Deka, B., Ha Nguyen, L., Amiri, S and Goulet, J-A. (2022). *The Gaussian Multiplicative Approximation for State-Space Models. Structural Control and Health Monitoring. Volume 29, Issue 3, e2904.*

The Gaussian Multiplicative Approximation for State-Space Models

Bhargob Deka,* Luong Ha Nguyen, Saeid Amiri, and James-A. Goulet
Department of Civil, Geologic and Mining Engineering
Polytechnique Montreal, CANADA

March 15, 2022

Abstract

Applications such as structural health monitoring (SHM) often rely on the analysis of time-series using methods such as state-space models (SSM). In this paper, we propose an analytical method called the *Gaussian multiplicative approximation* (GMA) that is applicable to multiplicative state-space models that are often encountered in practical SHM applications. The method enables the analytical inference of the mean vector and the covariance matrix for the product of two hidden states in the transition and/or observation models using linear estimation theory and the online estimation of model parameters as hidden states. The potential of combining the GMA and Bayesian dynamic linear models (BDLM) is illustrated through the development of (1) a generic component called *online autoregressive* that can estimate both the state variable and the parameter together; (2) a generic component called *trend multiplicative* for multiplicative seasonality model to identify non-harmonic periodic pattern whose amplitude changes linearly with time; and (3) a generic component called *double kernel regression* to identify non-harmonic periodic pattern that involves the product of two periodic kernel regression components. The SHM-based case studies presented confirm that the GMA exceeds the performance of the existing nonlinear Kalman filter methods in terms of accuracy along with the computational cost.

Keywords: state-space models, online parameter estimation, Gaussian multiplicative approximation, nonlinear Kalman filter, structural health monitoring, BDLM.

1 Introduction

Time series modeling is an integral part of data-driven structural health monitoring (SHM) [14, 50, 53, 57, 64]. Statistical and machine learning (ML) techniques have been applied in data-based methods for predicting structural responses [14, 50, 53] and risk assessment of civil structures [29, 30]. The ML techniques can be classified into two types: *supervised learning* and *unsupervised learning*. Supervised learning [42] consists of learning patterns in data using known sequences of input and output values, i.e., labeled data. On the other hand, unsupervised learning enables learning the underlying hidden pattern only by using input values without any knowledge of the corresponding output values. There are many techniques used for data interpretation in SHM, such as pattern-recognition techniques, regression methods, and ARMA, among many others, see [9, 62, 76]. Multiple linear regression (MLR) [21, 37, 65] and artificial neural networks (ANN) [39] are extensively used to model time series using labeled data by formulating the relationship between the known observed response and the covariates [16]. Time series decomposition employs *component models*, [16] such as

*Corresponding author: bhargobdeka11@gmail.com

additive or multiplicative models to decompose a time series into specific patterns having their own mathematical formulation. However, these methods have several limitations as they are not efficient in modeling non-stationary components, require a large amount of data, and requires retraining of the model as new data is collected.

The Bayesian dynamic linear models (BDLM) [26] are one type of state-space models (SSM) with linear transition and observation models that can be broken down in components. These models are used in the field of data-driven SHM [17, 50, 58, 73], but not restricted to [75], as it is suitable for non-stationary components, provide interpretable components and can perform online prediction without retraining the model. The BDLM uses generic structured components to identify specific patterns in structural responses that evolves with time. Each of the generic components has their own mathematical formulation comprising of one or more hidden states. *Local level* (`LL`), *local trend* (`LT`) and *local acceleration* (`LA`) components are used to model the baseline of any structural response without external effects such as temperature, pressure or traffic loads. These components capture the irreversible pattern which can be used to identify an anomaly in the behavior of the structure [46]. *Periodic* (`PD`) and *kernel regression* (`KR`) components are used to identify external effects having periodic pattern that can be, respectively, harmonic or non-harmonic in nature [45]. The autoregressive component (`AR`) is used in combination with these components to capture the residuals, i.e., any physical phenomenon that is not captured by the other structured components. Each component in the BDLM commonly involves unknown model parameters that need to be estimated. This can be done using offline learning methods such as Expectation-Maximization (EM) algorithm [23] or gradient-based methods [22], which consider the parameters to be invariant over time. The aforementioned procedure has the disadvantage of being computationally expensive in the presence of a large number of unknown model parameters and often converges to local maxima when the initial parameter values are poor [48]. Online estimation of parameters is possible using the Rao-Blackwellized Particle Filtering (RBPF) [48]. This method provides consistent estimates but at a high computational cost owing to the use of sampling methods. Hence, a major limitation in the existing BDLM framework is the inability to perform the online estimation of parameters in an analytically tractable manner with a low computational cost. In this article, analytical tractability is defined as the ability to compute exactly the statistical properties of a random variable using algebraic expressions. Being able to multiply hidden states would allow, for many cases, to treat model parameters as hidden states and thus to perform online estimation. Using existing methods, it is possible to perform the product of hidden states using nonlinear filtering methods such as the *unscented Kalman filter* (UKF) [67] and the *cubature Kalman filter* (CKF) [5]. Nevertheless, these methods are computationally demanding as they use sample points to approximate the posterior probability density function (PDF).

Using related and existing work [33, 63], this article presents explicitly all the moment equations for performing the analytically tractable estimation for the product of two hidden states using either Gaussian moment generating function or 2^{nd} order Taylor series expansion; We refer to this method as the *Gaussian multiplicative approximation* (GMA). The potential of combining the GMA and the BDLM is illustrated through the development of (1) a generic component called *online autoregressive* (`OAR`) that can estimate both the `AR` state ($x^{\texttt{AR}}$) and the `AR` parameter ($\phi^{\texttt{AR}}$) together; (2) a generic component called *trend multiplicative* (`TM`) for multiplicative seasonality model to identify non-harmonic periodic pattern whose amplitude changes linearly with time; and (3) a generic component called *double kernel regression* (`DKR`) to identify non-harmonic periodic pattern that involves the product of two periodic kernel regression components. Therefore, instead of using nonlinear filtering methods, the Kalman filter is still applicable for the nonlinear dynamic systems having product terms by using the GMA in the BDLM framework.

The layout of the paper is as follows. Section 2 presents the Gaussian filters used in nonlinear

state estimation. Section 3 introduces the GMA for computing the moments associated with the product of two Gaussian random variables and its application in state estimation using the BDLM. Section 4 offers the application of the GMA through new generic components called a) `OAR` for online estimation of the `AR` process; b) `TM` for modeling periodic pattern with linearly changing amplitude; and c) `DKR` for modeling periodic pattern with multiple periodicities, along with illustrative examples. Finally, section 5 presents the conclusion drawn from the results of the proposed method.

2 Gaussian Filters

This section presents two of the main Gaussian filters used for nonlinear functions, namely the UKF and the CKF, which are widely used in the literature.

2.1 Moment calculation using a Gaussian approximation

In the Gaussian filters, the prior $p(x_t|y_{t-1})$ and the likelihood $p(y_t|y_{t-1})$ of the hidden states are assumed to be Gaussian which makes the posterior $p(x_t|y_t)$ of the hidden states also Gaussian [5]. When the dynamic model is linear, the Gaussian filter is equivalent to the Kalman filter. The Gaussian assumption in the Kalman filter is useful as it makes the statistical properties of the conditional distribution analytically tractable, i.e., there is a closed-form solution available for the first two moments of the states. Hence, the Kalman filter reduces to calculating the mean and covariance matrices recursively to get the exact posterior PDF of the hidden states.
In the case of nonlinear systems, closed-form solutions are not available and requires approximations for the posterior PDF leading to sub-optimal solutions [1]. There are two main approaches in the literature for performing this approximation:

a. Local approach: In this approach the posterior PDF is assumed to be known and computed using numerical approximations [5]. The approximation is performed either through linearisation of the nonlinear function (Extended Kalman filter) [38] or by approximating the posterior PDF directly using weighted samples (UKF and CKF).

b. Global approach: In this approach, there is no assumption made for the type of the posterior PDF. The particle filter is an example of this approach, where the posterior PDF can have any type and is approximated using sampling methods [13]. This approach has the limitation of having high computational cost associated with sampling and can be inefficient for online state estimation [1].

The methods under the category of local approach are discussed further owing to their advantages of being accurate, analytically tractable as well as having a low computational cost, hence useful for online state estimation. The aforementioned techniques approximate the following moment integral,

$$I = \int g(x) \cdot f(x) dx,$$

where $g(x)$ is the nonlinear function defining the transition and observation models, and $f(x)$ is a known PDF of the state which can also be Gaussian. The numerical approximation reduces the integral to the form,

$$I \approx \sum_{i=1}^{n} w_i \cdot g(x_i),$$

where x_i are the samples, n is the number of samples, and w_i are the associated weights.

2.2 Unscented Kalman Filter (UKF)

The UKF is a nonlinear filtering method that approximates the posterior PDF using a set of points called the sigma points and an unscented transform (UT) [35] method for the associated weights [67]. The sigma points are generated symmetrically around the prior mean which has a considerably higher weight than the other points. These sigma points are propagated through the nonlinear model to estimate the mean and covariance of the posterior PDF, which is accurate to the third order for any nonlinear function [67]. Hence, the UKF is a derivative free method that is more accurate than the EKF [1].
Even though the UKF is better than the EKF, it suffers from instabilities and numerical inaccuracies [1, 5]. The covariance matrix may result in non positive-semi-definite (psd) owing to round-off errors introduced by sensitive numerical operations such as matrix square rooting, matrix inversion, and covariance update through matrix subtraction. Hence, a square root version of the UKF is necessary to prevent numerical ill-conditioning due to arithmetic imprecision even though the computational complexity increases [5]. However, a stable version of the square-root UKF is still not guaranteed owing to the presence of negatively weighted samples to update the posterior covariance matrix, which can still result in non-psd outcomes.

2.3 Cubature Kalman Filter (CKF)

The CKF is a nonlinear filtering method similar to the UKF, which is used to compute the multivariate moment integrals associated with nonlinear functions using numerical integration methods [5]. It approximates the Gaussian weighted moment integral by using weighted samples, which are created using the cubature rule. The cubature rule provides exact solutions when the nonlinear function for a set of hidden states of degree d is in the form

$$g(\boldsymbol{x}) = x_1^{d_1} x_2^{d_2} \dots x_n^{d_n},$$

where d_i are non-negative integers and $\sum_{i=1}^{n} d_i \leq d$. A third-degree spherical-radial cubature rule is used to compute the posterior mean exactly and the posterior covariance approximately [1, 5]. Using the invariant theory [61] along with the third degree cubature rule, $2n$ samples are generated where n is the size of the hidden state vector. The weights and the samples are only dependent on the size of the state vector and are independent of the nonlinear function $g(\boldsymbol{x})$. The CKF formulation is numerically more stable than the UKF and has the same computational complexity; hence the CKF is used in this article for comparison with the GMA for the online estimation of parameters.

3 Gaussian Multiplication Approximation (GMA)

This section presents the *Gaussian multiplicative approximation* for computing the moments associated with the product of two Gaussian random variables and its application in state estimation using the BDLM.

3.1 Moments of Product Term

Consider the case where the variables $\boldsymbol{x} = [x_1\, x_2]^\intercal$ are the input of the nonlinear function,

$$g(\boldsymbol{x}) = x_1 x_2. \tag{1}$$

The goal is to infer the probability density function (PDF) of $\boldsymbol{X}$ indirectly, using an observation y that is defined such that,

$$y = g(\boldsymbol{x}) + v,\ v : V \sim \mathcal{N}(v; 0, \sigma_v^2), \tag{2}$$

where V is a random variable representing the observation error with zero mean and variance σ_v^2. The posterior PDF of $\boldsymbol{X}$ given an observation y can be estimated using Bayes theorem as in,

$$f(\boldsymbol{x}|y) = \frac{f(\boldsymbol{x}, y)}{f(y)} = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{\boldsymbol{x}|y}, \boldsymbol{\Sigma}_{\boldsymbol{x}|y}), \tag{3}$$

which follows a Gaussian distribution with a mean vector $\boldsymbol{\mu}_{\boldsymbol{x}|y}$ and a covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|y}$ that are given by

$$\begin{aligned} \boldsymbol{\mu}_{\boldsymbol{x}|y} &= \boldsymbol{\mu}_{\boldsymbol{x}} + \frac{\boldsymbol{\Sigma}_{\boldsymbol{x}y}}{\sigma_y^2}(y - \mu_y), \\ \boldsymbol{\Sigma}_{\boldsymbol{x}|y} &= \boldsymbol{\Sigma}_{\boldsymbol{x}} - \frac{\boldsymbol{\Sigma}_{\boldsymbol{x}y} \cdot \boldsymbol{\Sigma}_{\boldsymbol{x}y}^{\intercal}}{\sigma_y^2}. \end{aligned}$$

Equation 3 holds when $f(\boldsymbol{x}, y)$ is Gaussian; In the case presented here, y is nonlinearly related to $\boldsymbol{x}$ through Equation 2 which makes the joint prior PDF, $f(\boldsymbol{x}, y)$, non-Gaussian. As the name indicates, the *Gaussian multiplicative approximation* (GMA) approximates the distribution of X_1X_2 as a Gaussian random variable for which the expected value, variance and covariance can be calculated exactly using moment generating functions under the assumption that X_1 and X_2 are themselves Gaussian so that

$$\begin{aligned} \boldsymbol{\mu}_{\boldsymbol{x}} &= \mathbb{E}[\boldsymbol{X}] &&= \begin{bmatrix} \mathbb{E}[X_1] \\ \mathbb{E}[X_2] \end{bmatrix}, \\ \boldsymbol{\Sigma}_{\boldsymbol{x}} &= \mathrm{var}(\boldsymbol{X}) &&= \begin{bmatrix} \mathrm{var}(X_1) & \mathrm{cov}(X_1, X_2) \\ \mathrm{cov}(X_2, X_1) & \mathrm{var}(X_2) \end{bmatrix}. \end{aligned}$$

The mean and variance of Y can be obtained by propagating the uncertainty associated with $\boldsymbol{X}$ through the model described in Equation 2 so that

$$\begin{aligned} \mu_y &= \mathbb{E}[Y] &&= \mathbb{E}[g(\boldsymbol{X})] + \mathbb{E}[V] &&= \mathbb{E}[X_1X_2], \\ \sigma_y^2 &= \mathrm{var}(Y) &&= \mathrm{var}(X_1X_2) + \mathrm{var}(V) &&= \mathrm{var}(X_1X_2) + \sigma_v^2. \end{aligned}$$

Using the *moment generating function* (MGF) [12, 66] for the multivariate Gaussian distribution or 2^{nd} order Taylor series expansion (TSE) [34] of the product of the two Gaussian random variables, the first two moments associated with the product term X_1X_2 can be computed exactly using

$$\mathbb{E}[X_1X_2] = \mu_1\mu_2 + \mathrm{cov}(X_1, X_2), \tag{4}$$

$$\begin{aligned} \mathrm{var}(X_1X_2) = {} & \sigma_1^2\sigma_2^2 + \mathrm{cov}(X_1, X_2)^2 + 2\mathrm{cov}(X_1, X_2)\mu_1\mu_2 + \\ & \sigma_1^2\mu_2^2 + \sigma_2^2\mu_1^2, \end{aligned} \tag{5}$$

and the covariance between $\boldsymbol{X}$ and Y is given by

$$\boldsymbol{\Sigma}_{\boldsymbol{x}y} = \mathrm{cov}(\boldsymbol{X}, Y) = \begin{bmatrix} \mathrm{cov}(X_1, X_1X_2) \\ \mathrm{cov}(X_2, X_1X_2) \end{bmatrix}.$$

Similarly, we can also derive the exact solution for the covariance between the product term X_1X_2 and any other Gaussian random variable X_3,

$$\mathrm{cov}(X_3, X_1X_2) = \mathrm{cov}(X_1, X_3)\mu_2 + \mathrm{cov}(X_2, X_3)\mu_1. \tag{6}$$

Finally, for the general case, the covariance between any two pair of product terms is given by

$$\begin{aligned}\text{cov}(X_1X_2, X_3X_4) &= \text{cov}(X_1, X_3)\text{cov}(X_2, X_4) \\ &\quad +\text{cov}(X_1, X_4)\text{cov}(X_2, X_3) + \text{cov}(X_1, X_3)\mu_2\mu_4 \\ &\quad +\text{cov}(X_1, X_4)\mu_2\mu_3 + \text{cov}(X_2, X_3)\mu_1\mu_4 \\ &\quad +\text{cov}(X_2, X_4)\mu_1\mu_3,\end{aligned} \tag{7}$$

where X_1X_2 and X_3X_4 are the product terms of the Gaussian random variables X_1, X_2 and X_3, X_4, respectively. The derivation of the GMA equations using both MGF and TSE are presented in Appendix A.

3.1.1 State Estimation

In the context of state-space models, the state estimation for cases involving product terms in the transition model can be performed by combining linear estimation theory and the GMA. Given two state variables $\boldsymbol{x} = [x_1\ x_2]^\intercal$, a generic multiplication transition model involving product of the hidden states x_1 and x_2 is given by

$$\begin{aligned}x_{1,t} &= x_{1,t-1}x_{2,t} + w_{1,t}, \quad w_1 : W_1 \sim \mathcal{N}(0, \sigma^2_{w_1}), \\ x_{2,t} &= x_{2,t-1} + w_{2.t}, \qquad\quad w_2 : W_2 \sim \mathcal{N}(0, \sigma^2_{w_2}),\end{aligned} \tag{8}$$

where $w = [w_1 w_2]^\intercal$ is the vector of error term associated with the transition model. The hidden states at time $t-1$ is assumed to follow a Gaussian PDF with mean vector and covariance matrix given by

$$\boldsymbol{X}_{t-1|t-1} \sim \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{t-1|t-1}, \boldsymbol{\Sigma}_{t-1|t-1}),$$

where, $\boldsymbol{\mu}_{t-1|t-1} = \mathbb{E}[\boldsymbol{X}_{t-1}|y_{1:t-1}]$, $\boldsymbol{\Sigma}_{t-1|t-1} = \text{cov}(\boldsymbol{X}_{t-1}|y_{1:t-1})$ and $y_{1:t-1} = \{y_1, \ldots, y_{t-1}\}$. In its current form, the transition model given by Equation 8 is nonlinear. However, the nonlinear transition model can be formulated as a linear dynamic model by augmenting the state vector $\tilde{\mathbf{x}} = [\boldsymbol{x}\ x^\mathtt{p}]^\intercal$ so that

$$\tilde{\mathbf{X}}_{t-1|t-1} \sim \mathcal{N}(\tilde{\mathbf{x}}; \tilde{\boldsymbol{\mu}}_{t-1|t-1}, \tilde{\boldsymbol{\Sigma}}_{t-1|t-1}),$$

where $\tilde{\boldsymbol{\mu}}_{t-1|t-1} = \begin{bmatrix} \boldsymbol{\mu} \\ \mu^\mathtt{p} \end{bmatrix}_{t-1|t-1}$ and $\tilde{\boldsymbol{\Sigma}}_{t-1|t-1} = \begin{bmatrix} \boldsymbol{\Sigma} & \text{cov}(\boldsymbol{X}, X^\mathtt{p}) \\ \text{cov}(X^\mathtt{p}, \boldsymbol{X}) & (\sigma^\mathtt{p})^2 \end{bmatrix}_{t-1|t-1}$.

The hidden state variable $X^\mathtt{p} = X_1X_2$ represents the product term and is assumed to be Gaussian with expected value $\mu^\mathtt{p} = \mathbb{E}[X_1X_2]$ and variance $(\sigma^\mathtt{p})^2 = \text{var}(X_1X_2)$. The covariance terms between $\boldsymbol{X}$ and $X^\mathtt{p}$ in $\tilde{\boldsymbol{\Sigma}}_{t-1|t-1}$ is given by

$$\text{cov}(\boldsymbol{X}, X^\mathtt{p}) = \begin{bmatrix} \text{cov}(X_1, X_1X_2) \\ \text{cov}(X_2, X_1X_2) \end{bmatrix}.$$

Using linear algebra, the transition model in Equation 8 can be written as

$$\begin{aligned}\tilde{\mathbf{x}}_t &= \mathbf{A}\tilde{\mathbf{x}}_{t-1}, \\ \begin{bmatrix} x_1 \\ x_2 \\ x^\mathtt{p} \end{bmatrix}_t &= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x^\mathtt{p} \end{bmatrix}_{t-1},\end{aligned} \tag{9}$$

where $\tilde{\mathbf{x}} = [x_1\ x_2\ x^\mathtt{p}]^\intercal$. The augmented state vector $\tilde{\boldsymbol{X}}_{t|t-1}$ follows a Gaussian PDF given by

$$\tilde{\boldsymbol{X}}_{t|t-1} \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_{t|t-1}, \tilde{\boldsymbol{\Sigma}}_{t|t-1}), \tag{10}$$

where $\tilde{\boldsymbol{\mu}}_{t|t-1} = \mathbf{A}\tilde{\boldsymbol{\mu}}_{t-1|t-1}$ and $\tilde{\boldsymbol{\Sigma}}_{t|t-1} = \mathbf{A}\tilde{\boldsymbol{\Sigma}}_{t-1|t-1}\mathbf{A}^\intercal + \mathbf{Q}$, considering $\mathbf{Q}$ is the process noise covariance matrix. The variance terms in the $\mathbf{Q}$ has to be estimated using offline optimization algorithms that maximizes the log-likelihood function. The observation model is defined as

$$y_t = \mathbf{C}\tilde{\mathbf{x}}_t + v_t, \quad v : V \sim \mathcal{N}(0, \mathbf{R}), \tag{11}$$

where $\mathbf{C}$ is the observation matrix, v_t is the observation error, and $\mathbf{R} = \sigma_v^2$ is the observation noise covariance matrix. Using Equations 10 and 11, both the prediction and the update steps in the Kalman filter can be carried out for a nonlinear system having product terms. Note that in this paper, the application of the GMA equations is shown explicitly using a product term in the state vector to simplify the use of the method, hence the last row in the $\mathbf{A}$ matrix has 0's as the product term $x^{\mathtt{p}}$ is a placeholder. However, the GMA equations can also be applied implicitly without the need to store the information specifically in a variable. At each step $t-1$ of the recursive procedure, the GMA equations 4-7 are applied to compute the moments of the product term X_1X_2 using the moments of X_1 and X_2 obtained from the Kalman filter. The method can also be extended to more than one product terms in the state vector either by placing more placeholders as shown by Equation 9 or computing the moments implicitly. The case studies 2 and 3 in this article are examples of more than one product terms in the state vector.
The computational complexity of using the GMA for estimating a state vector of size n is $\mathcal{O}(3n^3)$. When using either the UKF or the CKF to perform the same state estimation, the computational complexity is to the order $\mathcal{O}(14n^3)$ [6]. Hence, using the GMA for state estimation of product terms is more than four times faster than using a state-of-the-art nonlinear filter like the CKF. The derivation of the computational complexity for the GMA is provided in Appendix C.

4 Applied Examples

This section presents three cases where the performance of the GMA and the CKF is compared for the task of estimating the state variables in SSM having product terms. Case study 1 presents the application of the proposed method for the online estimation of the state and parameter for a first order autoregressive process. Case studies 2 and 3 are issued from SHM application for real data.

4.1 Case Study 1: First order online autoregressive process (OAR)

4.1.1 Model Formulation

Consider the transition model for a first order autoregressive process (AR) given by

$$\underbrace{x_t^{\mathtt{AR}} = \phi^{\mathtt{AR}} x_{t-1}^{\mathtt{AR}} + w_t^{\mathtt{AR}}}_{\text{transition model}}, \overbrace{w_t^{\mathtt{AR}} : W \sim \mathcal{N}(w^{\mathtt{AR}}; 0, (\sigma^{\mathtt{AR}})^2)}^{\text{process noise}},$$

where $x^{\mathtt{AR}}$ is the AR hidden state, $\phi^{\mathtt{AR}}$ is the AR coefficient and W is the zero mean Gaussian noise that is independent of $X_{t-1}^{\mathtt{AR}}$. The AR coefficient $\phi^{\mathtt{AR}}$ can be estimated online by considering it as a hidden state x^{ϕ}. The new transition model is defined as

$$\begin{aligned} x_t^{\mathtt{AR}} &= x_t^{\phi} x_{t-1}^{\mathtt{AR}} + w_t^{\mathtt{AR}}, \\ x_t^{\phi} &= x_{t-1}^{\phi}, \end{aligned} \tag{12}$$

Table 1: Comparison of mean square error and log-likelihood estimates for the GMA and the CKF

metric	MSE		Log-likelihood, $\sum_{t=1}^{T} \ln f(\boldsymbol{x}_t \vert y_{1:t})$		
	$x^{\texttt{AR}}$	x^{ϕ}	$x^{\texttt{AR}}$	x^{ϕ}	$(x^{\texttt{AR}}, x^{\phi})$
GMA	3.3e-03 ± 1.9e-04	**2.9e-02 ± 1.6e-02**	1181.3 ± 34.2	2081.9 ± 250.2	3281.4 ± 263.2
CKF	3.3e-03 ± 1.6e-04	3.3e-02 ± 1.2e-02	**1199.5 ± 30.15**	**2175.3 ± 198.8**	**3386.4 ± 201.8**

where the hidden states at time step $t-1$ are $\boldsymbol{x}_{t-1} = [x^{\texttt{AR}}\ x^{\phi}]^{\intercal}_{t-1}$. The linear transition model for this case is given by Equation 9. The mean vector $\tilde{\boldsymbol{\mu}}_{t-1|t-1}$ and the covariance matrix $\tilde{\boldsymbol{\Sigma}}_{t-1|t-1}$ of $\tilde{\mathbf{x}}_{t-1}$ are given by

$$\tilde{\boldsymbol{\mu}}_{t-1|t-1} = \begin{bmatrix} \mathbb{E}[X^{\texttt{AR}}] \\ \mathbb{E}[X^{\phi}] \\ \mathbb{E}[X^{\phi}X^{\texttt{AR}}] \end{bmatrix}_{t-1|t-1},$$

$$\tilde{\boldsymbol{\Sigma}}_{t-1|t-1} = \begin{bmatrix} \mathrm{var}(X^{\texttt{AR}}) & \mathrm{cov}(X^{\texttt{AR}}, X^{\phi}) & \mathrm{cov}(X^{\texttt{AR}}, X^{\phi}X^{\texttt{AR}}) \\ \vdots & \mathrm{var}(X^{\phi}) & \mathrm{cov}(X^{\phi}, X^{\phi}X^{\texttt{AR}}) \\ \mathrm{sym.} & \dots & \mathrm{var}(X^{\phi}X^{\texttt{AR}}) \end{bmatrix}_{t-1|t-1},$$

where the elements of $\tilde{\boldsymbol{\mu}}_{t-1|t-1}$ and $\tilde{\boldsymbol{\Sigma}}_{t-1|t-1}$ can be computed analytically using Equations 4-7. The observation model is given by

$$y_t = \mathbf{C}\tilde{\mathbf{x}}_t + v_t, v_t : V \sim \mathcal{N}(v; 0, \sigma_v^2),$$

where the observation matrix $\mathbf{C} = [1\ 0\ 0]$.

4.1.2 Numerical example

Simulated data is generated from a first order AR process using the following parameters: $\sigma_v = 0.1$, $\sigma^{\texttt{AR}} = 0.05$ and $\phi^{\texttt{AR}} = 0.9$. Five datasets containing 1000 data points are generated using these parameters with a uniform time step of one unit. The prior knowledge of the hidden states are initialized by

$$\begin{aligned} \tilde{\boldsymbol{\mu}}_0 &= [0\ 0\ 0]^{\intercal}, \\ \tilde{\boldsymbol{\Sigma}}_0 &= \mathrm{diag}([100\ 100\ 0]). \end{aligned}$$

Both $\tilde{\boldsymbol{\mu}}_{t|t}$ and $\tilde{\boldsymbol{\Sigma}}_{t|t}$ of $\tilde{\mathbf{x}}_t$ are estimated using the GMA and the CKF methods. Table 1 shows the average results along with their standard deviation for the mean square error (MSE) [42] and log-likelihood (LL) [26, 42] values for $x^{\texttt{AR}}$, x^{ϕ}, as well as their joint log-likelihood. Figure 1 compares the actual and estimated hidden state values obtained using the GMA and the CKF methods, which shows the convergence of the estimated states to the true values while there is a slight difference early on in state estimation. The results presented in Table 1 shows that the predictive performance of both the methods using MSE and LL values have negligible discrepancies. The LL values using the CKF are slightly higher than the GMA owing to the difference in the state estimation at the initial stage, which disappears as the state estimates merge together and move towards the true values.

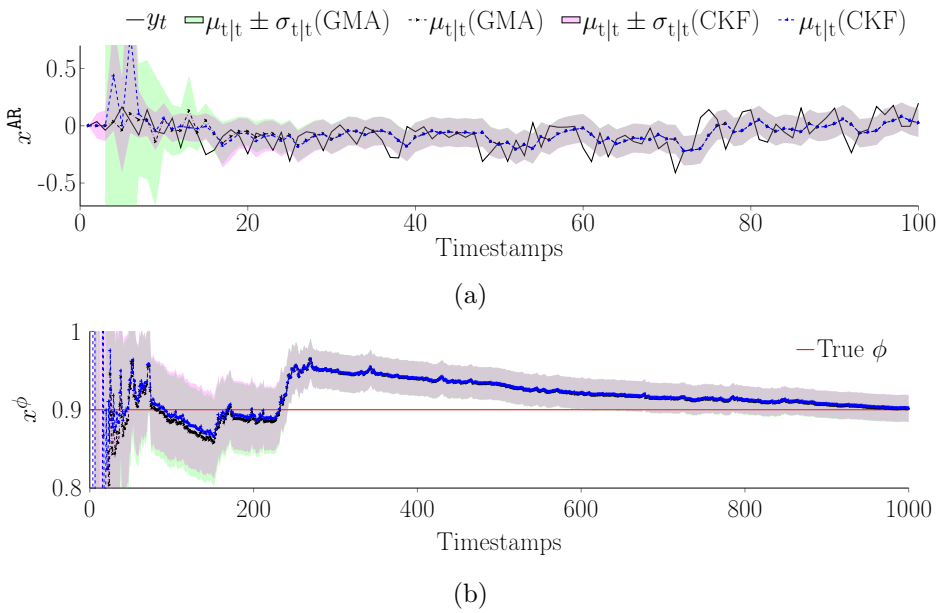

Figure 1: Comparison of the GMA and the CKF method for estimating a) $x^{\mathtt{AR}}$ and b) x^{ϕ}. Note that Figure 1a is a close-up view from the actual plot showing the first 100 time steps.

4.2 Case Study 2: Trend Multiplicative model

4.2.1 Data description

This case study is conducted on the water infiltration flow rate data [14,53,70] recorded on a concrete gravity dam in Canada. Such data is employed by engineers as a proxy for the dam health. The data ranges from September 26^{th} 2006 to December 31^{st} 2012. The raw data is averaged daily to have 2289 data points. The data have an increasing baseline along with a periodic component whose amplitude is increasing with time. A *multiplicative model* is the classical approach [16] to handle periodicity that varies with time, which can be performed by the product of baseline component with the static periodic component. The data are divided into a *training set* (1618 points) and a *test set* (671 points) to evaluate the predictive performance of the model. Figure 2 shows the entire dataset where the test set is represented by the shaded region.

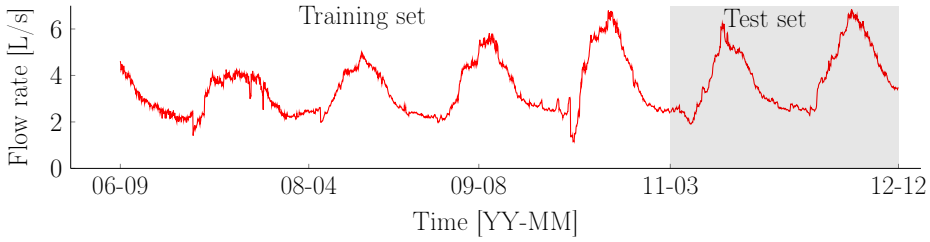

Figure 2: Flow rate data on the concrete gravity dam

4.2.2 Model Formulation

The components used for this model are a *local trend* (LT) that includes the *level* (LL) and the *trend* (LT) hidden states to model the baseline of the time series, a *periodic component* (S) that includes the amplitude ($\mathtt{S}_1$) to model the periodic pattern with a periodicity of one year, an online first order autoregressive component (OAR) to model the residual, and a new component called *trend multiplicative* (TM) to model the increasing amplitude with time. The TM component includes a new set of local level (LP) and trend (TP) hidden states to capture the constant rate of change in the local level of the periodic component. In this case, the transition model has two placeholders for the two product terms represented by $x_{t-1}^{\mathtt{P}_1} = \{x_{t-1}^{\mathtt{AR}} \cdot x_{t-1}^{\phi}\}$ for OAR and $x_t^{\mathtt{P}_2} = \{x_t^{\mathtt{LP}} \cdot x_t^{\mathtt{S}_1}\}$ for TM. The transition model for the new component TM is given by augmenting the product term, $x^{\mathtt{P}_2}$ to model the time-varying amplitude, with the local trend component provided by

$$\begin{bmatrix} x^{\mathtt{LP}} \\ x^{\mathtt{TP}} \\ x^{\mathtt{P}_2} \end{bmatrix}_t = \begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x^{\mathtt{LP}} \\ x^{\mathtt{TP}} \\ x^{\mathtt{P}_2} \end{bmatrix}_{t-1}.$$

The vector of hidden states of size $n = 10$ at time $t-1$ for all the components combined together is defined as

$$\boldsymbol{x}_{t-1} = [\overbrace{\underbrace{x^{\mathtt{LL}}\ x^{\mathtt{LT}}\ x^{\mathtt{S}_1}\ x^{\mathtt{S}_2}\ x^{\mathtt{AR}}\ x^{\phi}}_{\boldsymbol{x}^{\mathtt{S}_3}}\ x^{\mathtt{P}_1}}^{\boldsymbol{x}^{\mathtt{S}_1}}\ \overbrace{x^{\mathtt{LP}}\ x^{\mathtt{TP}}\ x^{\mathtt{P}_2}}^{\boldsymbol{x}^{\mathtt{S}_2}}]_{t-1}^{\intercal},$$

where $\boldsymbol{x}_{t-1}^{\mathtt{S}_1}$, $\boldsymbol{x}_{t-1}^{\mathtt{S}_2}$ and $\boldsymbol{x}_{t-1}^{\mathtt{S}_3}$ are subsets of the vector $\boldsymbol{x}_{t-1}$ of sizes n_1, n_2 and n_3 respectively. The prediction step in the Kalman filter is carried out sequentially using the mean vector $\boldsymbol{\mu}_{t-1|t-1}$ and the covariance matrix $\boldsymbol{\Sigma}_{t-1|t-1}$ of $\boldsymbol{X}_{t-1}$ given by

1. $\tilde{\boldsymbol{\mu}}_{1,t-1|t-1} = \boldsymbol{\mu}_{t-1|t-1} + \begin{bmatrix} \mathbf{0}_{n_1\times 1} \\ \mathbb{E}[X^{\mathtt{P}_1}] \\ \mathbf{0}_{n_2\times 1} \end{bmatrix}_{t-1|t-1},$

   $\tilde{\boldsymbol{\Sigma}}_{1,t-1|t-1} = \boldsymbol{\Sigma}_{t-1|t-1} + \begin{bmatrix} \mathbf{0}_{n\times n_1} & \mathrm{cov}(\mathbf{X}, X^{\mathtt{P}_1}) & \mathbf{0}_{n\times n_2} \end{bmatrix}_{t-1|t-1},$

2. $\tilde{\boldsymbol{\mu}}_{1,t|t-1} = \mathbf{A}\tilde{\boldsymbol{\mu}}_{1,t-1|t-1},$

   $\tilde{\boldsymbol{\Sigma}}_{1,t|t-1} = \mathbf{A}\tilde{\boldsymbol{\Sigma}}_{1,t-1|t-1}\mathbf{A}^{\intercal} + \mathbf{Q},$

3. $\tilde{\boldsymbol{\mu}}_{t|t-1} = \tilde{\boldsymbol{\mu}}_{1,t|t-1} + \begin{bmatrix} \mathbf{0}_{n_3\times 1} \\ \mathbb{E}[X^{\mathtt{P}_2}] \end{bmatrix}_{t|t-1},$

   $\tilde{\boldsymbol{\Sigma}}_{t|t-1} = \tilde{\boldsymbol{\Sigma}}_{1,t|t-1} + \begin{bmatrix} \mathbf{0}_{n_3\times n_3} & \mathrm{cov}(\boldsymbol{X}^{\mathtt{S}_3}, X^{\mathtt{P}_2}) \\ \mathrm{cov}(X^{\mathtt{P}_2}, \boldsymbol{X}^{\mathtt{S}_3}) & \mathrm{var}(X^{\mathtt{P}_2}) \end{bmatrix}_{t|t-1},$

where in Step 1 we explicitly compute the expected value $\mathbb{E}[X^{\mathtt{P}_1}]$ and the covariance matrix $\mathrm{cov}(\mathbf{X}, X^{\mathtt{P}_1})$ associated with the first product term $x_{t-1}^{\mathtt{P}_1}$, thereby computing the augmented mean vector $\tilde{\boldsymbol{\mu}}_{1,t-1|t-1}$ and covariance matrix $\tilde{\boldsymbol{\Sigma}}_{1,t-1|t-1}$, in Step 2 we carry out the prediction step using $\tilde{\boldsymbol{\mu}}_{1,t-1|t-1}$, $\tilde{\boldsymbol{\Sigma}}_{1,t-1|t-1}$, and the matrices $\mathbf{A}$ and $\mathbf{Q}$, and finally in Step 3 we compute the moments for the second product term $x_t^{\mathtt{P}_2}$ to obtain the predicted mean vector $\tilde{\boldsymbol{\mu}}_{t|t-1}$ and the covariance matrix $\tilde{\boldsymbol{\Sigma}}_{t|t-1}$. Using $\tilde{\boldsymbol{\mu}}_{t|t-1}$, $\tilde{\boldsymbol{\Sigma}}_{t|t-1}$, and Equation 11, the update step of the Kalman filter is performed. The complete model matrices $\mathbf{A}$, $\mathbf{C}$, $\mathbf{Q}$ and $\mathbf{R}$ required for state estimation using the Kalman filter

are described in Appendix B. The vector of unknown parameters which need to be estimated using an optimization algorithm [26, 42, 49] is given by

$$\boldsymbol{\theta} = [\sigma_w^{\mathtt{LT}} \ \sigma_w^{\mathtt{AR}} \ \sigma_w^{\mathtt{TP}} \ \sigma_v]^\intercal,$$

where $\sigma_w^{\mathtt{LT}}$ is the standard deviation of the local trend, $\sigma_w^{\mathtt{AR}}$ is the standard deviation of the `AR` process, $\sigma_w^{\mathtt{TP}}$ is the standard deviation of the local trend (`TP`) in `TM`, and σ_v is the standard deviation for the observation error. The initial parameter and the optimized parameter values using Newton-Raphson [42] optimization technique by maximizing the joint log-likelihood [27] are

$$\begin{aligned} \boldsymbol{\theta}_0 &= [10^{-6} \ 0.1 \ 10^{-6} \ 1]^\intercal, \\ \boldsymbol{\theta}^* &= [2.16 \times 10^{-6} \ 0.092 \ 6.5 \times 10^{-7} \ 0.054]^\intercal. \end{aligned}$$

4.2.3 State estimation

Figure 3 shows the observed flow rate data and the estimated values $\boldsymbol{\mu}_{t|t}$ along with its uncertainty bound $\boldsymbol{\mu}_{t|t} \pm \boldsymbol{\sigma}_{t|t}$ obtained using the Kalman filter for the training set and the test set. The grey

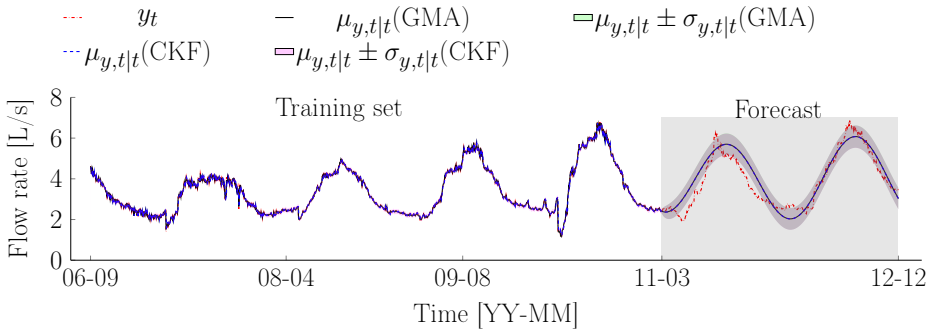

Figure 3: Estimated values of flow rate data using the GMA and the CKF

region shows the forecast period. Figure 4 shows the hidden state estimation of the flow rate data. The test set MSE and LL values obtained using the CKF and the GMA are $\{0.28, -541.5\}$ and $\{0.28, -541.6\}$ respectively. These results show that the proposed methodology has the same predictive capacity as that of the CKF and also provide interpretable sub-components of the time-varying amplitude hidden state.

4.3 Case Study 3: Double Kernel Regression

4.3.1 Data description

This case study is conducted on the traffic-load data [15, 28, 45] recorded on the Tamar bridge in UK. Correctly modeling traffic data is required for removing its effect on structural responses. The data ranges from September 01 to October 21, 2007. The raw data have 2409 data points with a uniform time steps of 30 minutes. The raw data show a constant baseline and two periodic components having a daily and a weekly periodicity. A multiplicative model is used to capture the dual periodicity using the product of the two periodic components. The data is divided into a *training set* (1649 points) and a *test set* (760). The entire dataset is shown in Figure 5.

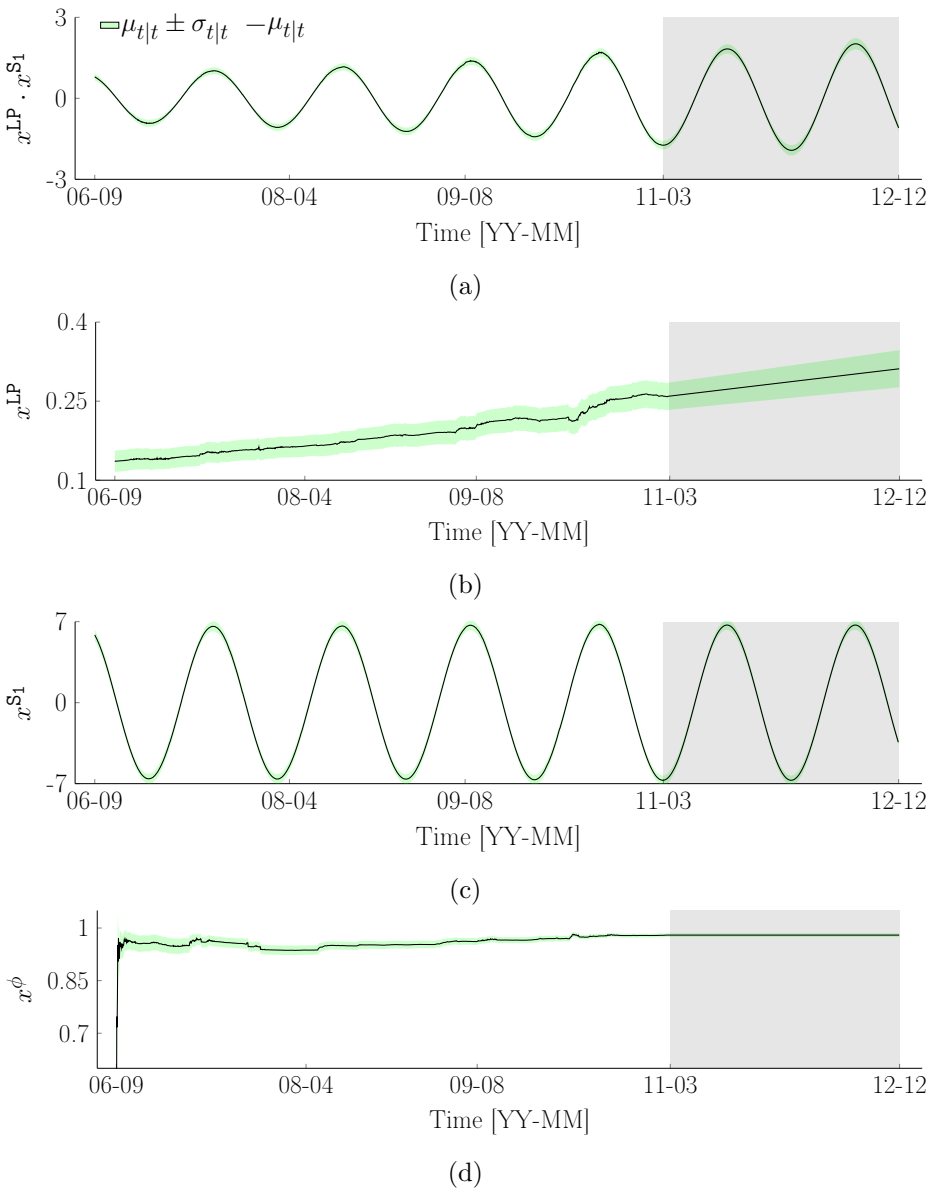

Figure 4: Illustration of the hidden state estimation of the flow rate data. Figure (a-c) represents the hidden states of `TM` component and Figure (d) represents x^{ϕ}.

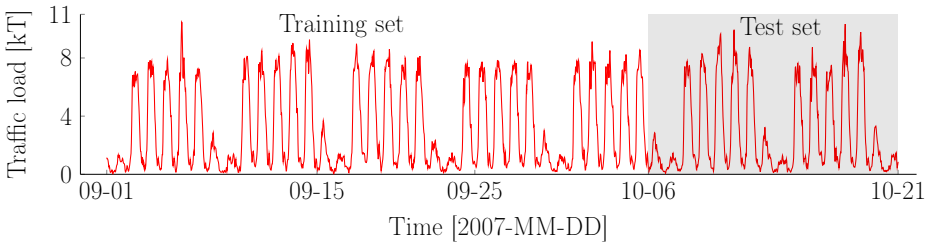

Figure 5: Traffic load on the bridge

4.3.2 Model Formulation

The components used for this model in the BDLM are a *local level* (LL) to model the constant baseline, two *kernel regression* (KR) components [45] each having 50 non-uniform and 30 uniform control-points to model periodic pattern with periodicity of 7 days and 1 day respectively, an online autoregressive component (OAR) and a new component called *double kernel regression* (DKR). DKR is used to model the product of two periodic pattern represented by the hidden states $x_0^{\mathtt{KR}_1}$ and $x_0^{\mathtt{KR}_2}$. The KR component for modeling 7 day periodic pattern requires more control points in the first two days due to the higher complexity in the sub-daily pattern compared to the rest of week. Note that increasing the number of control points can further improve accuracy at the cost of increasing computational cost. In this case, the transition model has two product terms represented by $x_{t-1}^{\mathtt{P}_1} = \{x_{t-1}^{\mathtt{AR}} \cdot x_{t-1}^{\phi}\}$ for OAR and $x_t^{\mathtt{P}_2} = \{x_0^{\mathtt{KR}_1} \cdot x_0^{\mathtt{KR}_2}\}_t$ for DKR. The vector of hidden states at time $t-1$ for all the components is defined as

$$\boldsymbol{x}_{t-1} = [\overbrace{\underbrace{x^{\mathtt{LL}}\ x^{\mathtt{AR}}\ x^{\phi}}_{\boldsymbol{x}^{\mathtt{S}_1}}\ x^{\mathtt{P}_1}\ \underbrace{x_0^{\mathtt{KR}_1} \ldots x_{50}^{\mathtt{KR}_1}\ x_0^{\mathtt{KR}_2} \ldots x_{30}^{\mathtt{KR}_2}}_{\boldsymbol{x}^{\mathtt{S}_2}}\ x^{\mathtt{P}_2}}^{\boldsymbol{x}^{\mathtt{S}_3}}]_{t-1}^{\intercal}. \tag{13}$$

The prediction step in the Kalman filter is carried out sequentially using the mean vector $\boldsymbol{\mu}_{t-1|t-1}$ and the covariance matrix $\boldsymbol{\Sigma}_{t-1|t-1}$ of $\boldsymbol{X}_{t-1}$ as shown in Section 4.2.3. The complete model matrices $\mathbf{A}$, $\mathbf{C}$, $\mathbf{Q}$ and $\mathbf{R}$ are described in Appendix B. The vector of unknown parameters is given by

$$\boldsymbol{\theta} = [\sigma_w^{\mathtt{LL}}\ \ell^{\mathtt{KR}_1}\ \ell^{\mathtt{KR}_2}\ \sigma_w^{\mathtt{AR}}\ \sigma_v]^{\intercal},$$

where $\sigma_w^{\mathtt{LL}}$ is the standard deviation of the local level, $\ell^{\mathtt{KR}_1}$ is the kernel length for the KR component with a period of 1 day, $\ell^{\mathtt{KR}_2}$ is the kernel length for the KR component with a period of 7 days, $\sigma_w^{\mathtt{AR}}$ is the standard deviation of the AR process, and σ_v is the standard deviation for the observation error. The initial parameter values and the optimized values using Newton-Raphson optimization technique are

$$\begin{aligned} \boldsymbol{\theta}_0 &= [10^{-6}\ 0.05\ 0.5\ 0.1\ 0.1]^{\intercal} \\ \boldsymbol{\theta}^{*} &= [1.01 \times 10^{-6}\ 0.359\ 0.24\ 0.275\ 1.93 \times 10^{-7}]^{\intercal}. \end{aligned}$$

4.3.3 State estimation

Figure 6 shows the observed traffic-load data and the estimated values $\boldsymbol{\mu}_{t|t}$ along with its uncertainty bound $\boldsymbol{\mu}_{t|t} \pm \boldsymbol{\sigma}_{t|t}$ obtained using Kalman filter for the training set and the test set using the GMA and the CKF. These results show that using the GMA in the BDLM has better predictive capacity than the CKF. The predictive capacity is also compared to the results presented in Nguyen et

al. [45] for the same dataset while using a single `KR` component with 101 control-points having a periodicity of 7 days. Table 2 presents the MSE, LL as well as the training time using `DKR` and `KR`. The results show that `DKR` has better predictive capacity than the `KR` component, fewer hidden states, and also has fewer parameters to optimize, which makes it computationally faster.

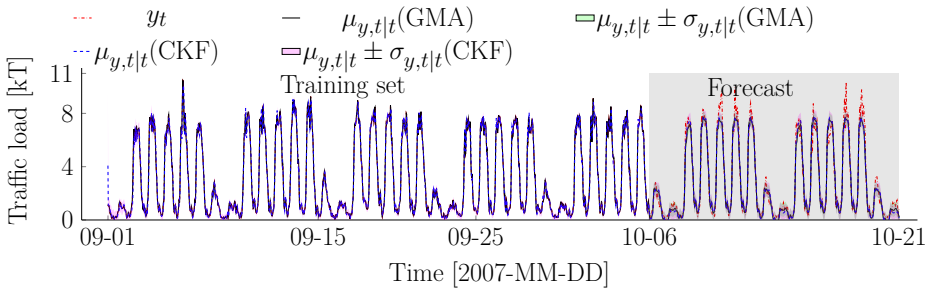

Figure 6: Estimated values of traffic load data using the GMA and the CKF

Table 2: Comparison of mean square error and log-likelihood estimates for `DKR` and `KR`

metric	MSE	Log-likelihood, $\sum_{t=1}^{T} \ln f(\boldsymbol{x}_t \vert y_{1:t})$
`DKR`	**0.30**	**−616.96**
`KR`	0.34	−656.30
`CKF`	0.32	−629.59

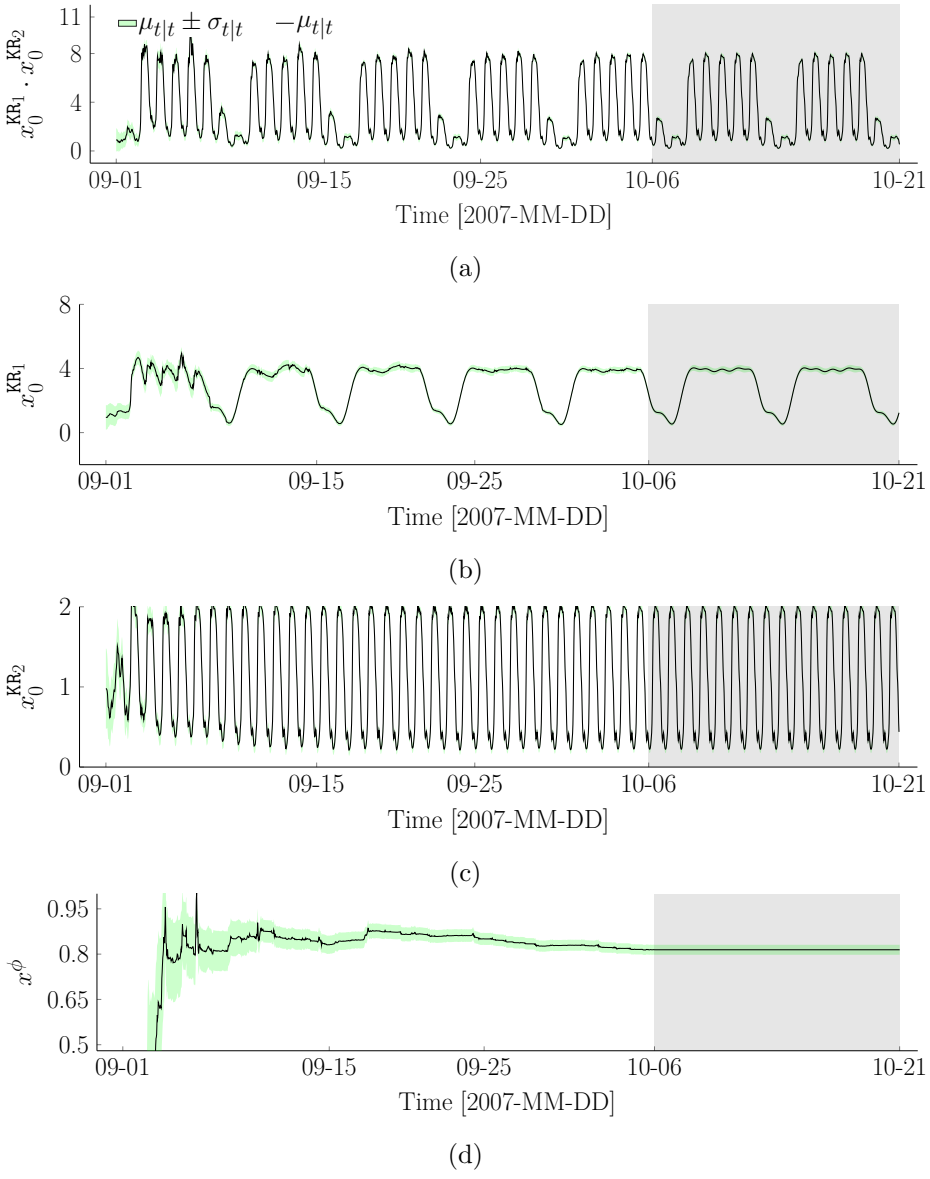

Figure 7: Illustration of the hidden state estimation of the traffic load data. Figure (a-c) represents the hidden states of `DKR` component and Figure (d) represents x^{ϕ}.

5 Conclusion

The Gaussian multiplicative approximation (GMA) method proposed in this paper is an analytical method for solving multiplicative state-space models. The method enables: (1) the analytical inference of the mean vector and the covariance matrix of the product of two hidden states in the transition and/or observation models using linear estimation theory, and (2) analytically tractable online estimation of model parameters as hidden states. The application of the GMA in the three case studies confirms that the method exceeds the performance of the existing nonlinear Kalman filter methods such as cubature Kalman filter in terms of both predictive capacity and computational complexity.

6 Acknowledgment

This project was funded by the Natural Sciences and Engineering Research Council of Canada (NSERC), Hydro-Québec (HQ), Hydro-Québec's Research Institute (IREQ), Institute For Data Valorization (IVADO). The authors thank Benjamin Miquel and Vincent Roy from Hydro-Québec for their help in the project.

References

[1] H Afshari, A Gadsden, and S Habibi. Gaussian filters for parameter and state estimation: A general review of theory and recent trends. *Signal Processing*, 135:218–238, 2017.

[2] D Agdas, A Rice, R Martinez, and R Lasa. Comparison of visual inspection and structural-health monitoring as bridge condition assessment methods. *Journal of Performance of Constructed Facilities*, 30(3):04015049, 2015.

[3] O Aguilar and M West. Analysis of hospital quality monitors using hierarchical time series models. In *Case Studies in Bayesian Statistics*, pages 287–302. Springer, 1999.

[4] Dawn An, Nam H Kim, and Joo-Ho Choi. Practical options for selecting data-driven or physics-based prognostics algorithms with reviews. *Reliability Engineering & System Safety*, 133:223–236, 2015.

[5] I Arasaratnam and S Haykin. Cubature Kalman filters. *IEEE Transactions on automatic control*, 54(6):1254–1269, 2009.

[6] I Arasaratnam, S Haykin, and R Hurd. Cubature Kalman filtering for continuous-discrete systems: theory and simulations. *IEEE Transactions on Signal Processing*, 58(10):4977–4993, 2010.

[7] A Assa and N Plataniotis. Adaptive Kalman filtering by covariance sampling. *IEEE Signal Processing Letters*, 24(9):1288–1292, 2017.

[8] J Barthorpe. *On model-and data-based approaches to structural health monitoring*. PhD thesis, University of Sheffield, 2010.

[9] G Bartram and S Mahadevan. Integration of heterogeneous information in SHM models. *Structural Control and Health Monitoring*, 21(3):403–422, 2014.

[10] P Batista, C Silvestre, and P Oliveira. Kalman and H infinity optimal filtering for a class of kinematic systems. *IFAC Proceedings Volumes*, 41(2):12528–12533, 2008.

[11] Y Bengio, P Simard, P Frasconi, et al. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[12] Michael George Bulmer. *Principles of statistics*. Courier Corporation, 1979.

[13] Eleni N Chatzi and Andrew W Smyth. The unscented kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing. *Structural Control and Health Monitoring: The Official Journal of the International Association for Structural Control and Monitoring and of the European Association for the Control of Structures*, 16(1):99–123, 2009.

[14] Siyu Chen, Chongshi Gu, Chaoning Lin, Yao Wang, and Mohammad Amin Hariri-Ardebili. Prediction, monitoring, and interpretation of dam leakage flow via adaptative kernel extreme learning machine. *Measurement*, 166:108161, 2020.

[15] EJ Cross, KY Koo, JMW Brownjohn, and K Worden. Long-term monitoring and data analysis of the tamar bridge. *Mechanical Systems and Signal Processing*, 35(1-2):16–34, 2013.

[16] B Dagum. Time series modeling and decomposition. *Statistica*, 70(4):433–457, 2010.

[17] G Di Girolamo, F Smarra, V Gattulli, F Potenza, F Graziosi, and A D'Innocenzo. Data-driven optimal predictive control of seismic induced vibrations in frame structures. *Structural Control and Health Monitoring*, 27(4):e2514, 2020.

[18] A Doucet, N De Freitas, K Murphy, and S Russell. Rao-Blackwellised particle filtering for dynamic Bayesian networks. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 176–183. Morgan Kaufmann Publishers Inc., 2000.

[19] Alireza Entezami, Hashem Shariatmadar, and Stefano Mariani. Early damage assessment in large-scale structures by innovative statistical pattern recognition methods based on time series modeling and novelty detection. *Advances in Engineering Software*, 150:102923, 2020.

[20] R Farrar and K Worden. *Structural Health Monitoring.: A Machine Learning Perspective*. John Wiley & Sons, 2012.

[21] S Gamse and M Oberguggenberger. Assessment of long-term coordinate time series using hydrostatic-season-time model for rock-fill embankment dam. *Structural Control and Health Monitoring*, 24(1):e1859, 2017.

[22] A Gelman, B Carlin, S Stern, B Dunson, A Vehtari, and B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.

[23] Z Ghahramani and E Hinton. Parameter estimation for linear dynamical systems. Technical report, Technical Report CRG-TR-96-2, University of Toronto, Dept. of Computer Science, 1996.

[24] A Ghaleb, A Zainal, A Rassam, and A Abraham. Improved vehicle positioning algorithm using enhanced innovation-based adaptive Kalman filter. *Pervasive and Mobile Computing*, 40:139–155, 2017.

[25] I Goodfellow, Y Bengio, and A Courville. *Deep learning*. MIT press, 2016.

[26] J-A Goulet. Bayesian dynamic linear models for structural health monitoring. *Structural Control and Health Monitoring*, 24(12):e2035, 2017.

[27] James-A Goulet. *Probabilistic machine learning for civil engineers*. MIT Press, 2020.

[28] James-A Goulet and Ki Koo. Empirical validation of bayesian dynamic linear models in the context of structural health monitoring. *Journal of Bridge Engineering*, 23(2):05017017, 2018.

[29] Mohammad Amin Hariri-Ardebili and Farhad Pourkamali-Anaraki. Simplified reliability analysis of multi hazard risk in gravity dams via machine learning techniques. *Archives of Civil and Mechanical Engineering*, 18(2):592–610, 2018.

[30] Mohammad Amin Hariri-Ardebili and Farhad Pourkamali-Anaraki. Support vector machine based reliability analysis of concrete dams. *Soil Dynamics and Earthquake Engineering*, 104:276–295, 2018.

[31] D J Henderson and C F Parmeter. *Applied nonparametric econometrics*. Cambridge University Press, 2015.

[32] S Hochreiter and J Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[33] Peter Hubertus Maria Janssen and Petre Stoica. On the expectation of the product of four matrix-valued gaussian random variables. *IEEE Transactions on Automatic Control*, 33(9):867–870, 1988.

[34] Andrew H Jazwinski. *Stochastic processes and filtering theory*. Courier Corporation, 2007.

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

[36] Junhee Kim and Jerome P Lynch. Subspace system identification of support excited structures,Äîpart ii: gray-box interpretations and damage detection. *Earthquake engineering & structural dynamics*, 41(15):2253–2271, 2012.

[37] P Léger and M Leclerc. Hydrostatic, temperature, time-displacement model for concrete dams. *Journal of engineering mechanics*, 133(3):267–277, 2007.

[38] L Ljung. Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems. *IEEE Transactions on Automatic Control*, 24(1):36–50, 1979.

[39] J Mata. Interpretation of concrete dam behaviour with artificial neural network and multiple linear regression models. *Engineering Structures*, 33(3):903–910, 2011.

[40] R Mehra. On the identification of variances and adaptive Kalman filtering. *IEEE Transactions on automatic control*, 15(2):175–184, 1970.

[41] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[42] P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[43] Satish Nagarajaiah and Yongchao Yang. Modeling and harnessing sparse and low-rank data structure: a new paradigm for structural dynamics, identification, damage detection, and health monitoring. *Structural Control and Health Monitoring*, 24(1):e1851, 2017.

[44] M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

[45] H Nguyen, I Gaudot, S Khazaeli, and J-A. Goulet. A kernel-based method for modeling non-harmonic periodic phenomena in Bayesian dynamic linear models. *Frontiers in Built Environment*, 5:8, 2019.

[46] H Nguyen and J-A Goulet. Anomaly detection with the switching Kalman filter for structural health monitoring. *Structural Control and Health Monitoring*, 25(4):e2136, 2018.

[47] H Nguyen and J-A. Goulet. Structural health monitoring with dependence on non-harmonic periodic hidden covariates. *Engineering Structures*, 166:187–194, 2018.

[48] H Nguyen and J-A Goulet. Real-time anomaly detection with Bayesian dynamic linear models. *Structural Control and Health Monitoring*, page e2404, 2019.

[49] Luong Ha Nguyen, Ianis Gaudot, and James-A Goulet. Uncertainty quantification for model parameters and hidden state variables in bayesian dynamic linear models. *Structural Control and Health Monitoring*, 26(3):e2309, 2019.

[50] LA Overbey and MD Todd. Analysis of local state space models for feature extraction in structural health monitoring. *Structural Health Monitoring*, 6(2):145–172, 2007.

[51] J Pathak, R Bharali, B Deka, S Pathak, I Ahmed, HL Dominik, and A Meslem. Building classification scheme and vulnerability model for the city of Guwahati, Assam. *EQRisk project report, Report*, (15-014), 2015.

[52] H Poveda, E Grivef, G Ferré, and N Christov. Kalman vs H infinity filter in terms of convergence and accuracy: Application to carrier frequency offset estimation. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 121–125. IEEE, 2012.

[53] Kiyoumars Roushangar, Saeede Garekhani, and Farhad Alizadeh. Forecasting daily seepage discharge of an earth dam using wavelet–mutual information–gaussian process regression approaches. *Geotechnical and Geological Engineering*, 34(5):1313–1326, 2016.

[54] J Russell and P Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[55] P Sage and W Husa. Algorithms for sequential adaptive estimation of prior statistics. In *1969 IEEE Symposium on Adaptive Processes (8th) Decision and Control*, pages 61–61. IEEE, 1969.

[56] H Sak, A Senior, and F Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Fifteenth annual conference of the international speech communication association*, 2014.

[57] Fernando Salazar, Rafael Morán, Miguel Á Toledo, and Eugenio Oñate. Data-based models for the prediction of dam behaviour: a review and some methodological considerations. *Archives of computational methods in engineering*, 24(1):1–21, 2017.

[58] S Sankararaman and S Mahadevan. Bayesian methodology for diagnosis uncertainty quantification and health monitoring. *Structural Control and Health Monitoring*, 20(1):88–106, 2013.

[59] Y Shi, C Han, and Y Liang. Adaptive UKF for target tracking with unknown process noise statistics. In *2009 12th International Conference on Information Fusion*, pages 1815–1820. IEEE, 2009.

[60] Dan Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[61] L Sobolev. Formulas of mechanical cubature on the surface of a sphere. *Sibirsk. Math*, 3:769–796, 1962.

[62] H Sohn, R Farrar, F Hunter, and K Worden. Structural health monitoring using statistical pattern recognition techniques. *Journal of dynamic systems, measurement, and control*, 123(4):706–711, 2001.

[63] Iickho Song and Seungwon Lee. Explicit formulae for product moments of multivariate gaussian random variables. *Statistics & Probability Letters*, 100:27–34, 2015.

[64] B Stojanovic, Milovan Milivojevic, M Ivanovic, Nikola Milivojevic, and Dejan Divac. Adaptive system for dam behavior modeling based on linear regression and genetic algorithms. *Advances in Engineering Software*, 65:182–190, 2013.

[65] M Tatin, M Briffaut, F Dufour, A Simon, and J-P Fabre. Thermal displacements of concrete dams: Accounting for water temperature in statistical models. *Engineering Structures*, 91:26–39, 2015.

[66] Christophe Vignat. A generalized isserlis theorem for location mixtures of gaussian random vectors. *Statistics & probability letters*, 82(1):67–71, 2012.

[67] 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)*, pages 153–158. Ieee, 2000.

[68] H Wang, Z Deng, B Feng, H Ma, and Y Xia. An adaptive Kalman filter estimating process noise covariance. *Neurocomputing*, 223:12–17, 2017.

[69] J Wang. Stochastic Modeling for Real-Time Kinematic GPS/GLONASS Positioning. *Navigation*, 46(4):297–305, 1999.

[70] Teng Wang, Daniele Perissin, Fabio Rocca, and Ming-Sheng Liao. Three gorges dam stability monitoring with time-series insar image analysis. *Science China Earth Sciences*, 54(5):720–732, 2011.

[71] Mike West and Jeff Harrison. *Bayesian forecasting and dynamic models*. Springer Science & Business Media, 2006.

[72] K Worden and G Manson. The application of machine learning to structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1851):515–537, 2006.

[73] Gang Yan, Hao Sun, and Oral Büyüköztürk. Impact load identification for composite structures using bayesian regularization and unscented kalman filter. *Structural Control and Health Monitoring*, 24(5):e1910, 2017.

[74] F Yu, Q Sun, C Lv, Y Ben, and Y Fu. A SLAM algorithm based on adaptive cubature Kalman filter. *Mathematical Problems in Engineering*, 2014, 2014.

[75] Yong Zeng and Shu Wu. *State-space models: Applications in economics and finance*, volume 1. Springer, 2013.

[76] H Zheng and A Mita. Damage indicator defined as the distance between ARMA models for structural health monitoring. *Structural Control and Health Monitoring: The Official Journal of the International Association for Structural Control and Monitoring and of the European Association for the Control of Structures*, 15(7):992–1005, 2008.

Appendix A

The GMA Equations using Gaussian moment generating function

Let $\boldsymbol{X} = [X_1 \dots X_p]^\intercal$ be a vector of Gaussian random variables, $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}$ is the mean vector, $\boldsymbol{\Sigma}$ is the covariance matrix, and $\mathbf{t} = [t_1 \dots t_p]^\intercal \in \mathbb{R}^p$, then the following Equation [33,63] is held which analytically computes the multivariate moments encountered in the nonlinear Kalman filter given by

$$\mathbb{E}[X_1^{k_1} \dots X_p^{k_p}] = \frac{\partial^k}{\partial t^{k_1} \dots \partial t^{k_p}} \exp\Big(\sum_{i=1}^{p} t_i \mu_i + \frac{1}{2} \sum_{i,j=1}^{p} t_i t_j \text{cov}(X_i, X_j)\Big)\Big|_{t_1=\dots=t_p=0} \tag{14}$$

where k_i's are non-negative integers and $k = \sum_{i=1}^{p} k_i$. This Equation is derived from the moment generating function of multivariate Gaussian;

$$\begin{aligned} \frac{\partial^k}{\partial t_1^{k_1} \dots \partial t_p^{k_p}} M_{\boldsymbol{x}}(\mathbf{t}^\intercal) &= \frac{\partial^k}{\partial t_1^{k_1} \dots \partial t_p^{k_p}} \mathbb{E}(\exp\left(\sum_{i=1}^{p} t_i X_i\right)) \\ &= \mathbb{E}\big(\frac{\partial^k}{\partial t^{k_1} \dots \partial t^{k_p}} \exp\left(\sum_{i=1}^{p} t_i X_i\right)\big) \\ &= \mathbb{E}[X_1^{k_1} \dots X_p^{k_p} \exp\left(\sum_{i=1}^{p} t_i X_i\right)] \end{aligned} \tag{15}$$

Setting $\mathbf{t} = [0 \dots 0]^\intercal$, we obtain

$$\mathbb{E}[X_1^{k_1} \dots X_p^{k_p}] = \frac{\partial^k}{\partial t^{k_1} \dots \partial t^{k_p}} M_{\boldsymbol{x}}(\mathbf{t}^\intercal)$$

Given Gaussian random variables, $M_{\boldsymbol{x}}(\mathbf{t}^\intercal) = \mathbb{E}[e^{\mathbf{t}^\intercal \boldsymbol{x}}] = e^{\mathbf{t}^\intercal \boldsymbol{\mu} + \frac{1}{2}\mathbf{t}^\intercal \boldsymbol{\Sigma} \mathbf{t}}$, the following equations to evaluate product terms can be directly obtained from Equation 14.

$$\begin{aligned} \mathbb{E}[X_1 X_2] &= \mu_1\mu_2 + \text{cov}(X_1, X_2), \\ \mathbb{E}[X_1 X_2 X_3] &= \text{cov}(X_1, X_2)\mu_3 + \text{cov}(X_1, X_3)\mu_2 + \text{cov}(X_2, X_3)\mu_1 + \mu_1\mu_2\mu_3, \\ \mathbb{E}[X_1 X_2 X_3 X_4] &= \text{cov}(X_1 X_2)\big(\text{cov}(X_3, X_4) + \mu_3\mu_4\big) + \text{cov}(X_1 X_3)\big(\text{cov}(X_2, X_4) + \mu_2\mu_4\big) + \\ &\quad \text{cov}(X_2 X_3)\big(\text{cov}(X_1, X_4) + \mu_1\mu_4\big) + \text{cov}(X_1, X_4)\mu_2\mu_3 + \text{cov}(X_2, X_4)\mu_1\mu_3 \\ &\quad + \text{cov}(X_3, X_4)\mu_1\mu_2 + \mu_1\mu_2\mu_3\mu_4. \end{aligned}$$

The GMA Equations using 2^{nd} order Taylor series expansion

Let us consider the function $h(\cdot)$ in two variables x_1 and x_2, where $h(x_1, x_2) = x_1 x_2$ that represents the product of two random variables. Using 2^{nd} order Taylor series expansion, we express $h(x_1, x_2)$ as

$$\begin{aligned} h(x_1, x_2) &\approx h(\mu_1, \mu_2) + \frac{\partial h}{\partial x_1}\Big|_{\mu_1,\mu_2}(x_1 - \mu_1) + \frac{\partial h}{\partial x_2}\Big|_{\mu_1,\mu_2}(x_2 - \mu_2) + 0.5 \cdot \frac{\partial^2 h}{\partial x_1^2}\Big|_{\mu_1,\mu_2}(x_1 - \mu_1)^2 \\ &+ 0.5 \cdot \frac{\partial^2 h}{\partial x_2^2}\Big|_{\mu_1,\mu_2}(x_2 - \mu_2)^2 + \cdot \frac{\partial^2 h}{\partial x_1 \partial x_2}\Big|_{\mu_1,\mu_2}(x_1 - \mu_1)(x_2 - \mu_2), \\ &\approx \mu_1\mu_2 + \mu_2(x_1 - \mu_1) + \mu_1(x_2 - \mu_2) + (x_1 - \mu_1)(x_2 - \mu_2), \end{aligned} \tag{16}$$

where $h(\mu_1, \mu_2) = \mu_1\mu_2$, $\frac{\partial h}{\partial x_1}\big|_{\mu_1,\mu_2} = \mu_2$, $\frac{\partial h}{\partial x_2}\big|_{\mu_1,\mu_2} = \mu_1$, and $\frac{\partial^2 h}{\partial x_1 \partial x_2}\big|_{\mu_1,\mu_2} = 1$. Using Equation 16, the expected value $\mathbb{E}[X_1X_2]$ is

$$\begin{aligned}\mathbb{E}[X_1X_2] &= \mu_1\mu_2 + \mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)],\\ &= \mu_1\mu_2 + \text{cov}(X_1, X_2),\end{aligned} \tag{17}$$

where using the properties of random variables $\mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)] = \text{cov}(X_1, X_2)$. The variance term $\text{var}(X_1X_2)$ is given by

$$\begin{aligned}\text{var}(X_1X_2) &= \text{var}\Big(\mu_1\mu_2 + \mu_2(X_1-\mu_1) + \mu_1(X_2-\mu_2) + (X_1-\mu_1)(X_2-\mu_2)\Big),\\ &= \text{var}\Big(\mu_2(X_1-\mu_1)\Big) + \text{var}\Big(\mu_1(X_2-\mu_2)\Big) + \text{var}\Big((X_1-\mu_1)(X_2-\mu_2)\Big)\\ &+ 2\text{cov}\Big(\mu_2(X_1-\mu_1), \mu_1(X_2-\mu_2)\Big) + 2\text{cov}\Big(\mu_2(X_1-\mu_1), (X_1-\mu_1)(X_2-\mu_2)\Big)\\ &+ 2\text{cov}\Big((x_1-\mu_1)(X_2-\mu_2), \mu_1(X_2-\mu_2)\Big),\\ &= \mu_2^2\sigma_1^2 + \mu_1^2\sigma_2^2 + \sigma_1^2\sigma_2^2 + \text{cov}(X_1, X_2)^2 + 2\mu_1\mu_2\text{cov}(X_1, X_2),\end{aligned} \tag{18}$$

where the terms in Equation 18 are evaluated as follows

$$\begin{aligned}\text{var}\Big((X_1-\mu_1)(X_2-\mu_2)\Big) &= \mathbb{E}[(X_1-\mu_1)^2(X_2-\mu_2)^2] - \mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)]^2 &(19)\\ &= \sigma_1^2\sigma_2^2 + 2\text{cov}(X_1, X_2)^2 - \text{cov}(X_1, X_2)^2,\\ &= \sigma_1^2\sigma_2^2 + \text{cov}(X_1, X_2)^2, &(20)\end{aligned}$$

where using Isserlis theorem [66], $\mathbb{E}[(X_1-\mu_1)^2(X_2-\mu_2)^2] = \sigma_1^2\sigma_2^2 + 2\text{cov}(X_1, X_2)^2$. Similarly, the covariance terms in Equation 18 are given by

$$\begin{aligned}2\text{cov}\Big(\mu_2(X_1-\mu_1), \mu_1(X_2-\mu_2)\Big) &= 2\mu_1\mu_2\text{cov}(X_1, X_2), &(21)\\ 2\text{cov}\Big(\mu_2(X_1-\mu_1), (X_1-\mu_1)(X_2-\mu_2)\Big) &= 2\mu_2\Big(\mathbb{E}[(X_1-\mu_1)^2(X_2-\mu_2)]\\ &- \cancelto{0}{\mathbb{E}[X_1-\mu_1]}\mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)]\Big),\\ &= 0, &(22)\\ 2\text{cov}\Big(\mu_1(X_2-\mu_2), (X_1-\mu_1)(X_2-\mu_2)\Big) &= 2\mu_1\Big(\mathbb{E}[(X_2-\mu_2)^2(X_1-\mu_1)]\\ &- \cancelto{0}{\mathbb{E}[X_2-\mu_2]}\mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)]\Big),\\ &= 0, &(23)\end{aligned}$$

where using Isserlis theorem, $\mathbb{E}[(X_1-\mu_1)^2(X_2-\mu_2)] = \mathbb{E}[(X_2-\mu_2)^2(X_1-\mu_1)] = 0$ as the expected values of the product of odd powers of Gaussian random variables of zero-mean values are 0. The covariance between X_3 and X_1X_2 is

$$\begin{aligned}\text{cov}(X_3, X_1X_2) &= \text{cov}\Big(X_3, \mu_1\mu_2 + \mu_2(X_1-\mu_1) + \mu_1(X_2-\mu_2) + (X_1-\mu_1)(X_2-\mu_2)\Big),\\ &= \text{cov}\Big(X_3, \mu_2(X_1-\mu_1)\Big) + \text{cov}\Big(X_3, \mu_1(X_2-\mu_2)\Big) + \text{cov}\Big(X_3, (X_1-\mu_1)(X_2-\mu_2)\Big),\\ &= \mu_2\text{cov}(X_3, X_1) + \mu_1\text{cov}(X_3, X_2),\end{aligned}$$

where similar to Equations 22 and 23, the term $\text{cov}\Big(X_3, (X_1-\mu_1)(X_2-\mu_2)\Big) = 0$ and evaluated as follows

$$\begin{aligned}
\text{cov}\Big(X_3, (X_1-\mu_1)(X_2-\mu_2)\Big) &= \mathbb{E}[X_3(X_1-\mu_1)(X_2-\mu_2)] - \mathbb{E}[X_3]\mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)], \\
&= \mathbb{E}[(X_3-\mu_3+\mu_3)(X_1-\mu_1)(X_2-\mu_2)] - \mu_3\text{cov}(X_1,X_2), \\
&= \mathbb{E}[(X_3-\mu_3)(X_1-\mu_1)(X_2-\mu_2)] + \mathbb{E}[(\mu_3)(X_1-\mu_1)(X_2-\mu_2)] \\
&- \mu_3\text{cov}(X_1,X_2), \\
&= 0 + \mu_3\text{cov}(X_1,X_2) - \mu_3\text{cov}(X_1,X_2), \\
&= 0.
\end{aligned}$$

The covariance between the product terms X_1X_2 and X_3X_4 is given by

$$\begin{aligned}
\text{cov}(X_1X_2, X_3X_4) &= \text{cov}\Big(\mu_1\mu_2 + \mu_2(X_1-\mu_1) + \mu_1(X_2-\mu_2) + (X_1-\mu_1)(X_2-\mu_2), \\
&\quad \mu_3\mu_4 + \mu_4(X_3-\mu_3) + \mu_3(X_4-\mu_4) + (X_3-\mu_3)(X_4-\mu_4)\Big), \\
&= \text{cov}\Big(\mu_2(X_1-\mu_1), \mu_4(X_3-\mu_3)\Big) + \text{cov}\Big(\mu_2(X_1-\mu_1), \mu_3(X_4-\mu_4\Big) \\
&+ \text{cov}\Big(\mu_2(X_1-\mu_1), (X_3-\mu_3)(X_4-\mu_4)\Big) \\
&+ \text{cov}\Big(\mu_1(X_2-\mu_2), \mu_4(X_3-\mu_3)\Big) + \text{cov}\Big(\mu_1(X_2-\mu_2), \mu_3(X_4-\mu_4)\Big) \\
&+ \text{cov}\Big(\mu_1(X_2-\mu_2), (X_3-\mu_3)(X_4-\mu_4)\Big) \\
&+ \text{cov}\Big((X_1-\mu_1)(X_2-\mu_2), \mu_4(X_3-\mu_3)\Big) + \text{cov}\Big((X_1-\mu_1)(X_2-\mu_2), \mu_3(X_4-\mu_4)\Big) \\
&+ \text{cov}\Big((X_1-\mu_1)(X_2-\mu_2), (X_3-\mu_3)(X_4-\mu_4)\Big), \\
&= \text{cov}(X_1,X_3)\text{cov}(X_2,X_4) + \text{cov}(X_1,X_4)\text{cov}(X_2,X_3) + \mu_2\mu_4\text{cov}(X_1,X_3) \\
&+ \mu_2\mu_3\text{cov}(X_1,X_4) + \mu_1\mu_4\text{cov}(X_2,X_3) + \mu_1\mu_3\text{cov}(X_2,X_4), \qquad (24)
\end{aligned}$$

where the terms in Equation 24 are evaluated as follows

$$\begin{aligned}
\text{cov}\Big(\mu_2(X_1-\mu_1), \mu_4(X_3-\mu_3)\Big) &= \mu_2\mu_4\text{cov}(X_1,X_3), \\
\text{cov}\Big(\mu_2(X_1-\mu_1), \mu_3(X_4-\mu_4\Big) &= \mu_2\mu_3\text{cov}(X_1,X_4), \\
\text{cov}\Big(\mu_2(X_1-\mu_1), (X_3-\mu_3)(X_4-\mu_4)\Big) &= \mu_2\text{cov}\Big((X_1-\mu_1), (X_3-\mu_3)(X_4-\mu_4)\Big), \\
&= \mu_2\mathbb{E}[(X_1-\mu_1)(X_3-\mu_3)(X_4-\mu_4)] \qquad (25) \\
&- \mathbb{E}[(X_1-\mu_1)]\mathbb{E}[(X_3-\mu_3)(X_4-\mu_4)], \\
&= 0,
\end{aligned}$$

$$
\begin{aligned}
\operatorname{cov}\Big(\mu_1(X_2-\mu_2),\mu_4(X_3-\mu_3)\Big) &= \mu_1\mu_4\operatorname{cov}(X_2,X_3),\\
\operatorname{cov}\Big(\mu_1(X_2-\mu_2),\mu_3(X_4-\mu_4)\Big) &= \mu_1\mu_3\operatorname{cov}(X_2,X_4),\\
\operatorname{cov}\Big(\mu_1(X_2-\mu_2),(X_3-\mu_3)(X_4-\mu_4)\Big) &= 0,\\
\operatorname{cov}\Big((X_1-\mu_1)(X_2-\mu_2),\mu_4(X_3-\mu_3)\Big) &= 0,\\
\operatorname{cov}\Big((X_1-\mu_1)(X_2-\mu_2),\mu_3(X_4-\mu_4)\Big) &= 0,\\
\operatorname{cov}\Big((X_1-\mu_1)(X_2-\mu_2),(X_3-\mu_3)(X_4-\mu_4)\Big) &= \mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)(X_3-\mu_3)(X_4-\mu_4)]\\
&- \mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)]\mathbb{E}[(X_3-\mu_3)(X_4-\mu_4)],\\
&= \operatorname{cov}(X_1,X_2)\operatorname{cov}(X_3,X_4)+\operatorname{cov}(X_1,X_3)\operatorname{cov}(X_2,X_4)\\
&+ \operatorname{cov}(X_1,X_4)\operatorname{cov}(X_2,X_3)-\operatorname{cov}(X_1,X_2)\operatorname{cov}(X_3,X_4),\\
&= \operatorname{cov}(X_1,X_3)\operatorname{cov}(X_2,X_4)+\operatorname{cov}(X_1,X_4)\operatorname{cov}(X_2,X_3),
\end{aligned}
$$

where using Isserlis theorem [66], the expected value of the product of centered-Gaussian random variables $X_i, X_j, X_k, \text{and} X_n$ is $\mathbb{E}[X_iX_jX_kX_n] = \sigma_{ij}\sigma_{kn}+\sigma_{ik}\sigma_{jn}+\sigma_{in}\sigma_{jk}$, considering $\sigma_{ij}=\operatorname{cov}(X_i,X_j)$.

Appendix B

This appendix presents the model matrices, $\mathbf{A}$, $\mathbf{C}$, $\mathbf{Q}$ and $\mathbf{R}$ used in the BDLM for case study 2 and 3 in the Section 4.2 and 4.3 as follows:

Trend Multiplicative

$$
\begin{aligned}
\mathbf{A} &= \text{blockdiag}\left(\begin{bmatrix}1 & \Delta t\\ 0 & 1\end{bmatrix},\begin{bmatrix}\cos\omega & \sin\omega\\ -\sin\omega & \cos\omega\end{bmatrix},\begin{bmatrix}0&0&1\\0&1&0\\0&0&0\end{bmatrix},\begin{bmatrix}1&\Delta t&0\\0&1&0\\0&0&0\end{bmatrix}\right),\\
\mathbf{C} &= [1\,0\,0\,0\,1\,0\,0\,0\,0\,1],\\
\mathbf{R} &= (\sigma_v)^2,\\
\mathbf{Q} &= \text{blockdiag}\left((\sigma_w^{\texttt{LT}})^2\begin{bmatrix}\frac{\Delta t^3}{3} & \frac{\Delta t^2}{2}\\ \frac{\Delta t^2}{2} & \Delta t\end{bmatrix},(\sigma_w^{\texttt{S}})^2\begin{bmatrix}1&0\\0&1\end{bmatrix},(\sigma_w^{\texttt{AR}})^2\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix},(\sigma_w^{\texttt{TP}})^2\begin{bmatrix}\frac{\Delta t^3}{3}&\frac{\Delta t^2}{2}&0\\ \frac{\Delta t^2}{2}&\Delta t&0\\0&0&0\end{bmatrix}\right).
\end{aligned}
$$

where frequency $\omega = \dfrac{2\pi\Delta t}{p}$ with $p = 365.24$ days and $\Delta t = 1$ day. The prediction and update steps in the Kalman filter are given by

$$
[\tilde{\boldsymbol{\mu}}_{t|t-1}, \tilde{\boldsymbol{\Sigma}}_{t|t-1}] = \text{Predict}\left(\boldsymbol{\mu}_{t-1|t-1}, \boldsymbol{\Sigma}_{t-1|t-1}, \mathbf{A}, \mathbf{C}, \mathbf{Q}, \mathbf{R}\right),
$$

$$
[\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t}] = \text{Update}\left(\tilde{\boldsymbol{\mu}}_{t|t-1}, \tilde{\boldsymbol{\Sigma}}_{t|t-1}, \mathbf{A}, \mathbf{C}, \mathbf{Q}, \mathbf{R}\right).
$$

Double Kernel Regression

$$
\begin{aligned}
\mathbf{A} &= \text{blockdiag}\left(1, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & \tilde{k}^{\mathtt{KR}_1}(t, \mathbf{t}^{\mathtt{KR}}) \\ \mathbf{0}_{50\times 1} & \mathbf{I}_{50\times 50} \end{bmatrix}, \begin{bmatrix} 0 & \tilde{k}^{\mathtt{KR}_2}(t, \mathbf{t}^{\mathtt{KR}}) \\ \mathbf{0}_{30\times 1} & \mathbf{I}_{30\times 30} \end{bmatrix}, 0\right), \\
\mathbf{C} &= [1\ 1\ 0\ 0\ \mathbf{0}_{1\times 51}\ \mathbf{0}_{1\times 31}\ 1], \\
\mathbf{R} &= (\sigma_v)^2, \\
\mathbf{Q} &= \text{blockdiag}\left((\sigma_w^{\mathtt{LL}})^2, (\sigma_w^{\mathtt{AR}})^2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{0}_{51\times 51}, \mathbf{0}_{31\times 31}, 0\right).
\end{aligned}
$$

Appendix C

Computational Complexity

Algorithm 1 Kalman filter algorithm with the GMA

Input: $\boldsymbol{\mu}_{t-1|t-1}$,$\boldsymbol{\Sigma}_{t-1|t-1}$
Output: $\boldsymbol{\mu}_{t|t}$,$\boldsymbol{\Sigma}_{t|t}$

1: $\boldsymbol{\mu}_{t|t-1} = \mathbf{A}\tilde{\boldsymbol{\mu}}_{t-1|t-1}$.
2: $\boldsymbol{\Sigma}_{t|t-1} = \mathbf{A}\tilde{\boldsymbol{\Sigma}}_{t-1|t-1}\mathbf{A}^\intercal + \mathbf{Q}$.
3: $\mathbf{K} = \boldsymbol{\Sigma}_{t-1|t-1}\mathbf{C}^\intercal(\mathbf{C}\boldsymbol{\Sigma}_{t-1|t-1}\mathbf{C}^\intercal + \mathbf{R})^{-1}$.
4: $r_t = y_t - \mathbf{C}\boldsymbol{\mu}_{t|t-1}$.
5: $\boldsymbol{\mu}_{t|t} = \boldsymbol{\mu}_{t|t-1} + \mathbf{K}r_t$.
6: $\boldsymbol{\Sigma}_{t|t} = (\mathbf{I} - \mathbf{KC})\boldsymbol{\Sigma}_{t|t-1}$.

Since the filtering method is recursive, it is enough to determine the computational complexity of a single time step going from $t-1$ to t to evaluate the total complexity of the algorithm. The computational complexity here refers to the time complexity of an algorithm which is denoted by the big $\boldsymbol{O}$ notation. The time complexity (or from here on complexity) of the matrix operations to be used in algorithm 1 are described as follows.

1. Matrix multiplication: The multiplication of two matrices of size $n \times n$ has a complexity of $\mathcal{O}(\mathbf{n^3})$. In general, matrix multiplication of two matrices of size $n \times m$ and $m \times p$ has a complexity of $\mathcal{O}(\mathbf{mnp})$.

2. Matrix addition: The addition of matrices of size $m \times n$ has a complexity of $\mathcal{O}(\mathbf{mn})$.

3. Algebraic operations: Algebraic operations are considered to have constant complexity or $\mathcal{O}(\mathbf{1})$ as these operations are unaffected by the size of state vector and will be performed in the same time.

Considering the big $\boldsymbol{O}$ notation for the various matrix operations and the size of the state vector to be n, the step-by-step complexity of algorithm 1 is presented.
Step 1: This step consists of multiplication of two matrices, $[\mathbf{A}]_{n\times n}$ and $[\tilde{\boldsymbol{\mu}}]_{n\times 1}$ with a complexity is $\mathcal{O}(\mathbf{n^2})$. The complexity of computing $\tilde{\boldsymbol{\mu}}$ is $\mathcal{O}(\mathbf{n})$. Since, the GMA equations are algebraic operations unaffected by the size of the state vector, these will have a complexity of $\mathcal{O}(\mathbf{1})$. Hence the total complexity in Step 1 is $\mathcal{O}(\mathbf{n^2 + n})$.

Step 2: This step consists of two matrix multiplication and one transposition operation for computing $[\mathbf{A}]_{n\times n}[\tilde{\mathbf{\Sigma}}]_{n\times n}[\mathbf{A}]^{\intercal}_{n\times n}$and one matrix addition. The complexity for this step is $\mathcal{O}(\mathbf{2n^3 + n^2})$.
Step 3: This step consists of computing the terms $[\mathbf{\Sigma}]_{n\times n}[\mathbf{C}^{\intercal}]_{n\times 1}$ having a complexity of $\mathcal{O}(\mathbf{n^2})$ and $[\mathbf{C}]_{1\times n}[\mathbf{\Sigma}]_{n\times n}[\mathbf{C}^{\intercal}]_{n\times 1}$ having a complexity of $\mathcal{O}(\mathbf{2n^2})$. Finally, the total complexity of calculating the Kalman gain is $\mathcal{O}(\mathbf{3n^2 + n})$.
Step 4: This step consists of a matrix multiplication of $[\mathbf{C}]_{1\times n}[\boldsymbol{\mu}]_{n\times 1}$ having a complexity of $\mathcal{O}(\mathbf{n})$.
Step 5: This step consists of a matrix addition of $[\boldsymbol{\mu}]_{n\times 1}$ and multiplication of a matrix by a scalar, $[\mathbf{K}]_{n\times 1}[r]_{1\times 1}$. The total complexity in this step is $\mathcal{O}(\mathbf{2n})$.
Step 6: The final step to compute $\mathbf{\Sigma}_{t|t}$ consists of two matrix multiplication and one matrix subtraction. The total complexity in this step is $\mathcal{O}(\mathbf{n^3 + 2n^2})$.
Hence, the total complexity of the Kalman filter algorithm using the GMA is of the order $\equiv \mathcal{O}(\mathbf{3n^3})$.