otimes n}\right) \\ &\quad + \frac{\delta^{n-4}}{2}\sum_{i=2d+2J+1}^{2d+2J+L} s_{i-2d-2J}\big(\tilde{u}_{i-2d-2J}\big)^{\otimes n} \\ &\quad + \frac{\delta^{n-4}}{2}\sum_{j=2d+2J+L+1}^{N} s_{i-2d-2J-L}\big(-\tilde{u}_{i-2d-2J-L}\big)^{\otimes n}.\end{aligned}$$

where we used the property $(av)^{\otimes n} = a^n v^{\otimes n}$ where a is any real number and v is a vector.

When n is even, we have

$$\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes n} = \beta^{n-2} \sum_{i=1}^{d} \left(\sqrt{\hat{C}_i}\right)^{\otimes n} + \delta^{n-4} \sum_{i=1}^{L} s_i \left(\tilde{u}_i\right)^{\otimes n} \tag{4.1}$$

and when n is odd, we obtain

$$\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes n} = \alpha^{n-1} \hat{\mu}^{\otimes n} + \gamma^{n-3} \sum_{i=1}^{J} \left(\tilde{v}_i\right)^{\otimes n}. \tag{4.2}$$

Now we wish to show that the second moment equation matches our covariance. By (4.1), setting $n = 2$ we have

$$\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes 2} = \sum_{i=1}^{d} \sqrt{\hat{C}_i}^{\otimes 2} + \delta^{-2} \sum_{i=1}^{L} s_i \tilde{u}_i^{\otimes 2} = \hat{C} + \delta^{-2} \tilde{C}$$

and applying the definition of $\hat{C} = C - \delta^{-2}\tilde{C}$ we have

$$\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes 2} = C,$$

as desired. Next, observe that by (4.2) and the definition of S,

$$\sum_{j=0}^{N} w_i(\sigma_i - \mu)^{\otimes 3} \quad = \quad \alpha^2 \hat{\mu}^{\otimes 3} + \sum_{i=1}^{J} \tilde{v}_i^{\otimes 3}$$

and since we assume that $\left|\left|\sum_{i=1}^{J} \tilde{v}_i^{\otimes 3} - S\right|\right|_F \leq \frac{\tau}{2}$, by the triangle inequality we have,

$$\left|\left|\sum_{j=0}^{N} w_i(\sigma_i - \mu)^{\otimes 3} - S\right|\right|_F \leq \frac{\tau}{2} + \alpha^2 ||\hat{\mu}^{\otimes 3}||_F$$

as desired. Lastly, we wish to show that the fourth moment equation matches our kurtosis. By (4.1) and the definition of K, we have

$$\begin{aligned} \sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes 4} &= \beta^2 \sum_{i=1}^{d} \sqrt{\hat{C}_i}^{\otimes 4} + \sum_{i=1}^{L} s_i \tilde{u}_i^{\otimes 4} \\ &= \beta^2 \bar{C} + \sum_{i=1}^{L} s_i \tilde{u}_i^{\otimes 4} \end{aligned}$$

and since we assume that $\left|\left|\sum_{i=1}^{L} s_i \tilde{u}_i^{\otimes 4} - K\right|\right|_F \leq \frac{\tau}{2}$, by the triangle inequality we have,

$$\left|\left|\sum_{j=0}^{N} w_i(\sigma_i - \mu)^{\otimes 4} - K\right|\right|_F \leq \frac{\tau}{2} + \beta^2 ||\bar{C}||_F$$

which completes the proof. □

Notice that the third and fourth moment equations do not exactly match the skewness and kurtosis, respectively. Of course, we only used an approximate CP decomposition to begin with, which accounts for the τ term in the error. Thus, the real goal is to bound the other error term by the same tolerance, τ. The following corollary shows how to control the error terms on the skewness and kurtosis.

Corollary 2. *Let τ be a specified tolerance for the absolute error of the skewness and kurtosis and set $\bar{C} = \sum_{i=1}^{d} \sqrt{\hat{C}_i}^{\otimes 4}$ and $\hat{\mu}$ as in Theorem 4. If we choose parameters α, β such*

that

$$\alpha < \sqrt{\frac{\tau}{2||\hat{\mu}^{\otimes 3}||_F}} \qquad \text{and} \qquad \beta < \sqrt{\frac{\tau}{2||\overline{C}||_F}}$$

then

$$\left|\left|\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes 3} - S\right|\right|_F < \tau \qquad \text{and} \qquad \left|\left|\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes 4} - K\right|\right|_F < \tau.$$

Proof. The inequality for β follows immediately from Theorem 4. Once β is chosen, then we can define,

$$||\hat{\mu}^{\otimes 3}||_F = \left|\left|\left(\left(1 - d\beta^{-2} - \hat{L}\delta^{-4}\right)\mu - \gamma^{-2}\tilde{\mu}\right)^{\otimes 3}\right|\right|_F$$

and choosing $\alpha < \sqrt{\frac{\tau}{2||\hat{\mu}^{\otimes 3}||_F}}$ we have

$$\left|\left|\sum_{i=-2}^{N} w_i(\sigma_i - \mu)^{\otimes 3} - S\right|\right|_F \leq \tau/2 + \alpha^2||\hat{\mu}^{\otimes 3}||_F < \tau$$

as desired. □

Corollary 2 could easily be reformulated to control relative error if desired, and taken to the extreme we could make the quadrature rule exact up to numerical precision. As a practical matter, this is not an effective strategy since it would result in a larger condition number for the numerical quadrature as shown in the following remark.

Algorithm 3 Higher Order Unscented Transform (HOUT)

Inputs: A function f, tolerance τ, and the mean, μ, covariance, C, skewness, S, and kurtosis, K, of a random variable X.

Outputs: Estimate of $\mathbb{E}[f(X)]$ with degree of exactness 4.

Compute the approximate CP decomposition $\left|\left|S - \sum_{i=1}^{J} \tilde{v}_i^{\otimes 3}\right|\right|_F \leq \tau/2$

Compute the approximate CP decomposition $\left|\left|K - \sum_{i=1}^{L} s_i \tilde{u}_i^{\otimes 4}\right|\right|_F \leq \tau/2$

Set $\tilde{C} = \sum_{i=1}^{L} s_i \tilde{u}_i^{\otimes 2}$.

Compute the largest eigenvalue $\lambda_{\max}^{\tilde{C}}$ of $\tilde{C}$ and the smallest eigenvalue $\lambda_{\min}^{C}$ of C

Choose $\delta > \sqrt{\frac{\lambda_{\max}^{\tilde{C}}}{\lambda_{\min}^{C}}}$ (note that C is positive definite so $\lambda_{\min}^{C} > 0$)

(Optional) While $C - \delta^{-2}\tilde{C}$ is positive definite, set $\delta = \delta/2$

Set $\hat{C} = C - \delta^{-2}\tilde{C}$

Compute the symmetric square root of $\hat{C}$ with columns $\sqrt{\hat{C}_i}$

Set $\bar{C} = \sum_{i=1}^{d} \sqrt{\hat{C}_i}^{\otimes 4}$

Choose $\beta < \sqrt{\frac{\tau}{2||\bar{C}||_F}}$ and choose $\gamma > 0$ (default $\gamma = J^{-1/3}$)

Set $\hat{L} = \sum_{i=1}^{L} s_i$ and $\tilde{\mu} = \sum_{i=1}^{J} \tilde{v}_i$ and $\hat{\mu} = (1 - d\beta^{-2} - \hat{L}\delta^{-4})\mu - \gamma^{-2}\tilde{\mu}$

Choose $\alpha < \sqrt{\frac{\tau}{2||\hat{\mu}^{\otimes 3}||_F}}$

Define the 4-moment σ-points, σ_i, and weights, w_i, according to Definition 11

Output: $\sum_{i=-2}^{N} w_i f(\sigma_i)$

Remark 2. *The absolute condition number of the higher order unscented transform is bounded above by* $\sum_{i=0}^{N} |w_i|$. *Using the bounds from Corollary 2 we find*

$$\sum_{i=0}^{N} |w_i| = \frac{1}{\alpha} + \frac{d}{\beta^2} + \frac{J}{\gamma^3} + \frac{L}{\delta^4} > \sqrt{\frac{||\bar{\mu}^{\otimes 3}||_F}{\tau}} + \frac{d||\bar{C}||_F}{\tau} + \frac{J}{\gamma^3} + \frac{L}{\delta^4} = \mathcal{O}(\tau^{-1})$$

which shows that the condition number has the potential to blow up as the tolerance is decreased.

We summarize the HOUT algorithm in Algorithm 3 and we now turn to some numerical experiments to demonstrate the HOUT.

4.2 Error Analysis

The standard approach to error estimates for the SUT is based on a Taylor's theorem approximation near the mean. These results can be immediately generalized to the HOUT as in the following theorem.

Theorem 5 (Taylor-type HOUT Error Bound)**.** *Let $f \in C^5(\mathbb{R}^d, \mathbb{R})$ and let $X \sim p$ be a random variable with distribution p that has compact support. Then the error in estimating $\mathbb{E}[f(X)]$ using the 4-moment σ–points of the HOUT and corresponding weights has the upper bound*

$$\left| \mathbb{E}[f(X)] - \sum_{i=1}^{m} w_i f(\sigma_i) \right| \leq ||D^5 f||_\infty \frac{d^5}{120} \left(||M_{5,abs}||_{\max} + ||\tilde{M}_{5,abs}||_{\max} \right)$$

where the $||D^5 f||_\infty$ is taken on the support of the measure and $M_{5,abs}, \tilde{M}_{5,abs}$ are the absolute fifth moments of p and the quadrature respectively.

Proof. Suppose $f \in C^5(\mathbb{R}^d, \mathbb{R})$. Now we wish to find the error bound where $\mathbb{E}[f(x)]$ with $x \sim p$ is the truth and $\sum_{i=1}^{m} w_i f(\sigma_i)$ is our estimate where m is the number of σ-points (nodes) in the quadrature. By Taylor's theorem with remainder we can expand f centered at μ as

$$\begin{aligned} f(x) &= f(\mu) + \nabla f(\mu)(x-\mu) + \frac{1}{2} \sum_{j,k=1}^{d} Hf(\mu)_{jk}(x-\mu)_j(x-\mu)_k \\ &+ \frac{1}{6} \sum_{j,k,l=1}^{d} D^3 f(\mu)_{jkl}(x-\mu)_j(x-\mu)_k(x-\mu)_l \\ &+ \frac{1}{24} \sum_{j,k,l,r=1}^{d} D^4 f(\mu)_{jklr}(x-\mu)_j(x-\mu)_k(x-\mu)_l(x-\mu)_r \\ &+ \frac{1}{120} \sum_{j,k,l,r,s=1}^{d} D^5 f(\mu^*)_{jklrs}(x-\mu)_j(x-\mu)_k(x-\mu)_l(x-\mu)_r(x-\mu)_s \end{aligned}$$

where $\mu^* \in B_{\|x-\mu\|}(\mu)$ (i.e. $\|\mu^* - \mu\| < \|x-\mu\|$), and $D^k f(x)_{j_1 \cdots j_k} \equiv \dfrac{\partial^k f}{\partial x_{j_k} \cdots \partial x_{j_1}}(x)$. Thus

$$
\begin{aligned}
\mathbb{E}[f(x)] &= \int_{\mathbb{R}^d} f(x)p(x)\,dx \\
&= f(\mu)\int_{\mathbb{R}^d} p(x)\,dx + \nabla f(\mu)\int_{\mathbb{R}^d}(x-\mu)p(x)\,dx \\
&\quad + \frac{1}{2}\sum_{j,k=1}^{d} Hf(\mu)_{jk}\int_{\mathbb{R}^d}(x-\mu)_j(x-\mu)_k p(x)\,dx \\
&\quad + \frac{1}{6}\sum_{j,k,l=1}^{d} D^3 f(\mu)_{jkl}\int_{\mathbb{R}^d}(x-\mu)_j(x-\mu)_k(x-\mu)_l p(x)\,dx \\
&\quad + \frac{1}{24}\sum_{j,k,l,r=1}^{d} D^4 f(\mu)_{jklr}\int_{\mathbb{R}^d}(x-\mu)_j(x-\mu)_k(x-\mu)_l(x-\mu)_r p(x)\,dx \\
&\quad + \frac{1}{120}\sum_{j,k,l,r,s=1}^{d}\int_{\mathbb{R}^d} D^5 f(\mu_x^*)_{jklrs}(x-\mu)_j(x-\mu)_k(x-\mu)_l(x-\mu)_r(x-\mu)_s p(x)\,dx \\
&= f(\mu) + \frac{1}{2}\sum_{j,k=1}^{d} Hf(\mu)_{jk}C_{jk} + \frac{1}{6}\sum_{j,k,l=1}^{d} D^3 f(\mu)_{jkl}S_{jkl} + \frac{1}{24}\sum_{j,k,l,r=1}^{d} D^4 f(\mu)_{jklr}K_{jklr} \\
&\quad + \frac{1}{120}\sum_{j,k,l,r,s=1}^{d}\int_{\mathbb{R}^d} D^5 f(\mu_x^*)_{jklrs}(x-\mu)_j(x-\mu)_k(x-\mu)_l(x-\mu)_r(x-\mu)_s p(x)\,dx
\end{aligned}
$$

where the subscript on μ_x^* denotes the implicit dependence on x of the remainder in Taylor's theorem. Since the quadrature exactly matches the first four moments we have,

$$
1 = \sum_{i=1}^{m} w_i, \qquad \mu = \sum_{i=1}^{m} w_i\sigma_i, \qquad C_{jk} = \sum_{i=1}^{m} w_i(\sigma_i-\mu)_j(\sigma_i-\mu)_k,
$$

$$
S_{jkl} = \sum_{i=1}^{m} w_i(\sigma_i-\mu)_j(\sigma_i-\mu)_k(\sigma_i-\mu)_l, \quad \text{and} \quad K_{jklr} = \sum_{i=1}^{m} w_i(\sigma_i-\mu)_j(\sigma_i-\mu)_k(\sigma_i-\mu)_l(\sigma_i-\mu)_r.
$$

So applying Taylor's theorem inside the quadrature formula yields,

$$
\begin{aligned}
\sum_{i=1}^{m} w_i f(\sigma_i) &= \sum_{i=1}^{m} w_i \Big(f(\mu) + \nabla f(\mu)(\sigma_i - \mu) + \frac{1}{2} \sum_{j,k=1}^{d} Hf(\mu)_{jk} (\sigma_i - \mu)_j (\sigma_i - \mu)_k \\
&\quad + \frac{1}{6} \sum_{j,k,l=1}^{d} D^3 f(\mu)_{jkl} (\sigma_i - \mu)_j (\sigma_i - \mu)_k (\sigma_i - \mu)_l \\
&\quad + \frac{1}{24} \sum_{j,k,l,r=1}^{d} D^4 f(\mu)_{jklr} (\sigma_i - \mu)_j (\sigma_i - \mu)_k (\sigma_i - \mu)_l (\sigma_i - \mu)_r \\
&\quad + \frac{1}{120} \sum_{j,k,l,r,s=1}^{d} D^5 f(\mu^*)_{jklrs} (x - \mu)_j (x - \mu)_k (x - \mu)_l (x - \mu)_r (x - \mu)_s \Big) \\
&= f(\mu) + \frac{1}{2} \sum_{j,k=1}^{d} Hf(\mu)_{jk} C_{jk} + \frac{1}{6} \sum_{j,k,l=1}^{d} D^3 f(\mu)_{jkl} S_{jkl} + \frac{1}{24} \sum_{j,k,l,r=1}^{d} D^4 f(\mu)_{jklr} K_{jklr} \\
&\quad + \frac{1}{120} \sum_{j,k,l,r,s=1}^{d} \left(\sum_{i=1}^{m} D^5 f(\mu^*_{\sigma_i})_{jklrs} w_i (\sigma_i - \mu)_j (\sigma_i - \mu)_k (\sigma_i - \mu)_l (\sigma_i - \mu)_r (\sigma_i - \mu)_s \right)
\end{aligned}
$$

Notice that the first four terms of the true expectation and the quadrature formula agree. Since the first four terms cancel, the error becomes,

$$
\begin{aligned}
&\left| \mathbb{E}[f(x)] - \sum_{i=1}^{m} w_i f(\sigma_i) \right| \\
&= \frac{1}{120} \left| \sum_{j,k,l,r,s=1}^{d} \int_{\mathbb{R}^d} D^5 f(\mu^*_x)_{jklrs} (x - \mu)^{\otimes 5}_{jklrs} \, dp - \sum_{i=1}^{m} D^5 f(\mu^*_{\sigma_i})_{jklrs} w_i (\sigma_i - \mu)^{\otimes 5}_{jklrs} \right| \\
&\leq \frac{1}{120} \sum_{j,k,l,r,s=1}^{d} \int_{\mathbb{R}^d} \left| D^5 f(\mu^*_x)_{jklrs} (x - \mu)^{\otimes 5}_{jklrs} \right| \, dp + \sum_{i=1}^{m} \left| D^5 f(\mu^*_{\sigma_i})_{jklrs} w_i (\sigma_i - \mu)^{\otimes 5}_{jklrs} \right| \\
&\leq \frac{||D^5 f||_\infty}{120} \sum_{j,k,l,r,s=1}^{d} \int_{\mathbb{R}^d} \left| (x - \mu)^{\otimes 5}_{jklrs} \right| \, dp + \sum_{i=1}^{m} \left| w_i (\sigma_i - \mu)^{\otimes 5}_{jklrs} \right| \\
&\leq ||D^5 f||_\infty \frac{d^5}{120} \left(||M_{5,abs}||_{\max} + ||\tilde{M}_{5,abs}||_{\max} \right)
\end{aligned}
$$

□

While the assumption of compact support is not strictly necessary, one must make some assumption on the decay of the probability measure in order to control the error. Moreover,

the Taylor's theorem approach does not allow less regular functions f or take advantage of additional regularity that may be present in f. Thus we take a more general approach based on the methods of polynomial approximation [36, 37].

The benefit of our ability to match four moments to arbitrary precision is that it allows us to apply the standard approach for quadrature error analysis based on polynomial approximation. In this section, we develop error bounds in the context of a quadrature that matches n moments. Since the HOUT matches four moments, the bounds developed in this section apply to the HOUT with $n = 4$. Of course, this immediately requires an assumption on the probability measure dp that the first n-moments exist. However, we will not require the existence of a density or any regularity assumptions on the measure.

The polynomials $1, x, (x-\mu)^{\otimes 2}, ..., (x-\mu)^{\otimes n}$ form a basis for the space of degree n polynomials in the components of $x \in \mathbb{R}^d$, denoted Π_n^d. Since expectations are linear, a quadrature which is exact on these basis polynomials will be exact for all polynomials of degree less than or equal to n, namely, $\mathbb{E}[q] = \sum_{i=1}^m w_i q(\sigma_i)$ for any $q \in \Pi_n^d$. Of course, the quadrature may only be accurate up to threshold and in finite precision arithmetic it cannot be exact. Moreover, the moments that the quadrature is matching may only be estimates of the true moments. To understand the propagation of such errors, we write the polynomial $q(x) = \sum_{s=0}^n \sum_{j_1,...,j_s=1}^d a_{j_1 \cdots j_s} (x-\mu)^{\otimes s}_{j_1 \cdots j_s}$ in the basis of moments. Note that

$$\begin{aligned} E_{\text{moments}} &\equiv \left| \mathbb{E}[q] - \sum_{i=1}^m w_i q(\sigma_i) \right| \\ &= \left| \sum_{s=0}^n \sum_{j_1,...,j_s=1}^d a_{j_1 \cdots j_s} \left(\mathbb{E}[(x-\mu)^{\otimes s}_{j_1 \cdots j_s}] - \sum_{i=1}^m w_i (\sigma_i - \mu)^{\otimes s}_{j_1 \cdots j_s} \right) \right| \\ &\leq c(q) \sum_{s=0}^n ||M_s - \tilde{M}_s||_{\max} \end{aligned}$$

where $c(q)$ is a constant depending only on the polynomial q and $M_s = \mathbb{E}[(x-\mu)^{\otimes s}]$ are the true moments and $\tilde{M}_s = \sum_{i=1}^m w_i (\sigma_i - \mu)^{\otimes s}$ are the moments matched by the algorithm.

Whenever we approximate a function f by a polynomial $q \in \Pi_n^d$, we should expect unbounded errors as the inputs approach infinity. Thus, in order to control the error on $\mathbb{E}[f]$ by polynomial approximation, we need to split the domain into the interior and exterior of a ball $\mathbb{B}_r(\mu)$ of radius r centered on μ. Outside the ball we define the error by

$$E_{\text{outside}} \equiv \int_{\mathbb{R}^d \cap \mathbb{B}_r(\mu)^c} |f - q| \, dp.$$

and bounding this error requires assuming that the probability measure decays sufficiently fast to control the error between f and q. Inside the ball we define the polynomial approximation error by

$$E_{\text{inside}} \equiv ||f - q||_\infty = \sup_{x \in \mathbb{B}_r(\mu)} |f(x) - q(x)|$$

and bounding this error will require an appropriate regularity assumption on f.

By combining these error terms, we can control the error of a quadrature formula on any function f by any polynomial q of degree n, namely,

$$\begin{aligned}
E_{\text{total}} &\equiv \left| \mathbb{E}[f] - \sum_{i=1}^m w_i f(\sigma_i) \right| \\
&\leq |\mathbb{E}[f] - \mathbb{E}[q]| + E_{\text{moments}} + \left| \sum_{i=1}^m w_i q(\sigma_i) - \sum_{i=1}^m w_i f(\sigma_i) \right| \\
&\leq E_{\text{moments}} + \int_{\mathbb{R}^d} |f - q| \, dp + \sum_{i=1}^m w_i |f(\sigma_i) - q(\sigma_i)| \\
&\leq E_{\text{moments}} + \int_{\mathbb{R}^d \cap \mathbb{B}_r(\mu)^c} |f - q| \, dp + \int_{\mathbb{B}_r(\mu)} ||f - q||_\infty \, dp + \sum_{i=1}^m w_i ||f - q||_\infty \\
&\leq E_{\text{moments}} + E_{\text{outside}} + 2E_{\text{inside}}
\end{aligned}$$

where we assume that r is sufficiently large that $\sigma_i \in \mathbb{B}_r(\mu)$ for all $i = 1, ..., m$. Notice that

the three error terms all depend on the choice of the polynomial q, and since the inequality holds for all $q \in \Pi_n^d$ we can write

$$\left| \mathbb{E}[f] - \sum_{i=1}^{m} w_i f(\sigma_i) \right| \leq \inf_{q \in \Pi_n^d} \left\{ E_{\text{moments}} + E_{\text{outside}} + 2E_{\text{inside}} \right\}.$$

From this general framework, many potential results can be derived depending on the localization of the probability measure and the regularity of f. If we assume that the moments are exactly approximated, then one such result would be the following theorem.

Theorem 6 (General HOUT Error Bound)**.** *Let $f \in C^n(\mathbb{R}^d, \mathbb{R})$ be bounded in absolute value by a polynomial, $|f(x)| \leq a + b||x||^t$. Let x be a random variable with probability density $p(x) < ce^{-\alpha||x-\mu||^\beta}$ for some $\alpha, \beta > 0$ and all $||x - \mu|| > r_0$. Let $Q(f) \equiv \sum_{i=1}^m w_i f(\sigma_i)$ be exact on the first k moments of p. For any radius $r \geq r_0$ such that $\sigma_i \in \mathbb{B}_r(\mu)$ we have*

$$|\mathbb{E}[f] - Q(f)| \leq c_1 \left(\frac{r}{k}\right)^n \left(\frac{||D^n f||_\infty}{k} + \sum_{|\gamma|=n} \sup_{|x-y|<\frac{1}{k}} |D_\gamma^n f(x) - D_\gamma^n f(y)| \right) + c_2\, k\, r^{t+k+d-\beta} e^{-\alpha r^\beta},$$

where c_1 depends on n, d and c_2 depends on a, b, α, β.

Proof. Recall that the total quadrature error is bounded above by the sum of the error due to the moments, E_{moments}, the error inside the ball, E_{inside}, and the error outside, E_{outside}. Since we assume that the quadrature exactly matches the first k moments, we have $E_{\text{moments}} = 0$. Next, combining the bound on f and the exponential decay bound on

the density we have,

$$
\begin{aligned}
E_{\text{outside}} &= \int_{\mathbb{B}_r(\mu)^c} |f-q|\,dp \le \int_{\mathbb{B}_r(\mu)^c} (a+b||x-\mu||^t + b_2||x-\mu||^k) c e^{-\alpha||x-\mu||^\beta}\,dx \\
&= \omega_d \int_r^\infty (as^{d-1} + bs^{t+d-1} + b_2 s^{k+d-1}) c e^{-\alpha s^\beta}\,ds \\
&\le c\omega_d \int_r^\infty a s^{d-\beta} s^{\beta-1} e^{-\alpha s^\beta} + b s^{t+d-\beta} s^{\beta-1} e^{-\alpha s^\beta} + b_2 s^{k+d-\beta} s^{\beta-1} e^{-\alpha s^\beta}\,ds \\
&= (c_3 r^{d-\beta} + c_4 r^{t+d-\beta} + c_5 r^{k+d-\beta}) e^{-\alpha r^\beta} \\
&\quad + c\omega_d \int_r^\infty a_1 s^{d-\beta-1} e^{-\alpha s^\beta} + b_3 s^{t+d-\beta-1} e^{-\alpha s^\beta} + b_4 s^{k+d-\beta-1} e^{-\alpha s^\beta}\,ds.
\end{aligned}
$$

The above integration by parts can be repeated until $d-\beta, t+d-\beta, k+d-\beta$ are all less than $\beta-1$, then the integrands are bounded above by $s^{\beta-1}e^{-\alpha s^\beta}$ which is integrable exactly. These integration by parts pick up polynomial terms multiplied by $e^{-\alpha r^\beta}$ all of which are bounded by $r^{t+k+d-\beta}e^{-\alpha r^\beta}$. Since there are fewer than k such terms, we have

$$
E_{\text{outside}} \le c_2\, k\, r^{t+k+d-\beta} e^{-\alpha r^\beta}.
$$

Finally, we turn to the error of polynomial approximation inside $\mathbb{B}_r(\mu)$. Defining $\tilde{f}(x) = f(rx+\mu)$ on the unit ball, we can apply Theorem 3.4 of [37] which says there exists a polynomial $\tilde{q}$ such that,

$$
||\tilde{f} - q||_{\infty,\mathbb{B}_1(0)} \le \frac{c_1}{k^n} \left(\frac{||D^n \tilde{f}||_\infty}{k} + \sum_{|\gamma|=n} \sup_{|x-y|<1/k} |D^n_\gamma \tilde{f}(x) - D^n_\gamma \tilde{f}(y)| \right).
$$

By the chain rule we have $|D^n \tilde{f}| = r^n |D^n f|$ so that

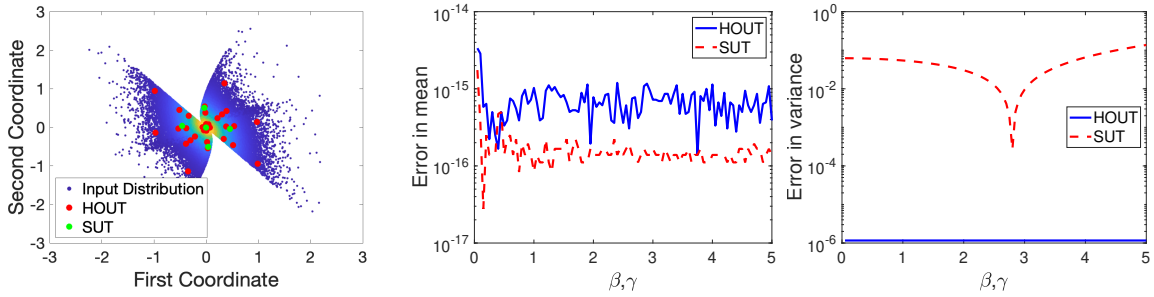
$$||f - q||_\infty \leq c_1 \left(\frac{r}{k}\right)^n \left(\frac{||D^n f||_\infty}{k} + \sum_{|\gamma|=n} \sup_{|x-y|<1/k} |D^n_\gamma f(x) - D^n_\gamma f(y)| \right)$$

where $q(x) = \tilde{q}((x-\mu)/r)$. □

The proof of Theorem 6 follows from upper bounds on the error of the multivariate polynomial of best approximation found in [36,37] together with bounds on the integrals of polynomials multiplied by an exponential.

Of course, the HOUT currently has only been derived for $k = 4$, however we chose to derive the general error bounds to show how matching more moments can potentially improve the estimation in the future.

4.3 Numerical Experiments



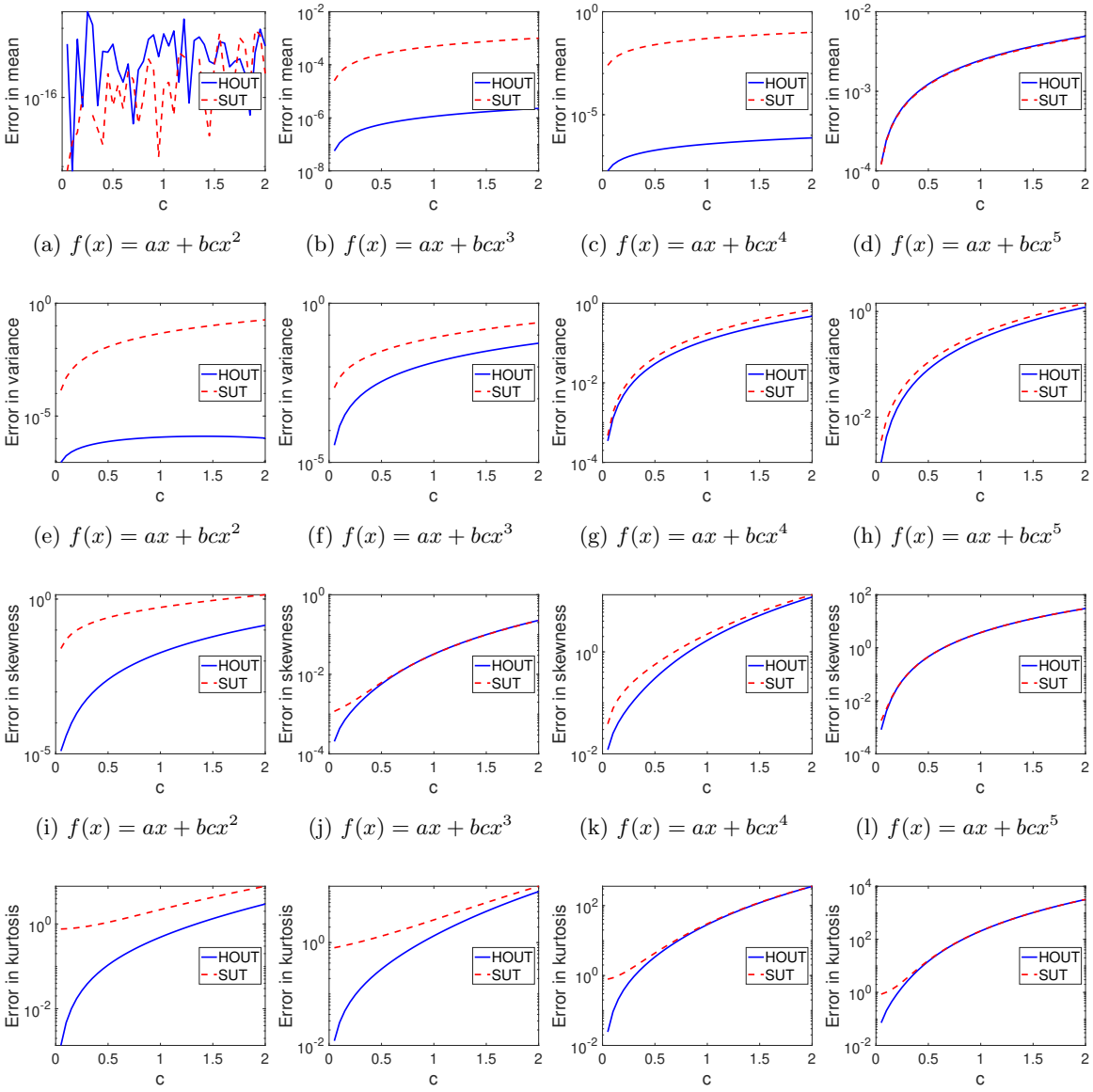
(a) (b) (c)

Figure 4.1: (a) Comparison between the higher order unscented transform ensemble (HOUT, red dots) and the Scaled Unscented Transform ensemble (SUT, green dots) on a non-Gaussian distribution. Note that the SUT uses 5 σ-points while the HOUT uses 69 σ-points. (b,c) Estimating the output mean and covariance for various values of β in the SUT and various values of γ in the HOUT.

We first compare the HOUT and SUT on various polynomials applied to a two dimensional input distribution. In order to generate a non-Gaussian input distribution, we start by

generating an ensemble of 10^5 standard Gaussian random variables, $Z \in \mathbb{R}^2$ and then transforming them by a map $X = AZ + B(Z \odot Z \odot \text{sign}(Z))$ where A, B are random 2×2 matrices with entries chosen from a Gaussian distribution with mean 0 and standard deviation $1/10$ and $\odot$ is componentwise multiplication. The resulting ensemble is shown in Fig. 4.1(a) along with the HOUT (red dots) and SUT (green dots) ensembles.

The SUT has the free parameter β but the HOUT requires a certain inequality for β and instead the HOUT has γ as a free parameter. In order to explore the effect of these parameters on the SUT and HOUT, we considered a random quadratic polynomial $f : \mathbb{R}^2 \to \mathbb{R}$. In Fig. 4.1 we show the error of the HOUT and SUT estimates of the mean $\mathbb{E}[f(X)]$ and variance $\mathbb{E}[(f(X) - \mathbb{E}[f(X)])^2]$ as a function of β for the SUT and γ for the HOUT. Notice that since f is a quadratic polynomial, the mean is also a quadratic polynomial, whereas the variance is a quartic polynomial. Since the SUT has degree of exactness two, it is exact on the mean but not on the variance. The HOUT has degree of exactness four and is exact on both up to the specified tolerance (10^{-5} in these experiments). Reducing the tolerance below this point led to increased error, most likely due to the conditioning of the HOUT quadrature rule.



(a) $f(x) = ax + bcx^2$ (b) $f(x) = ax + bcx^3$ (c) $f(x) = ax + bcx^4$ (d) $f(x) = ax + bcx^5$

(e) $f(x) = ax + bcx^2$ (f) $f(x) = ax + bcx^3$ (g) $f(x) = ax + bcx^4$ (h) $f(x) = ax + bcx^5$

(i) $f(x) = ax + bcx^2$ (j) $f(x) = ax + bcx^3$ (k) $f(x) = ax + bcx^4$ (l) $f(x) = ax + bcx^5$

(m) $f(x) = ax + bcx^2$ (n) $f(x) = ax + bcx^3$ (o) $f(x) = ax + bcx^4$ (p) $f(x) = ax + bcx^5$

Figure 4.2: Comparison between the higher order unscented transform (HOUT) and the Scaled Unscented Transform (SUT) when estimating the mean (top row), variance (second row), skewness (third row), and kurtosis (bottom row) with different polynomials. Notice that the SUT has degree of exactness two while the HOUT has degree of exactness four.

Using the same two-dimensional distribution, X, we passed it through several polynomial functions of the form $f(x) = ax + bcx^n$ for $n = 2, 3, 4, 5$ where a and b are made random 1×2 vectors. To show the influence of the strength of the nonlinearity, we sweep

through different values of c. In Fig. 4.2 we compare the HOUT and SUT for estimating the mean and variance of the output of each of these polynomials. As expected, the HOUT is exact for the means up to $n = 4$ and for the variances up to $n = 2$ due to having degree of exactness four. For higher degree polynomials, the HOUT has comparable or better performance. Whenever the nonlinearity is not too strong, such as when c is small and/or the power n is small, the HOUT has a big advantage. However, for some strong nonlinearities when c and the power n is large then the HOUT and SUT may have similar performance.

Of course, the HOUT and SUT are intended for use beyond polynomial functions. In fact, the most common application is for forecasting dynamical systems. Next, we consider the problem of forecasting the chaotic Lorenz-63 dynamical system [38]. We integrate the Lorenz-63 system with a Runge-Kutta order four method and a time step $\tau = 0.1$. In order to generate a non-Gaussian initial state, we start by choosing a random point on the attractor and adding a small amount of Gaussian noise. We then run the ensemble forward $N_1 = 5$ steps and we consider this the initial state, see Fig. 4.3(a) (blue) and Fig. 4.3(b) (blue). We compute the statistics of the initial state using the ensemble shown, and use these statistics to generate the HOUT and SUT as shown in Fig. 4.3(b). All three ensembles are then integrated forward in time N_2 additional steps and the true forecast statistics from the large ensemble are compared to the HOUT and SUT estimates. An example is shown in Fig. 4.3(c) with $N_2 = 15$.

We then repeat this experiment 500 times with different randomly selected initial states on the attractor and we compute the geometric average of the error between the HOUT estimate and the true statistics at each forecast time, shown in Fig. 4.3(d-g)(blue). Similarly, we compute the geometric average of the error between the SUT estimate and the true statistics (red) at each forecast time, shown in Fig. 4.3(d-g)(red). We note that the HOUT provides improved estimates of the first four moments up to at least 4 forecast steps, which is 0.4 model time units. In particular, the mean forecast is improved by an order of magnitude in this forecast range.

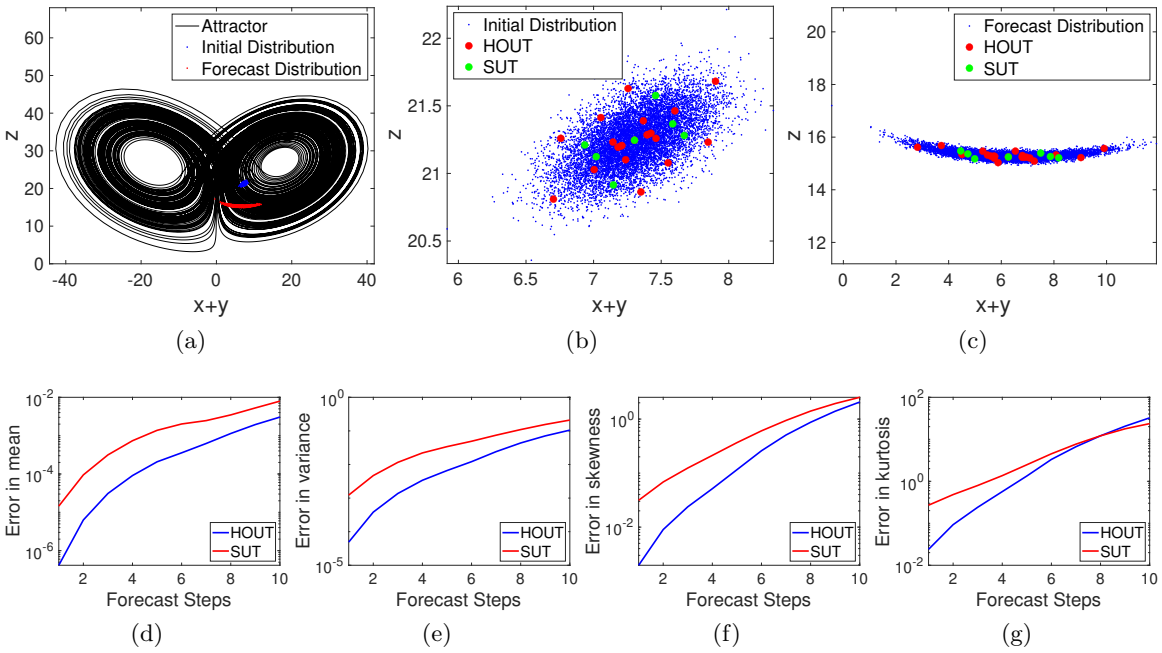


Figure 4.3: Comparison between the higher order unscented transform (HOUT) and the Scaled Unscented Transform (SUT) when estimating the mean $\mathbb{E}[f(X)]$ (top row) and higher moments of the Lorenz-63 model at various forecast horizons. In (a) we show the Lorenz-63 attractor (black) along with an example initial ensemble (blue) and forecast ensemble (red) used to compute the true statistics. In (b,c) we show the initial and forecast ensembles (blue) together with the HOUT (red) and SUT (green) ensembles. Results in (d-g) show the forecast accuracy versus the forecast steps and are geometrically averaged over 500 different initial conditions on the attractor.

Chapter 5: A Higher Order Kalman Filter

We just went over the completed forecast step of the proposed Higher Order Kalman Filter. Now, let us explore how we can get the assimilation step. How we do this is by observing the various ways to derive the Kalman Filter. There is the Bayesian approach, the Minimum Mean-Square Estimate (MMSE) approach and the Closure approach. Since each of these approaches leads to the same Kalman filter, each one provides a different avenue for potentially generalizing the Kalman Filter. We'll now discuss each of these briefly in turn and how they might be generalized to higher order Kalman Filtering.

5.1 Bayesian approach

5.1.1 Generalizing the Bayesian Approach

To see about generalizing this approach, instead of having Gaussian noise, we will allow a much more general class of noise. Thus instead of using likelihood and prior that are exponentials with a quadratic function as an exponent, we will use exponentials with a quartic function as an exponent. Not only are we generalizing this approach by using non Gaussian statistics that have non trivial skewness and kurtosis, we are also allowing the noise to be more complicated at the same time. One nice feature of this generalization is that we will also be using a maximum entropy distribution. The idea is if you specify the first k moments of a distribution, then there are a lot of distributions that have those k moments. But if you try to find out of all those possible distributions, the one that has maximum entropy, then there is a unique answer and it turns out to be e to a kth degree polynomial. We speculate that one of the reasons the Kalman filter works so well is that the Kalman update is based on a Gaussian assumption which is also a maximum entropy assumption. So we should note that doing this e to a quartic polynomial idea it may be

a very natural generalization because it is also using maximum entropy distribution for 4 moments. One of our goals would be to connect the performance of the filter with this maximum entropy distribution. Is there some reason that choosing a maximum entropy distribution is a really nice choice? We wish to investigate that.

Notice even in the one-dimensional case, this is a difficult problem to solve. As motivation, let us briefly explore the one-dimensional derivation of the original Kalman filter. In the one-dimensional case, the prior and likelihood are defined as

$$p(x) = e^{-\frac{(x-\mu^-)^2}{2(\sigma^2)^-}} \quad \text{and} \quad p(y|x) = e^{-\frac{(y-Hx)^2}{2r^2}},$$

respectively. Let

$$\begin{aligned} q(x) &= -\frac{(x-\mu^-)^2}{2(\sigma^2)^-} \\ r(x) &= -\frac{(y-Hx)^2}{2r^2} \end{aligned}$$

Then expanding the above equations give us

$$\begin{aligned} q(x) &= -\frac{(\mu^-)^2}{2(\sigma^2)^-} + \frac{\mu^-}{(\sigma^2)^-}x - \frac{1}{2(\sigma^2)^-}x^2 = a_0 + a_1 x + a_2 x^2 \\ r(x) &= -\frac{y^2}{2r^2} + \frac{Hy}{r^2}x - \frac{H^2}{2r^2}x^2 = b_0 + b_1 x + b_2 x^2 \end{aligned}$$

Thus by Bayes' Law

$$p(x|y) \propto e^{q(x)}e^{r(x)} = e^{q(x)+r(x)}$$

where grouping like terms we have

$$q(x) + r(x) = -\frac{(\mu^-)^2 r^2 + (\sigma^2)^- y^2}{2r^2(\sigma^2)^-} + \left(\frac{(\mu^-)^2}{(\sigma^2)^-} + \frac{Hy}{r^2}\right) x - \frac{1}{2}\left(\frac{1}{2(\sigma^2)^-} + \frac{H^2}{2r^2}\right) x^2.$$

Since the posterior is Gaussian, it has the following form

$$p(x|y) \propto e^{-\frac{(x-\mu^+)^2}{2(\sigma^2)^+}} = e^{\hat{p}(x)}$$

So after expanding we have

$$\hat{p}(x) = -\frac{(\mu^+)^2}{2(\sigma^2)^+} + \frac{\mu^+}{(\sigma^2)^+} x - \frac{1}{2(\sigma^2)^+} x^2 = c_0 + c_1 x + c_2 x^2$$

Then once we set $\hat{p}(x) = q(x) + r(x)$, or more specifically setting $c_1 = a_1 + b_1$ and $c_2 = a_2 + b_2$, we find that posterior moments are defined as follows

$$\begin{aligned} \mu^+ &= (\sigma^2)^+ \left(\frac{(\mu^-)^2}{(\sigma^2)^-} + \frac{Hy}{r^2}\right) \\ (\sigma^2)^+ &= \left(\frac{1}{2(\sigma^2)^-} + \frac{H^2}{2r^2}\right)^{-1} \end{aligned}$$

From this we can see the general idea for how we can write the posterior moments in terms of the prior moments is by setting the coefficients of $q(x) + r(x)$ equal to the coefficients of $\hat{p}(x)$. Now, the nice thing about the gaussian is that it is very easy to read off the mean and covariance from the polynomial that e is raised to. That may be harder when we deal with say quartic polynomials as opposed to quadratic polynomials.

The general approach we would like to take is using Bayes' law so we get

$$p(x|y) \propto e^{q(x)} e^{r(x)}$$

where $q(x)$ and $r(x)$ are two quartic polynomials and we wish to write it as

$$p(x|y) \propto e^{\hat{p}(x)}$$

where $\hat{p}(x)$ is a quartic polynomial. The real challenge here comes with figuring out the connection between the moments and the coefficients to that polynomial. We are going to follow John Harlim's approach in [39].

5.1.2 A Solution to the Moment Problem

Let $q(x)$ be a k-degree polynomial defined as such

$$q(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_k x^k,$$

where the distribution is $p(x) = e^{q(x)}$ and the raw moments (moments about the origin) are defined as

$$m_j = \mathbb{E}[x^j] = \int_{-\infty}^{\infty} x^j p(x)\ dx$$

where a_0 is chosen so that $m_0 = 1$.

In the terms of the first four moments of a distribution that we have been discussing, i.e. the mean, variance, skewness and kurtosis, where

$$\begin{aligned}
\mu &= \mathbb{E}[X] \\
\sigma^2 &= \mathbb{E}[(X-\mu)^2] = \mathbb{E}[X^2] - \mu^2 \\
S &= \mathbb{E}[(X-\mu)^3] = \mathbb{E}[X^3] - 3\mu\mathbb{E}[X^2] - 2\mu^3 \\
\kappa &= \mathbb{E}[(X-\mu)^4] = \mathbb{E}[X^4] - 4\mu\mathbb{E}[X^3] + 6\mu^2\mathbb{E}[X^2] - 3\mu^4,
\end{aligned}$$

(in this case the second, third and fourth moments would be considered central moments),

they can be represented in the following way in terms of m_j,

$$\begin{aligned}
\mu &= m_1 \\
\sigma^2 &= m_2 - m_1^2 \\
S &= m_3 - 3m_1m_2 - 2m_1^3 \\
\kappa &= m_4 - 4m_1m_3 + 6m_1^2m_2 - 3m_1^4.
\end{aligned}$$

We wish to find $\mathbb{E}[x^j q'(x)]$ in terms of the moments for $j = 0, 1, 2, ..., k-1$. Plugging in the derivative

$$q'(x) = a_1 + 2a_2x + 3a_3x^2 + \cdots + ka_kx^{k-1},$$

we have

$$\begin{aligned}
\mathbb{E}[x^j q'(x)] &= \mathbb{E}[x^j(a_1 + 2a_2x + 3a_3x^2 + \cdots + ka_kx^{k-1})] \\
&= \mathbb{E}[a_1x^j + 2a_2x^{j+1} + 3a_3x^{j+2} + \cdots + ka_kx^{j+k-1}]
\end{aligned}$$

By linearity,

$$\mathbb{E}[x^j q'(x)] = a_1\mathbb{E}[x^j] + 2a_2\mathbb{E}[x^{j+1}] + 3a_3\mathbb{E}[x^{j+2}] + \cdots + ka_k\mathbb{E}[x^{j+k-1}]$$

Thus in terms of the moments,

$$\mathbb{E}[x^j q'(x)] = a_1m_j + 2a_2m_{j+1} + 3a_3m_{j+2} + \cdots + ka_km_{j+k-1} \tag{5.1}$$

Now we use integration by parts to find another expression for $\mathbb{E}[x^j q'(x)]$. By the definition of expectation,

$$\mathbb{E}[x^j q'(x)] = \int_{-\infty}^{\infty} x^j q'(x)p(x)\ dx = \int_{-\infty}^{\infty} x^j q'(x)e^{q(x)}\ dx$$

Let $u = x^j$ and $dv = q'(x)e^{q(x)}\ dx$. Then $du = jx^{j-1}\ dx$ and $v = e^{q(x)}$. Thus by integration by parts,

$$\begin{aligned}
\mathbb{E}[x^j q'(x)] &= x^j e^{q(x)}\Big|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} jx^{j-1}e^{q(x)}\ dx \\
&= 0 - j\int_{-\infty}^{\infty} x^{j-1}e^{q(x)}\ dx \\
&= -j\int_{-\infty}^{\infty} x^{j-1}p(x)\ dx \\
&= -j\mathbb{E}[x^{j-1}]
\end{aligned}$$

Hence we have

$$\mathbb{E}[x^j q'(x)] = -jm_{j-1}. \tag{5.2}$$

From (5.1) and (5.2), we have

$$a_1 m_j + 2a_2 m_{j+1} + 3a_3 m_{j+2} + \cdots + ka_k m_{j+k-1} = -jm_{j-1} \tag{5.3}$$

So, for $j = 0, 1, 2, ..., k-1$, we get the following system of equations

$$\begin{aligned}
a_1 m_0 + 2a_2 m_1 + 3a_3 m_2 + \cdots + ka_k m_{k-1} &= 0 \\
a_1 m_1 + 2a_2 m_2 + 3a_3 m_3 + \cdots + ka_k m_k &= -m_0 \\
a_1 m_2 + 2a_2 m_3 + 3a_3 m_4 + \cdots + ka_k m_{k+1} &= -2m_1 \\
a_1 m_3 + 2a_2 m_4 + 3a_3 m_5 + \cdots + ka_k m_{k+2} &= -3m_2 \\
&\vdots \\
a_1 m_{k-1} + 2a_2 m_k + 3a_3 m_{k+1} + \cdots + ka_k m_{2k-2} &= -(k-1)m_{k-2}
\end{aligned}$$

which can be represented as the following matrix equation

$$\begin{pmatrix} m_0 & 2m_1 & 3m_2 & \cdots & km_{k-1} \\ m_1 & 2m_2 & 3m_3 & \cdots & km_k \\ m_2 & 2m_3 & 3m_4 & \cdots & km_{k+1} \\ m_3 & 2m_4 & 3m_5 & \cdots & km_{k+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ m_{k-1} & 2m_k & 3m_{k+1} & \cdots & km_{2k-2} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} 0 \\ -m_0 \\ -2m_1 \\ -3m_2 \\ \vdots \\ -(k-1)m_{k-2} \end{pmatrix}$$

Hence

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} m_0 & 2m_1 & 3m_2 & \cdots & km_{k-1} \\ m_1 & 2m_2 & 3m_3 & \cdots & km_k \\ m_2 & 2m_3 & 3m_4 & \cdots & km_{k+1} \\ m_3 & 2m_4 & 3m_5 & \cdots & km_{k+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ m_{k-1} & 2m_k & 3m_{k+1} & \cdots & km_{2k-2} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ -m_0 \\ -2m_1 \\ -3m_2 \\ \vdots \\ -(k-1)m_{k-2} \end{pmatrix}$$

Notice the difficulty here is that we need $k-2$ additional moments in order to solve for the k coefficients. So we are going to explore the different ways of closing these equations either by making some ansatz for m_{k+1}, m_{k+2}, $\ldots$, m_{2k-2} or maybe looking for some additional equations to add. This challenge is only going to get harder in higher dimensions with tensors.

5.1.3 Three-Moment Filter

Let us first explore the simpler case in which we try the Bayesian approach with the following cubic polynomial

$$q(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3,$$

then by (5.3) we have

$$a_1 m_j + 2a_2 m_{j+1} + 3a_3 m_{j+2} = -j m_{j-1}.$$

Then for $j = 0, 1, 2$, we have the following system of equations

$$\begin{aligned} a_1 m_0 + 2a_2 m_1 + 3a_3 m_2 &= 0 \\ a_1 m_1 + 2a_2 m_2 + 3a_3 m_3 &= -m_0 \\ a_1 m_2 + 2a_2 m_3 + 3a_3 m_4 &= -2m_1 \end{aligned} \tag{5.4}$$

which can be represented as the following matrix equation

$$\begin{pmatrix} m_0 & 2m_1 & 3m_2 \\ m_1 & 2m_2 & 3m_3 \\ m_2 & 2m_3 & 3m_4 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} 0 \\ -m_0 \\ -2m_1 \end{pmatrix}$$

Hence

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} m_0 & 2m_1 & 3m_2 \\ m_1 & 2m_2 & 3m_3 \\ m_2 & 2m_3 & 3m_4 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ -m_0 \\ -2m_1 \end{pmatrix}$$

Solving the above equation gives us

$$\begin{aligned} a_1 &= \frac{2m_3 m_1^2 - 2m_1 m_2^2 - m_4 m_1 + m_3 m_2}{m_4 m_1^2 - 2m_1 m_2 m_3 + m_2^3 - m_4 m_2 + m_3^2} \\ a_2 &= \frac{2m_1^2 m_2 - 2m_3 m_1 - m_2^2 + m_4}{2(m_4 m_1^2 - 2m_1 m_2 m_3 + m_2^3 - m_4 m_2 + m_3^2)} \\ a_3 &= \frac{-2m_1^3 + 3m_2 m_1 - m_3}{3(m_4 m_1^2 - 2m_1 m_2 m_3 + m_2^3 - m_4 m_2 + m_3^2)} \end{aligned}$$

Thus in terms of μ, σ^2, S, κ we have

$$\begin{aligned} a_1 &= -\frac{4\mu^5 + 4\mu^3\sigma^2 + S\mu^2 - \mu(\sigma^2)^2 + \kappa\mu - S\sigma^2}{S^2 + 8S\mu^3 + 16\mu^6 + (\sigma^2)^3 - \kappa\sigma^2} \\ a_2 &= \frac{8\mu^4 + 2S\mu - (\sigma^2)^2 + \kappa}{2(S^2 + 8S\mu^3 + 16\mu^6 + (\sigma^2)^3 - \kappa\sigma^2)} \\ a_3 &= -\frac{4\mu^3 + S}{3(S^2 + 8S\mu^3 + 16\mu^6 + (\sigma^2)^3 - \kappa\sigma^2)} \end{aligned}$$

Now, we would like to solving the system of equations (5.4) for m_1, m_2, m_3 in terms of a_1, a_2, a_3 which can be represented as the following matrix equation

$$\begin{pmatrix} 2a_2 & 3a_3 & 0 & 0 \\ a_1 & 2a_2 & 3a_3 & 0 \\ 2 & a_1 & 2a_2 & 3a_3 \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{pmatrix} = \begin{pmatrix} a_1 \\ -1 \\ 0 \end{pmatrix}$$

Gaussian Elimination results in the following augmented matrix

$$\left(\begin{array}{cccc|c} 1 & 0 & 0 & \frac{27a_3^3}{2(4a_2^3-6a_1a_3a_2+9a_3^2)} & \frac{-(3a_3a_1^2-4a_2^2a_1-6a_2a_3)}{2(4a_2^3-6a_1a_3a_2+9a_3^2)} \\ 0 & 1 & 0 & \frac{-9a_2a_3^2}{4a_2^3-6a_1a_3a_2+9a_3^2} & \frac{-(a_2a_1^2-3a_3a_1+2a_2^2)}{4a_2^3-6a_1a_3a_2+9a_3^2} \\ 0 & 0 & 1 & \frac{3(4a_2^2a_3-3a_1a_3^2)}{2(4a_2^3-6a_1a_3a_2+9a_3^2)} & \frac{a_1^3-2a_2a_1-6a_3}{2(4a_2^3-6a_1a_3a_2+9a_3^2)} \end{array}\right)$$

thus

$$
\begin{aligned}
m_1 &= \frac{-(3a_3a_1^2 - 4a_2^2a_1 - 6a_2a_3) - 27a_3^3m_4}{2(4a_2^3 - 6a_1a_3a_2 + 9a_3^2)} \\
m_2 &= \frac{-(a_2a_1^2 - 3a_3a_1 + 2a_2^2) + 9a_2a_3^2m_4}{4a_2^3 - 6a_1a_3a_2 + 9a_3^2} \\
m_3 &= \frac{a_1^3 - 2a_2a_1 - 6a_3 - 3(4a_2^2a_3 - 3a_1a_3^2)m_4}{2(4a_2^3 - 6a_1a_3a_2 + 9a_3^2)}
\end{aligned}
$$

We can see here that there are infinitely many solutions that depend on m_4. However, perhaps optimizing the variance with respect to m_4 can give us a possible closure. By taking the above solutions for m_1 and m_2, we see that the variance is

$$
\sigma^2 = \frac{-(a_2a_1^2 - 3a_3a_1 + 2a_2^2) + 9a_2a_3^2m_4}{4a_2^3 - 6a_1a_3a_2 + 9a_3^2} - \left(\frac{-(3a_3a_1^2 - 4a_2^2a_1 - 6a_2a_3) - 27a_3^3m_4}{2(4a_2^3 - 6a_1a_3a_2 + 9a_3^2)}\right)^2.
$$

To optimize the variance, we wish to solve for

$$
\frac{d}{d\,m_4}(\sigma^2) = -\frac{9a_3^2(81a_3^4m_4 - 8a_2^4 + (9a_1^2 - 36a_2)a_3^2)}{2(-4a_2^3 + 6a_1a_3a_2 - 9a_3^2)^2} = 0
$$

which yields the following solution

$$
m_4 = \frac{8a_2^4 + 36a_3^2a_2 - 9a_1^2a_3^2}{81a_3^4}.
$$

Plugging this into equation for m_1, m_2, and m_3 we get

$$\begin{aligned} m_1 &= -\frac{a_2}{3a_3} \\ m_2 &= \frac{2a_2^2 + 3a_1a_3}{9a_3^2} \\ m_3 &= -\frac{4a_2^3 + 3a_1a_3a_2 + 9a_3^2}{27a_3^3}. \end{aligned}$$

Thus

$$\begin{aligned} \mu &= -\frac{a_2}{3a_3} \\ \sigma^2 &= \frac{a_2^2 + 3a_1a_3}{9a_3^2} \\ S &= \frac{4a_2^3 + 6a_1a_3a_2 - 9a_3^2}{27a_3^3} \\ \kappa &= \frac{a_2(25a_2^3 + 30a_1a_3a_2 + 36a_3^2)}{81a_3^4}. \end{aligned}$$

This means that with respect $p(x|y) = e^{\hat{p}(x)}$ where $\hat{p}(x) = a_0^+ + a_1^+x + a_2^+x^2 + a_3^+x^3$, the

posterior moments are

$$\begin{aligned}
\mu^+ &= -\frac{a_2^+}{3a_3^+} \\
(\sigma^2)^+ &= \frac{(a_2^+)^2 + 3a_1^+ a_3^+}{9(a_3^+)^2} \\
S^+ &= \frac{4(a_2^+)^3 + 6a_1^+ a_3^+ a_2^+ - 9(a_3^+)^2}{27(a_3^+)^3} \\
\kappa^+ &= \frac{a_2^+(25(a_2^+)^3 + 30a_1^+ a_3^+ a_2^+ + 36(a_3^+)^2)}{81(a_3^+)^4}
\end{aligned}$$

so in terms of the prior $p(x) = e^{q(x)}$ and the likelihood $p(y|x) = e^{r(x)}$ where $r(x) = b_0 + b_1 x + b_2 x^2 + b_3 x^3$,

$$\begin{aligned}
\mu^+ &= -\frac{a_2 + b_2}{3(a_3 + b_3)} \\
(\sigma^2)^+ &= \frac{(a_2 + b_2)^2 + 3(a_1 + b_1)(a_3 + b_3)}{9(a_3 + b_3)^2} \\
S^+ &= \frac{4(a_2 + b_2)^3 + 6(a_1 + b_1)(a_3 + b_3)(a_2 + b_2) - 9(a_3 + b_3)^2}{27(a_3 + b_3)^3} \\
\kappa^+ &= \frac{(a_2 + b_2)(25(a_2 + b_2)^3 + 30(a_1 + b_1)(a_3 + b_3)(a_2 + b_2) + 36(a_3 + b_3)^2)}{81(a_3 + b_3)^4}
\end{aligned}$$

where in terms of the prior moments

$$\begin{aligned}
a_1 &= -\frac{4(\mu^-)^5 + 4(\mu^-)^3(\sigma^2)^- + S^-(\mu^-)^2 - \mu^-((\sigma^2)^-)^2 + \kappa^-\mu^- - S^-(\sigma^2)^-}{(S^-)^2 + 8S^-(\mu^-)^3 + 16(\mu^-)^6 + ((\sigma^2)^-)^3 - \kappa^-(\sigma^2)^-} \\
a_2 &= \frac{8(\mu^-)^4 + 2S^-\mu^- - ((\sigma^2)^-)^2 + \kappa^-}{2((S^-)^2 + 8S^-(\mu^-)^3 + 16(\mu^-)^6 + ((\sigma^2)^-)^3 - \kappa^-(\sigma^2)^-)} \\
a_3 &= -\frac{4(\mu^-)^3 + S^-}{3((S^-)^2 + 8S^-(\mu^-)^3 + 16(\mu^-)^6 + ((\sigma^2)^-)^3 - \kappa^-(\sigma^2)^-)}.
\end{aligned}$$

5.1.4 Four-Moment Filter

Since the HOUT involves the first four moments, ideally we would like to construct a 4 moment filter. Let $q(x)$ be

$$q(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$$

where $a_4 < 0$. Then by (5.3) for $n = 4$, we have

$$a_1 m_j + 2a_2 m_{j+1} + 3a_3 m_{j+2} + 4a_4 m_{j+3} = -j m_{j-1}$$

So, for $j = 0, 1, 2, 3$, we get the following system of equations

$$\begin{aligned} a_1 m_0 + 2a_2 m_1 + 3a_3 m_2 + 4a_4 m_3 &= 0 \\ a_1 m_1 + 2a_2 m_2 + 3a_3 m_3 + 4a_4 m_4 &= -m_0 \\ a_1 m_2 + 2a_2 m_3 + 3a_3 m_4 + 4a_4 m_5 &= -2m_1 \\ a_1 m_3 + 2a_2 m_4 + 3a_3 m_5 + 4a_4 m_6 &= -3m_2 \end{aligned} \tag{5.5}$$

which can be represented as the following matrix equation

$$\begin{pmatrix} m_0 & 2m_1 & 3m_2 & 4m_3 \\ m_1 & 2m_2 & 3m_3 & 4m_4 \\ m_2 & 2m_3 & 3m_4 & 4m_5 \\ m_3 & 2m_4 & 3m_5 & 4m_6 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix} = \begin{pmatrix} 0 \\ -m_0 \\ -2m_1 \\ -3m_2 \end{pmatrix}$$

Hence

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix} = \begin{pmatrix} m_0 & 2m_1 & 3m_2 & 4m_3 \\ m_1 & 2m_2 & 3m_3 & 4m_4 \\ m_2 & 2m_3 & 3m_4 & 4m_5 \\ m_3 & 2m_4 & 3m_5 & 4m_6 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ -m_0 \\ -2m_1 \\ -3m_2 \end{pmatrix}$$

Solving the above equation gives us

$$\begin{aligned} a_1 = & \frac{\begin{array}{r} 2m_6m_1^2m_3 - 2m_1^2m_4m_5 - 2m_6m_1m_2^2 - m_1m_2m_3m_5 + 5m_1m_2m_4^2 - 2m_1m_3^2m_4 - m_6m_1m_4 \\ + m_1m_5^2 + 3m_2^3m_5 - 6m_2^2m_3m_4 + 3m_2m_3^3 + m_6m_2m_3 - m_2m_4m_5 - m_3^2m_5 + m_3m_4^2 \end{array}}{\begin{array}{r} m_6m_1^2m_4 - m_1^2m_5^2 - 2m_6m_1m_2m_3 + 2m_1m_2m_4m_5 + 2m_1m_3^2m_5 - 2m_1m_3m_4^2 + m_6m_2^3 \\ - 2m_2^2m_3m_5 - m_2^2m_4^2 + 3m_2m_3^2m_4 - m_6m_2m_4 + m_2m_5^2 - m_3^4 + m_6m_3^2 - 2m_3m_4m_5 + m_4^3 \end{array}} \\ a_2 = & \frac{\begin{array}{r} 2m_6m_1^2m_2 - 2m_1^2m_3m_5 - 3m_1m_2^2m_5 + m_1m_2m_3m_4 + 2m_1m_3^3 - 2m_6m_1m_3 + 2m_1m_4m_5 \\ + 3m_2^3m_4 - 3m_2^2m_3^2 - m_6m_2^2 + 5m_2m_3m_5 - 3m_2m_4^2 - m_3^2m_4 + m_6m_4 - m_5^2 \end{array}}{\begin{array}{r} 2(m_6m_1^2m_4 - m_1^2m_5^2 - 2m_6m_1m_2m_3 + 2m_1m_2m_4m_5 + 2m_1m_3^2m_5 - 2m_1m_3m_4^2 + m_6m_2^3 \\ - 2m_2^2m_3m_5 - m_2^2m_4^2 + 3m_2m_3^2m_4 - m_6m_2m_4 + m_2m_5^2 - m_3^4 + m_6m_3^2 - 2m_3m_4m_5 + m_4^3) \end{array}} \\ a_3 = & \frac{\begin{array}{r} -2m_6m_1^3 + 3m_5m_1^2m_2 + 4m_1^2m_3m_4 - 3m_1m_2^2m_4 - 5m_1m_2m_3^2 + 3m_6m_1m_2 \\ - m_5m_1m_3 - 2m_1m_4^2 + 3m_2^3m_3 - 3m_5m_2^2 + 2m_2m_3m_4 + m_3^3 - m_6m_3 + m_5m_4 \end{array}}{\begin{array}{r} 3(m_6m_1^2m_4 - m_1^2m_5^2 - 2m_6m_1m_2m_3 + 2m_1m_2m_4m_5 + 2m_1m_3^2m_5 - 2m_1m_3m_4^2 + m_6m_2^3 \\ - 2m_2^2m_3m_5 - m_2^2m_4^2 + 3m_2m_3^2m_4 - m_6m_2m_4 + m_2m_5^2 - m_3^4 + m_6m_3^2 - 2m_3m_4m_5 + m_4^3) \end{array}} \\ a_4 = & \frac{\begin{array}{r} 2m_5m_1^3 - 5m_1^2m_2m_4 - 2m_1^2m_3^2 + 8m_1m_2^2m_3 - 3m_5m_1m_2 \\ + 3m_1m_3m_4 - 3m_2^4 + 4m_2^2m_4 - 4m_2m_3^2 + m_5m_3 - m_4^2 \end{array}}{\begin{array}{r} 4(m_6m_1^2m_4 - m_1^2m_5^2 - 2m_6m_1m_2m_3 + 2m_1m_2m_4m_5 + 2m_1m_3^2m_5 - 2m_1m_3m_4^2 + m_6m_2^3 \\ - 2m_2^2m_3m_5 - m_2^2m_4^2 + 3m_2m_3^2m_4 - m_6m_2m_4 + m_2m_5^2 - m_3^4 + m_6m_3^2 - 2m_3m_4m_5 + m_4^3) \end{array}} \end{aligned}$$

Now, we would like to solving the system of equations (5.5) for m_1, ..., m_6 in terms of a_1, ..., a_6 which can be represented as the following matrix equation

$$\begin{pmatrix} 2a_2 & 3a_3 & 4a_4 & 0 & 0 & 0 \\ a_1 & 2a_2 & 3a_3 & 4a_4 & 0 & 0 \\ 2 & a_1 & 2a_2 & 3a_3 & 4a_4 & 0 \\ 0 & 3 & a_1 & 2a_2 & 3a_3 & 4a_4 \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \\ m_6 \end{pmatrix} = \begin{pmatrix} a_1 \\ -1 \\ 0 \\ 0 \end{pmatrix}$$

Gaussian Elimination results in the following augmented matrix

$$\left(\begin{array}{cccc|c|c}
1 & 0 & 0 & 0 & \frac{1}{\xi}(-81a_3^4+216a_2a_4a_3^2-96a_1a_4^2a_3 & -\frac{4}{\xi}(27a_4a_3^3-48a_2a_4^2a_3 & \frac{2}{\xi}(2a_4a_1^3-6a_2a_3a_1^2+4a_2^3a_1 \\
 & & & & +192a_4^3-64a_2^2a_4^2) & +16a_1a_4^3) & +9a_3^2a_1-16a_2a_4a_1 \\
 & & & & & & +6a_2^2a_3+18a_3a_4) \\
0 & 1 & 0 & 0 & \frac{2}{\xi}(27a_2a_3^3-18a_1a_4a_3^2+48a_4^2a_3 & \frac{8}{\xi}(16a_4^3-8a_2^2a_4^2 & 3a_3a_1^3-4a_2^2a_1^2-8a_4a_1^2 \\
 & & & & -48a_2^2a_4a_3+32a_1a_2a_4^2) & -6a_1a_3a_4^2+9a_2a_3^2a_4) & +18a_2a_3a_1-8a_2^3+16a_2a_4 \\
0 & 0 & 1 & 0 & \frac{1}{\xi}(32a_4a_2^3-36a_3^2a_2^2-96a_4^2a_2 & \frac{4}{\xi}(-12a_3a_4a_2^2+8a_1a_4^2a_2 & \frac{1}{\xi}(2a_2a_1^3-9a_3a_1^2-4a_2^2a_1 \\
 & & & & +27a_1a_3^3-72a_3^2a_4) & -24a_3a_4^2+9a_1a_3^2a_4) & +24a_4a_1-30a_2a_3) \\
0 & 0 & 0 & 1 & \frac{2}{\xi}(12a_3a_2^3-8a_1a_4a_2^2 & \frac{8}{\xi}(4a_4a_2^3-8a_4^2a_2 & \\
 & & & & -18a_1a_3^2a_2+12a_3a_4a_2 & -6a_1a_3a_4a_2+2a_1^2a_4^2 & \frac{1}{\xi}(-a_1^4+8a_2a_1^2-12a_3a_1 \\
 & & & & +27a_3^3-24a_1a_4^2+12a_1^2a_3a_4) & +9a_3^2a_4) & +12a_2^2-24a_4)
\end{array}\right)$$

where

$$\xi = 16a_2^4-36a_1a_3a_2^2-80a_4a_2^2+90a_3^2a_2+16a_1^2a_4a_2+9a_1^2a_3^2+96a_4^2-60a_1a_3a_4$$

thus

$$\begin{aligned}
m_1 &= \frac{\begin{array}{c}2(2a_4a_1^3-6a_2a_3a_1^2+4a_2^3a_1+9a_3^2a_1-16a_2a_4a_1+6a_2^2a_3 \\ +18a_3a_4)+(81a_3^4-216a_2a_4a_3^2+96a_1a_4^2a_3-192a_4^3 \\ +64a_2^2a_4^2)m_5+4(27a_4a_3^3-48a_2a_4^2a_3+16a_1a_4^3)m_6\end{array}}{16a_2^4-36a_1a_3a_2^2-80a_4a_2^2+90a_3^2a_2+16a_1^2a_4a_2+9a_1^2a_3^2+96a_4^2-60a_1a_3a_4} \\
m_2 &= \frac{\begin{array}{c}3a_3a_1^3-4a_2^2a_1^2-8a_4a_1^2+18a_2a_3a_1-8a_2^3+16a_2a_4 \\ -2(27a_2a_3^3-18a_1a_4a_3^2+48a_4^2a_3-48a_2^2a_4a_3 \\ +32a_1a_2a_4^2)m_5-8(16a_4^3-8a_2^2a_4^2-6a_1a_3a_4^2+9a_2a_3^2a_4)m_6\end{array}}{16a_2^4-36a_1a_3a_2^2-80a_4a_2^2+90a_3^2a_2+16a_1^2a_4a_2+9a_1^2a_3^2+96a_4^2-60a_1a_3a_4} \\
m_3 &= \frac{\begin{array}{c}2a_2a_1^3-9a_3a_1^2-4a_2^2a_1+24a_4a_1-30a_2a_3 \\ -(32a_4a_2^3-36a_3^2a_2^2-96a_4^2a_2+27a_1a_3^3-72a_3^2a_4)m_5 \\ -4(-12a_3a_4a_2^2+8a_1a_4^2a_2-24a_3a_4^2+9a_1a_3^2a_4)m_6\end{array}}{16a_2^4-36a_1a_3a_2^2-80a_4a_2^2+90a_3^2a_2+16a_1^2a_4a_2+9a_1^2a_3^2+96a_4^2-60a_1a_3a_4} \\
m_4 &= \frac{\begin{array}{c}-a_1^4+8a_2a_1^2-12a_3a_1+12a_2^2-24a_4-2(12a_3a_2^3 \\ -8a_1a_4a_2^2-18a_1a_3^2a_2+12a_3a_4a_2+27a_3^3-24a_1a_4^2 \\ +12a_1^2a_3a_4)m_5-8(4a_4a_2^3-8a_4^2a_2-6a_1a_3a_4a_2+2a_1^2a_4^2+9a_3^2a_4)m_6\end{array}}{16a_2^4-36a_1a_3a_2^2-80a_4a_2^2+90a_3^2a_2+16a_1^2a_4a_2+9a_1^2a_3^2+96a_4^2-60a_1a_3a_4}
\end{aligned}$$

Similarly, to what we did for the three-moment filter, perhaps optimizing the variance and kurtosis with respect to m_5 and m_6 can give us some possible closures. The idea would

be computing the gradients

$$\begin{aligned}\nabla\sigma^2 &= \begin{pmatrix}\frac{\partial\sigma^2}{\partial m_5}\\ \frac{\partial\sigma^2}{\partial m_6}\end{pmatrix}\\ \nabla\kappa &= \begin{pmatrix}\frac{\partial\kappa}{\partial m_5}\\ \frac{\partial\kappa}{\partial m_6}\end{pmatrix}\end{aligned}$$

and setting to 0 and then solving for m_5 and m_6. Then once we find the equations for m_5 and m_6, we can then plug them back in to the above equations and then find the representation of the posterior moments in terms of the quartic's coefficients.

Once can see from this process that while we do not have a simple formula for the posterior moments solely in terms of prior moments and likelihood moments, we have a general procedure that can be applied to an algorithm. Now we will discuss other approaches that could be developed more in the future.

5.2 Minimum Mean-Square Estimate (MMSE) Approach

The Kalman filter can also be derived by by minimizing the expected value of the square of the magnitude of this vector, $\mathbb{E}[\|x_k - \hat{x}_{k|k}\|^2]$ which is equivalent to minimizing the trace of the posterior estimate of the covariance matrix $P_{k|k}$. The derivation begins by making the ansatz

$$\begin{aligned}\hat{x}_{k|k} &= \begin{bmatrix}A_k & B_k\end{bmatrix}\begin{bmatrix}\hat{x}_{k|k-1}\\ \hat{y}_k\end{bmatrix}\\ &= A_k\hat{x}_{k|k-1} + B_k\hat{y}_k\end{aligned}$$

which is called a linear filter since the next estimate is a linear combination of the previous estimate and the observation. Constraining the filter to be unbiased, meaning the $\mathbb{E}[\hat{x}_{k|k}] = x_k$, we find that $A_k = I - B_k H_k$ so the filter becomes

$$\hat{x}_{k|k} = (I - B_k H_k)\hat{x}_{k|k-1} + B_k \hat{y}_k.$$

Now minimize over all possible B_k, the trace of the covariance matrix $P_{k|k}$ and we find that the minimum is achieved by the Kalman gain matrix so $B_k = K_k$. This gives the MMSE filter

$$\hat{x}_{k|k} = (I - K_k H_k)\hat{x}_{k|k-1} + K_k \hat{y}_k,$$

which is identical to the Kalman filter.

A possible way to generalize the MMSE approach is by looking for look for optimal quadratic filters where we propose a quadratic ansatz as follows

$$\hat{x}_{k|k} = C_k \begin{bmatrix} \hat{x}_{k|k-1} \\ \hat{y}_k \end{bmatrix} + D_k \times_3 \begin{bmatrix} \hat{x}_{k|k-1} \\ \hat{y}_k \end{bmatrix} \times_2 \begin{bmatrix} \hat{x}_{k|k-1} \\ \hat{y}_k \end{bmatrix}$$

where C_k is a $n \times (n+m)$ matrix and D_k is a $n \times (n+m) \times (n+m)$ 3-tensor in which n is the dimension of the state and m is the dimension of the observation.

Notice that,

$$\begin{aligned}
\left(C \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} + D \times_3 \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} \times_2 \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} \right)_i &= \sum_{j=1}^{n} C_{ij}\hat{x}_j + \sum_{j=n+1}^{n+m} C_{ij}\hat{y}_{j-n} \\
&\quad + \sum_{j=1}^{n} \left[\sum_{k=1}^{n} D_{ijk}\hat{x}_k + \sum_{k=n+1}^{n+m} D_{ijk}\hat{y}_{k-n} \right] \hat{x}_j \\
&\quad + \sum_{j=n+1}^{n+m} \left[\sum_{k=1}^{n} D_{ijk}\hat{x}_k + \sum_{k=n+1}^{n+m} D_{ijk}\hat{y}_{k-n} \right] \hat{y}_{j-n} \\
&= \sum_{j=1}^{n} C_{ij}\hat{x}_j + \sum_{j=n+1}^{n+m} C_{ij}\hat{y}_{j-n} + \sum_{j=1}^{n}\sum_{k=1}^{n} D_{ijk}\hat{x}_k\hat{x}_j \\
&\quad + \sum_{j=1}^{n}\sum_{k=n+1}^{n+m} D_{ijk}\hat{y}_{k-n}\hat{x}_j + \sum_{j=n+1}^{n+m}\sum_{k=1}^{n} D_{ijk}\hat{x}_k\hat{y}_{j-n} \\
&\quad + \sum_{j=n+1}^{n+m}\sum_{k=n+1}^{n+m} D_{ijk}\hat{y}_{k-n}\hat{y}_{j-n}
\end{aligned}$$

is a quadratic polynomial in the entries of $\hat{x}$ and $\hat{y}$.

Recall that $\mathbb{E}[\hat{y}_k] = Hx_k$ and assume that $\mathbb{E}[\hat{x}_{k|k-1}] = x_k$, then

$$\begin{aligned}
\left(\mathbb{E}\left[C \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} + D \times_3 \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} \times_2 \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} \right] \right)_i &= \sum_{j=1}^{n} C_{ij}\mathbb{E}[\hat{x}_j] + \sum_{j=n+1}^{n+m} C_{ij}\mathbb{E}[\hat{y}_{j-n}] + \sum_{j=1}^{n}\sum_{k=1}^{n} D_{ijk}\mathbb{E}[\hat{x}_k\hat{x}_j] \\
&\quad + \sum_{j=1}^{n}\sum_{k=n+1}^{n+m} D_{ijk}\mathbb{E}[\hat{y}_{k-n}\hat{x}_j] + \sum_{j=n+1}^{n+m}\sum_{k=1}^{n} D_{ijk}\mathbb{E}[\hat{x}_k\hat{y}_{j-n}] \\
&\quad + \sum_{j=n+1}^{n+m}\sum_{k=n+1}^{n+m} D_{ijk}\mathbb{E}[\hat{y}_{k-n}\hat{y}_{j-n}]
\end{aligned}$$

and using the fact that $P_{kj} = \mathbb{E}[(\hat{x}_k - x_k)(\hat{x}_j - x_j)]$ and $R_{kj} = \mathbb{E}[\nu_k\nu_j]$ is the covariance

matrix of the noise,

$$\left(\mathbb{E}\left[C\begin{bmatrix}\hat{x}\\ \hat{y}\end{bmatrix} + D\times_3\begin{bmatrix}\hat{x}\\ \hat{y}\end{bmatrix}\times_2\begin{bmatrix}\hat{x}\\ \hat{y}\end{bmatrix}\right]\right)_i = \sum_{j=1}^{n} C_{ij}x_j + \sum_{j=n+1}^{n+m} C_{ij}(Hx)_{j-n} + \sum_{j=1}^{n}\sum_{k=1}^{n} D_{ijk}(P_{kj}+x_kx_j)$$
$$+\sum_{j=1}^{n}\sum_{k=n+1}^{n+m} D_{ijk}(Hx)_{k-n}x_j + \sum_{j=n+1}^{n+m}\sum_{k=1}^{n} D_{ijk}x_k(Hx)_{j-n}$$
$$+\sum_{j=n+1}^{n+m}\sum_{k=n+1}^{n+m} D_{ijk}\left((Hx)_{k-n}(Hx)_{j-n}+R_{kj}\right)]$$

For the filter to be unbiased, we need to set the right side of the above equation equal to x_i.

We still want an unbiased filter and the minimized variance. We will probably have room for more constraints as we could even consider minimizing the kurtosis to help us look for the best possible C_k and D_k. There are some concerns however. The linear filter has some nice properties that we are kind of giving up by going to this more strongly nonlinear form. Stability for one thing might be an issue with a quadratic filter so we may need to use our constraints to enforce stability.

5.3 Closure Approach

The evolution of the posterior probability density, $p(x,t)$, is described by the Kushner partial differential equation

$$dp = \mathcal{L}^* p\,dt + p(h-\overline{h})R^{-1}\,dz$$

where

$$\mathcal{L}^* p = -\sum_i \frac{\partial}{\partial x_i}(f_i p) + \sum_{i,j}\frac{\partial^2}{\partial x_j \partial x_i}\left(Q_{ij}p\right)$$

is the Kolmogorov Forward operator where $Q_{ij} = \frac{1}{2}(qq^\top)_{ij} = \frac{1}{2}\sum_k q_{ik}q_{jk}$ is the system noise covariance matrix, h is the observation function,

$$\overline{h} = \mathbb{E}_p[h(x,t)] = \int h(x,t)p(x,t)\,dx$$

is the expected observation, $R_{ij} = (rr^\top)_{ij} = \sum_k r_{ik}r_{jk}$ is the observation noise covariance matrix, and

$$dz = dy - \overline{h}dt$$

is the the innovation process where dy are the true observations.

Starting off from this Kushner partial differential equation, we can compute ordinary differential equations for the mean and variance. Recall that the mean is

$$\mu(t) = \mathbb{E}[x] = \int xp(x,t)\,dx.$$

Then the derivative of the mean is

$$\dot{\mu}(t) = \int x\,\frac{\partial}{\partial t}p(x,t)\,dx.$$

We then substitute $\frac{\partial}{\partial t}p(x,t)$ with the Kushner equation and integrate by parts and find that

$$\dot{\mu}(t) = \left(f(x_0) + Df(x_0)(\mu - x_0)\right)\,dt + \sigma Dh(x_0)^\top R^{-1}\,dz.$$

Similarly we can compute a dynamical system for the covariance

$$\dot{\sigma} = \left(\sigma F^\top + F\sigma + Q - \sigma H^\top R^{-1} H\sigma^\top\right)\,dt + S\cdot H^\top R^{-1}\,dz$$

where S is the skewness tensor. If we assume the skewness equals zero (i.e. assume the

distribution is gaussian), this closure gives us the Kalman equations.

We propose to go further by deriving the ordinary differential equations for the skewness and kurtosis. Instead of assuming the skewness is zero, we will find new closures for the first four moments rather than just the first two moments. One downside is that when we get to the equation for the kurtosis, it will involve the fifth moment so we are going to have to make a closure there. The easiest thing would be to just assume that the fifth moment is zero. We will try to find more realistic closures in the future.

Bibliography

[1] D. C. Easley and T. Berry, "A higher order unscented transform," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 9, no. 3, pp. 1094–1131, 2021.

[2] S. J. Julier and J. K. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," Technical report, Robotics Research Group, Department of Engineering Science, Tech. Rep., 1996.

[3] ——, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[4] R. Van Der Merwe, A. Doucet, N. De Freitas, and E. A. Wan, "The unscented particle filter," in *Advances in neural information processing systems*, 2001, pp. 584–590.

[5] T. Berry and T. Sauer, "Adaptive ensemble Kalman filtering of non-linear systems," *Tellus A: Dynamic Meteorology and Oceanography*, vol. 65, no. 1, p. 20331, 2013.

[6] S. Särkkä, "Unscented rauch–tung–striebel smoother," *IEEE transactions on automatic control*, vol. 53, no. 3, pp. 845–849, 2008.

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

[8] E. A. Wan, R. Van Der Merwe, and A. T. Nelson, "Dual estimation and the unscented transformation," in *Advances in neural information processing systems*, 2000, pp. 666–672.

[9] F. Hamilton, T. Berry, N. Peixoto, and T. Sauer, "Real-time tracking of neuronal network structure using data assimilation," *Physical Review E*, vol. 88, no. 5, p. 052715, 2013.

[10] F. Hamilton, T. Berry, and T. Sauer, "Ensemble Kalman filtering without a model," *Physical Review X*, vol. 6, no. 1, p. 011021, 2016.

[11] J. Harlim, A. J. Majda *et al.*, "Catastrophic filter divergence in filtering nonlinear dissipative systems," *Communications in Mathematical Sciences*, vol. 8, no. 1, pp. 27–43, 2010.

[12] J. L. Anderson, "An adaptive covariance inflation error correction algorithm for ensemble filters," *Tellus A: Dynamic Meteorology and Oceanography*, vol. 59, no. 2, pp. 210–224, 2007.

[13] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on automatic control*, vol. 54, no. 6, pp. 1254–1269, 2009.

[14] B. Jia, M. Xin, and Y. Cheng, "Sparse-grid quadrature nonlinear filtering," *Automatica*, vol. 48, no. 2, pp. 327–341, 2012.

[15] F. Nobile, R. Tempone, and C. G. Webster, "A sparse grid stochastic collocation method for partial differential equations with random input data," *SIAM Journal on Numerical Analysis*, vol. 46, no. 5, pp. 2309–2345, 2008.

[16] D. Xiu and G. E. Karniadakis, "The wiener–askey polynomial chaos for stochastic differential equations," *SIAM journal on scientific computing*, vol. 24, no. 2, pp. 619–644, 2002.

[17] O. Le Maître and O. M. Knio, *Spectral methods for uncertainty quantification: with applications to computational fluid dynamics*. Springer Science & Business Media, 2010.

[18] W. Hackbusch, *Tensor spaces and numerical tensor calculus*. Springer, 2012, vol. 42.

[19] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[20] J. Haastad, "Tensor rank is np-complete," in *International Colloquium on Automata, Languages, and Programming*. Springer, 1989, pp. 451–460.

[21] C. J. Hillar and L.-H. Lim, "Most tensor problems are np-hard," *Journal of the ACM (JACM)*, vol. 60, no. 6, pp. 1–39, 2013.

[22] E. Kofidis and P. A. Regalia, "On the best rank-1 approximation of higher-order supersymmetric tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 3, pp. 863–884, 2002.

[23] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques," *GAMM-Mitteilungen*, vol. 36, no. 1, pp. 53–78, 2013.

[24] A. Falco and A. Nouy, "A proper generalized decomposition for the solution of elliptic problems in abstract form by using a functional Eckart–Young approach," *Journal of Mathematical Analysis and Applications*, vol. 376, no. 2, pp. 469–480, 2011.

[25] L. De Lathauwer, P. Comon, B. De Moor, and J. Vandewalle, "Higher-order power method," *Nonlinear Theory and its Applications, NOLTA'95*, vol. 1, pp. 91–96, 1995.

[26] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proceedings of 1995 American Control Conference-ACC'95*, vol. 3. IEEE, 1995, pp. 1628–1632.

[27] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal processing, sensor fusion, and target recognition VI*, vol. 3068. International Society for Optics and Photonics, 1997, pp. 182–193.

[28] S. J. Julier, "Skewed approach to filtering," in *Signal and Data Processing of Small Targets 1998*, vol. 3373. International Society for Optics and Photonics, 1998, pp. 271–282.

[29] ——, "The scaled unscented transformation," in *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)*, vol. 6. IEEE, 2002, pp. 4555–4559.

[30] A. Stegeman and P. Comon, "Subtracting a best rank-1 approximation may increase tensor rank," *Linear Algebra and its Applications*, vol. 433, no. 7, pp. 1276–1300, 2010.

[31] T. G. Kolda, "A counterexample to the possibility of an extension of the Eckart–Young low-rank approximation theorem for the orthogonal rank tensor decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 24, no. 3, pp. 762–767, 2003.

[32] L. De Lathauwer, B. De Moor, and J. Vandewalle, "On the best rank-1 and rank-$(r_1, r_2, \ldots r_n)$ approximation of higher-order tensors," *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324–1342, 2000.

[33] P. A. Regalia and E. Kofidis, "The higher-order power method revisited: convergence proofs and effective initialization," in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100)*, vol. 5. IEEE, 2000, pp. 2709–2712.

[34] S. Banach, "Über homogene polynome in (l^2)," *Studia Mathematica*, vol. 7, no. 1, pp. 36–44, 1938.

[35] Z. Li, Y. Nakatsukasa, T. Soma, and A. Uschmajew, "On orthogonal tensors and best rank-one approximation ratio," *SIAM Journal on Matrix Analysis and Applications*, vol. 39, no. 1, pp. 400–425, 2018.

[36] K. Atkinson and W. Han, *Theoretical numerical analysis*. Springer, 2005, vol. 39.

[37] D. L. Ragozin, "Constructive polynomial approximation on spheres and projective spaces." *Transactions of the American Mathematical Society*, vol. 162, pp. 157–170, 1971.

[38] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of the atmospheric sciences*, vol. 20, no. 2, pp. 130–141, 1963.

[39] W. Hao and J. Harlim, "An equation-by-equation method for solving the multidimensional moment constrained maximum entropy problem," *Communications in Applied Mathematics and Computational Science*, vol. 13, no. 2, pp. 189–214, 2018.

Curriculum Vitae

Education

Ph.D. Candidate in Mathematics, Expected graduation: May 2022, George Mason University, Dissertation: *Higher Order Kalman Filtering for Nonlinear Systems*, Advisor: Dr. Tyrus Berry

M.S. in Mathematics, May 2017, George Mason University

B.S. (Summa Cum Laude) in Mathematics, Minor in Italian, May 2015, George Mason University

Teaching and Research Experience

Oct 7, 2021 – Present	*Guest Researcher* NIST, Gaithersburg Researching Nitrogen Vacancy (NV) Diamond-Based Quantum Metrology with Dr. Zeeshan Ahmed and Dr. Tyrus Berry
Aug 25, 2018 – Mar 24, 2022	*Research Assistant* George Mason University, Fairfax Researching higher order methods in data assimilation with Dr. Tyrus Berry
May 25, 2019 – Jun 24, 2019	*Course Instructor* George Mason University, Fairfax Taught Precalculus.
May 25, 2018 – Jun 24, 2018	*Course Instructor* George Mason University, Fairfax Taught Precalculus.
Jan 10, 2018 – May 24, 2018	*Teaching Assistant* George Mason University, Fairfax Taught three recitations of Analytic Geometry and Calculus III for Instructor Nacir Hmidouch and one recitation of Elementary Differential Equations for Adjunct Prof. Chistopher Paldino.

Aug 25, 2017 – Jan 9, 2018	*Teaching Assistant* George Mason University, Fairfax Taught three recitations of Analytic Geometry and Calculus III for Dr. Thomas Wanner and three recitations of Analytic Geometry and Calculus I for Dr. Flavia Colonna.
Jan 10, 2017 – May 24, 2017	*Teaching Assistant* George Mason University, Fairfax Taught three recitations of Analytic Geometry and Calculus III for Asst. Prof. Kumnit Nong.
Jan 22, 2017 – May 27, 2017	*Grader* George Mason University, Fairfax Graded Linear Algebra exams for Dr. Neil Epstein.
Aug 25, 2016 – Jan 9, 2017	*Teaching Assistant* George Mason University, Fairfax Taught two recitations of Analytic Geometry and Calculus III for Dr. Igor Griva and one recitation of Analytic Geometry and Calculus III Honors for Dr. Robert L. Sachs.
Mar 20, 2016 – May 28, 2016	*Grader* George Mason University, Fairfax Graded Introductory Calculus with Business Applications quizzes and tests for Prof. Karen L Crossin.
Aug 31, 2015 – Dec 26, 2015	*Grader* George Mason University, Fairfax Graded Linear Algebra homework for Dr. Jeng-Eng Lin.

Talks and Conferences

- October 6, 2021: Sensor Science Division Seminar, National Institute of Standards and Technology (NIST). *Learning Hidden States from Noisy Observations.*
- August 27, 2021: Dynamics Days Europe 2021. *Generalizing the Unscented Ensemble Transform to Higher Moments.*
- August 25, 2020: Dynamics Days Digital 2020. *Generalizing the Unscented Ensemble Transform to Higher Moments.*
- August 14, 2020: SIAM Conference on Mathematics of Planet Earth (MPE20). *Generalizing the Unscented Ensemble Transform to Higher Moments.*
- November 22, 2019: Student Research Talks, George Mason University. *Generalizing the Unscented Ensemble Transform to Higher Moments.*
- February 26, 2019: Mathematics and Climate Seminar, George Mason University. *Lorenz Equations.*

Awards

2021 *Clarke Family Award for Excellence in Algebra, Analysis, and Topology*

This award was established in 2015 by Robert W. Clarke to provide scholarships to encourage and recognize graduate student excellence in the study of mathematics and carries a monetary prize of $1,000.

2015 *William Weaver Prize in Italian Studies*

This award was established in the name of the late pre-eminent translator of Italian literature into English and recognizes the academic achievements of students of Italian. It is a monetary prize in the amount of $1,000.

2015 *Excellence in Advanced Italian*

This award recognizes one's progress and efforts at study of Italian language and culture at the advanced level for the 2014-2015 academic year.

2015 *Excellence in Advanced Italian*

This award recognizes one's progress and efforts at study of Italian language and culture at the advanced level for the 2013-2014 academic year.

Organizations

Association for Women In Mathematics (GMU Chapter)

President	April 2020 - April 2022
Treasurer	March 2018 - April 2020
Vice President	April 2017 - March 2018