14th International Conference on Applications of Statistics and Probability in Civil Engineering, ICASP14
Dublin, Ireland, July 9-13, 2023

Probabilistic time series modelling using Bayesian neural networks

Van-Dai Vuong
Department of Civil, Geologic and Mining Engineering, Polytechnique Montréal, Canada

James.A- Goulet
Department of Civil, Geologic and Mining Engineering, Polytechnique Montréal, Canada

ABSTRACT: Interpretable models play a key role in structural health monitoring (SHM) systems dealing with time series data. The long short-term memory (LSTM) neural network architecture is widely used to make time series forecast. Key limitations of the LSTM are that it is a deterministic model which only provides point-estimates for predictions, and it is a black-box because its results are not interpretable. The Tractable Approximate Gaussian Inference (TAGI) method allows learning the network parameters analytically, and accounts for both the epistemic and aleatory uncertainties. In this paper, we first propose the Bayesian TAGI-LSTM framework which is capable of quantifying the parameter uncertainties as well as the uncertainties associated with predictions. We then couple TAGI-LSTM with Bayesian Linear Dynamic Models (BDLM) in a probabilistic manner to create a hybrid model which can provide interpretable results along with the prediction uncertainties. The experimental results from SHM case studies show that our method provides a similar predictive performance compared to standard BDLM models while being interpretable and not requiring feature engineering nor parameter optimization.

1. INTRODUCTION

Interpretable models play a key role in structural health monitoring systems (SHM) dealing with time series data. Interpretable models provide users with meaningful insights extracted from the raw data such as the level describing the irreversible structural behaviour, the trend describing its rate of change, and the seasonality for isolating the reversible effects caused by environmental factors. Bayesian Linear Dynamic Models (BDLM) (Goulet, 2017) is a special type of state-space models (SSMs); this probabilistic method provides interpretable results by decomposing raw data into reversible and irreversible effects. In BDLM, the harmonic and non-harmonic reversible effects can be modelled by periodic components for which the parameter estimation relies on optimization techniques. The key limitation preventing the scaling of BDLM to a large number of time series is that it requires feature engineering to define the dependencies between the model's components. Long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) is a widely used neural network architecture to model time series. LSTM is capable of modelling complex patterns such that it could replace the existing BDLM's periodic components, and automatically model the non-linear dependencies within and between time series. The key limitation of the existing LSTM neural networks is that they either rely on deterministic parameter estimation which fails to take into account the epistemic uncertainties and only provide point-estimates for predictions, or they rely on gradient-based optimization techniques that are incompatible with SSMs. As a result, LSTMs cannot be coupled analytically with

probabilistic time series models.
The objective of this paper is twofold; first, we develop the Bayesian TAGI-LSTM neural networks taking into consideration both the epistemic and aleatory uncertainties. The weight and bias parameters of TAGI-LSTM are estimated analytically using Bayes' theorem instead of the optimization-based backpropagation algorithm. Second, we couple TAGI-LSTM with BDLM in a probabilistic manner resulting in a novel hybrid model inheriting the best features from both methods. Our hybrid model provides interpretable results along with the prediction uncertainties while not requiring feature engineering nor parameter optimization.

The paper is organized as follow: Section 2 reviews the BDLM and TAGI methods as well as the LSTM architecture which are the foundations for this work. Section 3 introduces the methodologies for the TAGI-LSTM neural network and the hybrid model which couples TAGI-LSTM with BDLM. Section 4 compares these new models with BDLM on two SHM case studies.

2. RELATED WORK

In this section, we review BDLM, LSTM, and TAGI, which are the three methods that we build upon, along with a review of the applications of LSTM in SHM.

2.1. *Bayesian dynamic linear models (BDLM)*

BDLM (Goulet, 2017) is a class of SSMs where the transition and observation models are linear, and the hidden state vector is assembled from predefined sub-components. BDLM can provide interpretable results by decomposing data into the baseline hidden states describing the irreversible effect, and the periodic hidden states describing the reversible one. The advantage of BDLM is that it is a probabilistic method which can provide the uncertainties associated with the predictions as well as the uncertainties related to the unobserved hidden states. However, the key disadvantage preventing BLDM from scaling to analyze thousands of time series is that it requires feature engineering in order to define models. For example, the environmental variables such as water temperature and reservoir level typically have a delayed effect on dam responses such as displacements and crack openings. In order to take these delayed effects into account in BDLM, users need to specify the lag time between the environmental variable and the response, and manually define the dependencies between the model's components. Furthermore, the BDLM's parameters are obtained by optimization, and are considered as having fixed values which does not allow for adaptation to changing conditions.

2.2. *Long-short term memory neural network (LSTM)*

LSTM (Hochreiter and Schmidhuber, 1997) is a special type of recurrent neural networks (RNNs) (Rumelhart et al., 1986) designed to model a long-term memory. The hidden states $\boldsymbol{h}$ and cell states $\boldsymbol{c}$ act as two memories for LSTM. The hidden states store information about short-term dependencies, whereas the cell states encode long-term dependencies. Both long and short-term dependencies are valuable sources of information when making future predictions. LSTM uses a gating system in order to automatically select which information is stored in its memories. The gates, cell states and hidden states of a LSTM cell are defined by the following equations

$$\boldsymbol{f}_t = \sigma(\boldsymbol{W}^f \boldsymbol{x}_t + \boldsymbol{U}^f \boldsymbol{h}_{t-1} + \boldsymbol{b}^f), \tag{1a}$$
$$\boldsymbol{i}_t = \sigma(\boldsymbol{W}^i \boldsymbol{x}_t + \boldsymbol{U}^i \boldsymbol{h}_{t-1} + \boldsymbol{b}^i), \tag{1b}$$
$$\boldsymbol{o}_t = \sigma(\boldsymbol{W}^o \boldsymbol{x}_t + \boldsymbol{U}^o \boldsymbol{h}_{t-1} + \boldsymbol{b}^o), \tag{1c}$$
$$\tilde{\boldsymbol{c}}_t = \tanh(\boldsymbol{W}^c \boldsymbol{x}_t + \boldsymbol{U}^c \boldsymbol{h}_{t-1} + \boldsymbol{b}^c), \tag{1d}$$
$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t, \tag{1e}$$
$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t), \tag{1f}$$

where $\boldsymbol{x}$ is the covariate vector, $\boldsymbol{W}$ is the weight matrix for the input, $\boldsymbol{U}$ is the weight matrix for the hidden states, $\boldsymbol{b}$ is the bias vector, and the superscripts $\{f,i,o,c\}$ indicate the forget, input, output and cell gates, respectively; $\sigma(\cdot)$ and $\tanh(\cdot)$ are the logistic sigmoid and hyperbolic tangent activation functions; $\odot$ denotes the element-wise multiplication operation.

Unlike BDLM, LSTM has the ability to figure out the dependencies between input and output automatically without feature engineering. Therefore, LSTM can be applied to analyze a large number of time series. For example, in order to consider the same delayed effects between the environmental variables and the dam's responses as mentioned in Section 2.1, users only need to include a sequence of environmental variables as inputs to the LSTM, and the input-output dependency will be figured out automatically by the network.

A limitation of LSTM is that it is a deterministic model so that it cannot be coupled probabilistically with SSMs in order to provide interpretable results along with the prediction uncertainties.

2.3. Tractable Approximate Gaussian Inference (TAGI)

TAGI (Goulet et al., 2021) is a Bayesian approach for neural networks. Theoretically, TAGI can be applied to any existing network architectures. In practice, doing so requires developing specific formulations for each architecture. Goulet et al. (2021) and Nguyen and Goulet (2022) showed that the Bayesian TAGI-FNN and TAGI-CNN models match the performance of their corresponding deterministic networks trained by backpropgation while being able to provide predictive uncertainties. TAGI considers the parameters as well as all variables in the network as Gaussian random variables. It applies the Gaussian Multiplicative Approximation (GMA) in order to analytically approximate the product of Gaussian random variables as a Gaussian distribution (Goulet et al., 2021). Then, it leverages the Gaussian assumptions in order to analytically infer the network parameters.

The key advantage of TAGI is that it allows to perform Bayesian estimation analytically while other Bayesian methods rely on Monte Carlo sampling or other approximated methods which prevents them from scaling efficiently to large problems. We will show in this paper that TAGI can be applied to the LSTM architecture in order to create a Bayesian LSTM model.

2.4. Application of BDLM and LSTM in SHM

In the field of SHM, LSTM is a common tool to predict future structural responses. More specially, Qu et al. (2019) used a LSTM model to predict the deformations at multiple locations in a concrete dam taking into account the interactions between these locations. Wang et al. (2020) used LSTM to predict the crack opening development in the Longyangxia arch concrete dam in China. Nguyen-Le et al. (2020) combined a LSTM network with a hidden Markov model to predict the crack propagation in various concrete structures. Li et al. (2020) combined Loess, extra-trees and LSTM to predict displacements of concrete dams. Yang et al. (2020) combined LSTM with attention mechanism to predict a concrete dam's deformation. Tian et al. (2020) used vertical displacement data from multiple sensors located at different locations in a cable-stayed bridge as inputs for a Bidirectional LSM network to predict the tension in the bridge's cables.

A common limitation among all the above studies is that they do not take into account the uncertainties in the network parameters. Instead, they consider these parameters as fixed values that are obtained using gradient backpropagation. Moreover, the results are not interpretable so that users could not distinguish between the reversible and irreversible effects in the data.

3. METHODOLOGY

In this section, we first present the methodology to build a Bayesian TAGI-LSTM neural network. Then, we describe how to couple it probabilistically with BDLM in order to create a hybrid model.

3.1. TAGI-LSTM

We use the same LSTM architecture's equations as presented in Section 2.2, but now, we consider the parameters $\boldsymbol{\theta}$ as well as all variables including the four LSTM gates, the hidden states $\boldsymbol{H}$, and the cell states $\boldsymbol{C}$ as Gaussian random variables. We also consider these random variables as having diagonal covariance matrices so that we can maintain the computational tractability of the TAGI method. By considering $\boldsymbol{\theta}$ as having probability distributions instead of having deterministic values, we consider the epistemic uncertainty related to the model's parameters. Figure 1 presents the graph for an example of TAGI-LSTM network having an input layer containing the covariate $\boldsymbol{x}$, one LSTM layer, and a fully connected output layer $\boldsymbol{z}^{(\mathrm{O})}$. We denote the marginal prior knowledge for the forget gate at time t given

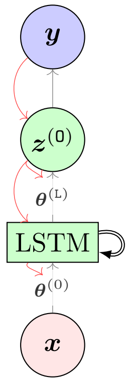


Figure 1: Graphical representation of a TAGI-LSTM network. Black arrows represent the network forward connection and red arrows represent the layer-wise inference procedure.

the data $\boldsymbol{y}_{1:t-1} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_{t-1}\}$ by the Gaussian random vector $\boldsymbol{F}_{t|t-1} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{F}}_{t|t-1}, \boldsymbol{\Sigma}^{\boldsymbol{F}}_{t|t-1})$, where $\boldsymbol{\mu}^{\boldsymbol{F}}_{t|t-1} = \mathbb{E}[\boldsymbol{F}_t|\boldsymbol{y}_{1:t-1}]$, and $\boldsymbol{\Sigma}^{\boldsymbol{F}}_{t|t-1} = \mathrm{cov}(\boldsymbol{F}_t|\boldsymbol{y}_{1:t-1})$. In the forward step, we propagate the uncertainties from the input covariates $\boldsymbol{x}$ and from the parameters $\boldsymbol{\theta}$ through the LSTM layer up to the output layer. To this end, we estimate the prior probability density functions (PDFs) for the LSTM gates $\{\boldsymbol{F}_{t|t-1}, \boldsymbol{I}_{t|t-1}, \boldsymbol{O}_{t|t-1}, \tilde{\boldsymbol{C}}_{t|t-1}\}$, the hidden states $\boldsymbol{H}_{t|t-1}$ and the cell states $\boldsymbol{C}_{t|t-1}$, as well as the output layer's hidden states $\boldsymbol{Z}^{(0)}_{t|t-1}$ using Equation 1. Applying Equation 1 requires to perform multiplications and nonlinear transformations of Gaussian random variables which do not result in Gaussian distributions. Following the TAGI method, we use GMA to approximate the product of two Gaussian random variables by a Gaussian distribution. We also approximate the nonlinear activation functions by their corresponding locally linearized functions (Goulet et al., 2021). This allows us to analytically estimate the prior predictive PDFs for all quantities in the network.

The observations $\boldsymbol{y}$ are related to the output layer's hidden states $\boldsymbol{z}^{(0)}$ by

$$\boldsymbol{y}_t = \boldsymbol{z}^{(0)}_t + \boldsymbol{v}_t, \quad \boldsymbol{v} : \boldsymbol{V} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_V),$$

where the observation error $\boldsymbol{v}_t$ takes into account both the precision of the measuring devices as well as the model's aleatory uncertainty.

In the backward step, after observing new data $\boldsymbol{y}_t$, we want to send information from the output layer back to the input layer in order to update the prior knowledge about the parameters and hidden states which has been obtained from the forward pass. In order to maintain the computational tractability of the method, we apply a layer-wise inference procedure to obtain the posteriors for hidden states and parameters (Goulet et al., 2021). The backward step is depicted by the red arrows in Figure 1. We use the Gaussian conditional equation to estimate the posterior for the hidden states of the output layer $\boldsymbol{Z}^{(0)}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{Z}^{(0)}}_{t|t}, \boldsymbol{\Sigma}^{\boldsymbol{Z}^{(0)}}_{t|t})$, and apply the Rauch-Tung-Striebel procedure (Rauch et al., 1965) to obtain the posteriors for the hidden states $\boldsymbol{H}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{H}}_{t|t}, \boldsymbol{\Sigma}^{\boldsymbol{H}}_{t|t})$, and for the cell states $\boldsymbol{C}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{C}}_{t|t}, \boldsymbol{\Sigma}^{\boldsymbol{C}}_{t|t})$ of each LSTM layer, as well as $\boldsymbol{\theta}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{\theta}}_{t|t}, \boldsymbol{\Sigma}^{\boldsymbol{\theta}}_{t|t})$ for the network parameters.

Equations 1a-1e show that there is a connection between the hidden and cell states at time $t-1$ and t. This means that there is a flow of information through time from the first to the last time steps. We can leverage this connection in order to perform smoothing over time (Rauch et al., 1965) for TAGI-LSTM and the hybrid model presented in the next section which couples TAGI-LSTM and BDLM .

3.2. *Coupling TAGI-LSTM with BDLM*

We consider a BDLM model having a level, a trend and a seasonality hidden state such that the hidden state vector is $\boldsymbol{z} = [z^{\mathtt{L}}\ z^{\mathtt{T}}\ z^{\mathtt{S}}]^{\intercal} \in \mathbb{R}^3$ (Goulet, 2020). The linear transition functions for the level and trend hidden states are given as

$$\begin{bmatrix} z^{\mathtt{L}} \\ z^{\mathtt{T}} \end{bmatrix}_t = \mathbf{A} \begin{bmatrix} z^{\mathtt{L}} \\ z^{\mathtt{T}} \end{bmatrix}_{t-1} + \boldsymbol{w}_t, \qquad (2)$$

where $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ is the transition matrix, $\boldsymbol{w}_t$ is a realization from the independent and identically distributed (i.i.d.) process error $\boldsymbol{w} : \boldsymbol{W} \sim \mathcal{N}(\boldsymbol{0}, \mathbf{Q})$. In BDLM, the seasonality can be modelled by either the periodic (Goulet, 2017) or the Kernel regression components (Nguyen et al., 2019). However, obtaining the parameters for these components relies on optimization. Here, instead, we use the TAGI-LSTM network presented in Section 3.1

to predict the seasonality as

$$z_t^{\mathtt{S}} = \mathtt{LSTM}(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}, \boldsymbol{\theta}), \tag{3}$$

where $\boldsymbol{x}_t$ is the input covariates, $\boldsymbol{h}_{t-1}$ is the hidden state vector of the TAGI-LSTM network at time $t-1$, and $\boldsymbol{\theta}$ is the network parameters. The values of $\boldsymbol{\theta}$ are updated recursively when observing new data.

The posterior PDF for the hidden states at time $t-1$ is given as $\boldsymbol{Z}_{t-1|t-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t-1|t-1}, \boldsymbol{\Sigma}_{t-1|t-1})$, where $\boldsymbol{\mu}_{t-1|t-1} = \mathbb{E}[\boldsymbol{Z}_{t-1}|y_{1:t-1}]$ and $\boldsymbol{\Sigma}_{t-1|t-1} = \mathrm{cov}(\boldsymbol{Z}_{t-1}|y_{1:t-1})$. The prior knowledge for the hidden state vector at time t is assumed to follow a multivariate Gaussian distribution $\boldsymbol{Z}_{t|t-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t|t-1}, \boldsymbol{\Sigma}_{t|t-1})$ where,

$$\begin{aligned} \boldsymbol{\mu}_{t|t-1} &= \begin{bmatrix} \mu^{\mathtt{L}} \\ \mu^{\mathtt{T}} \\ \mu^{\mathtt{S}} \end{bmatrix}_{t|t-1}, \\ \boldsymbol{\Sigma}_{t|t-1} &= \begin{bmatrix} \mathrm{var}(Z^{\mathtt{L}}) & \mathrm{cov}(Z^{\mathtt{L}}, Z^{\mathtt{T}}) & \mathrm{cov}(Z^{\mathtt{L}}, Z^{\mathtt{S}}) \\ \vdots & \mathrm{var}(Z^{\mathtt{T}}) & \mathrm{cov}(Z^{\mathtt{T}}, Z^{\mathtt{S}}) \\ \mathrm{sym.} & \cdots & \mathrm{var}(Z^{\mathtt{S}}) \end{bmatrix}_{t|t-1}. \end{aligned} \tag{4}$$

Given that the transition model in Equation 2 is linear, we can obtain the exact solutions for the means $\mu^{\mathtt{L}}_{t|t-1}$ and $\mu^{\mathtt{T}}_{t|t-1}$, the variances $\mathrm{var}(Z^{\mathtt{L}}_{t|t-1})$ and $\mathrm{var}(Z^{\mathtt{T}}_{t|t-1})$, and the covariance $\mathrm{cov}(Z^{\mathtt{L}}_{t|t-1}, Z^{\mathtt{T}}_{t|t-1})$ using the prediction step of Kalman filter (Kalman, 1960). The mean $\mu^{\mathtt{S}}_{t|t-1}$ and variance $\mathrm{var}(Z^{\mathtt{S}}_{t|t-1})$ of the seasonality hidden state are obtained by the TAGI-LSTM one-step-ahead prediction as presented in Section 3.1. We need to estimate the cross-covariances $\mathrm{cov}(Z^{\mathtt{L}}_{t|t-1}, Z^{\mathtt{S}}_{t|t-1})$ and $\mathrm{cov}(Z^{\mathtt{T}}_{t|t-1}, Z^{\mathtt{S}}_{t|t-1})$ for defining the prior distribution for $\boldsymbol{Z}_t$. In practice, these covariance terms are typically close to zero, and can theoritically be obtained through Monte Carlo sampling. However, when analyzing thousands of time series, this approach becomes computational prohibitive. Therefore, we make an assumption that the seasonality hidden variable $Z^{\mathtt{S}}$ is independent from other hidden states so that $\mathrm{cov}(Z^{\mathtt{L}}_{t|t-1}, Z^{\mathtt{S}}_{t|t-1}) = \mathrm{cov}(Z^{\mathtt{T}}_{t|t-1}, Z^{\mathtt{S}}_{t|t-1}) = 0$.

The linear observation equation is given by

$$y_t = \mathbf{C}\boldsymbol{z}_t + v_t, \tag{5}$$

where $\mathbf{C} = [1\ 0\ 1]$ is the observation matrix, and $v : V \sim \mathcal{N}(0, \sigma_V^2)$ is the observation error. Using Equations 2-5, we can apply the Kalman filter to estimate the posterior $\boldsymbol{Z}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t})$. Then, only the posterior $\boldsymbol{Z}^{\mathtt{S}}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\mathtt{S}}_{t|t}, \boldsymbol{\sigma}^{\mathtt{S}}_{t|t})$ is used to update the TAGI-LSTM's parameters $\boldsymbol{\theta}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{\theta}}_{t|t}, \boldsymbol{\Sigma}^{\boldsymbol{\theta}}_{t|t})$, the hidden states $\boldsymbol{H}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{H}}_{t|t}, \boldsymbol{\Sigma}^{\boldsymbol{H}}_{t|t})$, and the cell states $\boldsymbol{C}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{C}}_{t|t}, \boldsymbol{\Sigma}^{\boldsymbol{C}}_{t|t})$ following the layer-wise procedure presented by Goulet et al. (2021).

4. Result

In this section, we compare the performance of the Bayesian TAGI-LSTM and hybrid models with BDLM on two SHM case studies.

4.1. Case study #1

In this case study, we compare the predictive performance between TAGI-LSTM and BDLM. The data used for this case study is the pendulum dataset CB2 of the ICOLD-BW2022 benchmark (Malm R et al., 2022) which measures a dam's radial displacement. The CB2 data is available from 2000 until the end of 2012 with the average frequency of 1.5 week, whereas the daily data for the reservoir's water level (`WL`) and the air temperature (`AT`) are available from 2000 to 2018. The task consists in providing daily predictions for the CB2 dataset from 2013 to 2018.

We build a TAGI-LSTM model which uses the water level and air temperature as explanatory variables to predict the CB2 data. We train and test our model using daily data which means that there are many missing data points for the CB2 dataset. When the data for CB2 is missing, we only make predictions without updating the network parameters. Given the small number of missing data points in the water level and temperature times series, we replace their missing values by 0 instead of creating additional models to predict them. When making forecast $z_t^{\mathtt{(0)}}$ at time t, we take into account the dependencies of the past observations by including a window of $\mathtt{L}$ past output's hidden states $z^{\mathtt{(0)}}_{t-\mathtt{L}1:t-1}$ in the covariate vector $\boldsymbol{x}_t$. We also consider the lagging effects that the temperature and the reservoir's level may have on the displacement by including a window of $\mathtt{M}$ observations of both covariates as inputs for the LSTM model such that $\boldsymbol{x}_t = [x^{\mathtt{WL}}_{t-\mathtt{M}+1:t}\ x^{\mathtt{AT}}_{t-\mathtt{M}+1:t}\ z^{\mathtt{(0)}}_{t-\mathtt{L}:t-1}]^{\mathtt{T}}$. By contrast, in order to account for the same dependencies in BDLM, Deka (2022) needed to do an extensive exercise of feature engineering. That is, decomposing the water level and temperature data into components, and manually defining the dependencies between these components and the dam's displacement. This approach becomes unpractical when analyzing a large number of time series. The standard deviation for the observation error σ_V as well as the window lengths `L` and `M` are hyper-parameters of the TAGI-LSTM model which needs to be learnt from data. We perform a grid-search to find the best values for the hyper-parameters. For each candidate value in the grid, we train our models with early-stopping (Murphy, 2013) on a subset of training data from 2000 to end of 2009, and report the log-likelihood for the validation period from 2010 to end of 2012. The values which maximize the log-likelihood of the validation set are chosen as the final hyper-parameter values. Appendix A presents the optimal hyper-parameter values used in our model.

Figure 2 shows that the TAGI-LSTM model can not only provide the predictions but also the uncertainties associated with these predictions; and the TAGI-LSTM model provides a performance compatable to that of BDLM while not requiring feature engineering.

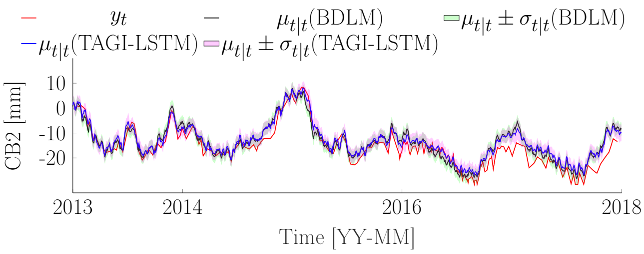


Figure 2: Comparison of predictions on test set between TAGI-LSTM and BDLM models.

4.2. Case study #2

In this case study, we evaluate the ability of the hybrid model proposed in Section 3.2 not only at forecasting but also at decomposing the data into interpretable components. We compare our model with BDLM on two daily time series data. The first data records the displacement (DIS), and the second data measures the crack opening (CR) at two specific locations in a concrete dam. The task consists in forecasting the last year of data (365 data points). The validation set includes one year of data before the test set, and the rest of the data before the validation set is used for training. We train our models on multiple epochs to identify the optimal ones. The hyper-parameters of the network are presented in Appendix A. The initial hidden states $\boldsymbol{\mu}_0^{(i+1)}$ and $\boldsymbol{\Sigma}_0^{(i+1)}$ at the $i+1^{th}$ epoch are the smoothed estimates $\boldsymbol{\mu}_{0|\mathtt{T}}^{(i)}$ and $\boldsymbol{\Sigma}_{0|\mathtt{T}}^{(i)}$ at the i^{th} epoch with `T` being the last training time. In order to make a fair comparison with BDLM models, we retrain our models with the full training data including the validation data with the optimal number of epochs previously obtained. We build a separate BDLM model for each dataset where each model has a local level, a local trend, a Kernel regression, and an autoregressive component. Figure 3 compares the predictions on the test set of the crack opening time series between our method and BDLM where it shows that our hybrid model provides more accurate predictions. Figure 4 compares the predictions on the test set as well as the hidden states between our model and BDLM for the displacement time series. It shows that our model is capable of decomposing the data into interpretable components. Table 1 reports the test Root Mean Square Error (RMSE) performance for the two methods. It shows that our hybrid model provides similar results compared to BDLM while relieving the burden of performing optimization and feature engineering. Figures 4c-4f show that our model provides smaller uncertainties for the level and trend hidden states compared to those obtained by BDLM. This is because we

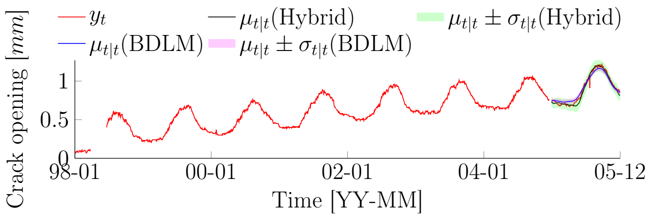
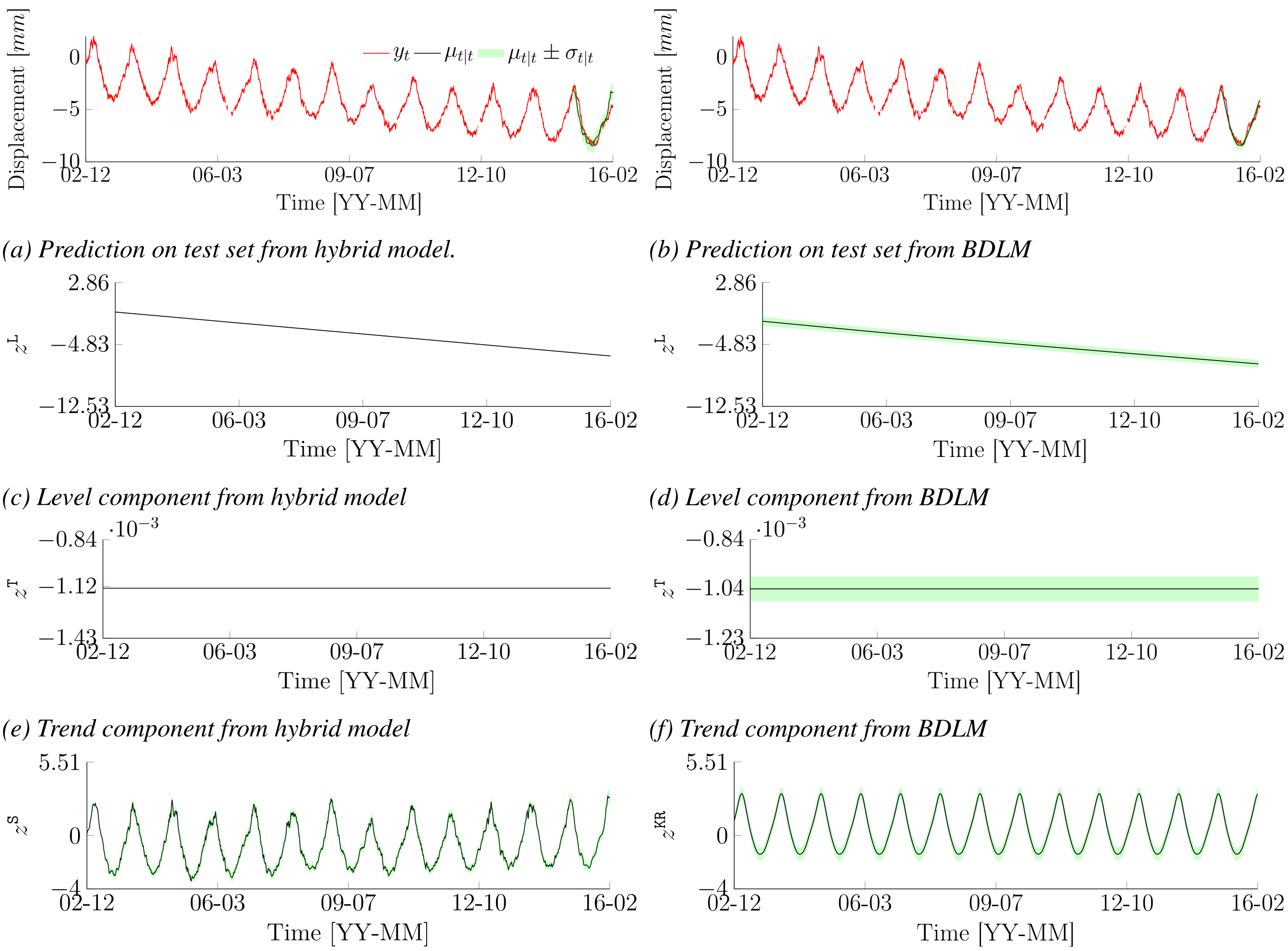


Figure 3: Comparison of predictions on test set between our hybrid model and BDLM for crack opening time series.



(a) Prediction on test set from hybrid model.

(b) Prediction on test set from BDLM

(c) Level component from hybrid model

(d) Level component from BDLM

(e) Trend component from hybrid model

(f) Trend component from BDLM

(g) Seasonality component from hybrid model

(h) Kernel regression component from BDLM

Figure 4: Comparison of predictions and hidden states between our hybrid model and BDLM for displacement time series.

train our model over multiple epochs, and the uncertainties of these hidden states reduce with respect to the number of epochs. Figures 4g-4h show that the seasonality modelled by TAGI-LSTM can capture long and short-term variations, whereas the one modelled by the Kernel regression component is smoother.

Table 1: RMSE performance of our hybrid model and BDLM on test set for displacement and crack opening time series.

Dataset	Hybrid	BDLM
Displacement	0.688	**0.368**
Crack opening	**0.027**	0.036

5. CONCLUSION

In this paper, we combined the existing LSTM architecture and the TAGI method to create the Bayesian TAGI-LSTM neural network. We showed that the TAGI-LSTM network takes into account both the epistemic and aleatory uncertainties. We also showed that TAGI-LSTM can automatically figure out the input-output dependencies without requiring feature engineering. This aspect is the key to enable analyzing a large number of time series. The parameters of TAGI-LSTM can be learnt analytically using Bayesian inference. This allows coupling TAGI-LSTM with BDLM in a probabilistic manner. The results from case studies showed that the hybrid model can decompose data into interpretable components while providing the predictive uncertainties.

6. ACKNOWLEDGEMENTS

This project is funded by Natural Sciences and Engineering Research Council of Canada (NSERC), Hydro Québec and Hydro Québec's Research Institute (IREQ).

A. APPENDIX

Table 2: Network architecture and hyper-parameters

	Case study #1	Case study #2	
Dataset	CB2	DIS	CR
# LSTM layer	1	1	1
# LSTM nodes	50	100	100
# optimal epoch	29	50	15
$\mathtt{L}$	35	365	365
$\mathtt{M}$	21	-	-
σ_V	0.2	0.3	0.2

B. REFERENCES

Deka, B. (2022). "Analytical bayesian parameter inference for probabilistic models with engineering applications." Ph.D. thesis, Polytechnique Montréal, Canada.

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

Goulet, J.-A. (2020). *Probabilistic Machine Learning for Civil Engineers*. MIT Press.

Goulet, J.-A., Nguyen, L. H., and Amiri, S. (2021). "Tractable approximate gaussian inference for Bayesian neural networks." *Journal of Machine Learning Research*, 22, 1–23.

Hochreiter, S. and Schmidhuber, J. (1997). "Long short-term memory." *Neural computation*, 9, 1735–80.

Kalman, R. E. (1960). "A New Approach to Linear Filtering and Prediction Problems." *Journal of Basic Engineering*, 82(1), 35–45.

Li, Y., Bao, T., Gong, J., Shu, X., and Zhang, K. (2020). "The prediction of dam displacement time series using STL, extra-trees, and stacked LSTM neural network." *IEEE Access*, 8, 94440–94452.

Malm R, K., M Simon, A., F, S., and R, H. (2022). "Summary of Theme A: Behaviour prediction of a concrete arch dam." 16th *International Benchmark Workshop on Numerical Analysis of Dams*, International Commission on Large Dams.

Murphy, K. P. (2013). *Machine learning : a probabilistic perspective*. MIT Press.

Nguyen, L. H., Gaudot, I., Khazaeli, S., and Goulet, J.-A. (2019). "A kernel-based method for modeling nonharmonic periodic phenomena in bayesian dynamic linear models." *Frontiers in Built Environment*, 5, 8.

Nguyen, L.-H. and Goulet, J.-A. (2022). "Analytically tractable hidden-states inference in bayesian neural networks." *Journal of Machine Learning Research*, 23(50), 1–33.

Nguyen-Le, D., Tao, Q. B., Nguyen, V.-H., Abdel Wahab, M., and Nguyen-Xuan, H. (2020). "A data-driven approach based on long short-term memory and hidden Markov model for crack propagation prediction." *Engineering Fracture Mechanics*, 235, 107085.

Qu, X., Yang, J., and Chang, M. (2019). "A deep learning model for concrete dam deformation prediction based on RS-LSTM." *Journal of Sensors*, 2019, 1–14.

Rauch, H. E., Tung, F., and Striebel, C. T. (1965). "Maximum likelihood estimates of linear dynamic systems." *AIAA Journal*, 3(8), 1445–1450.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). "Learning representations by back-propagating errors." *Nature*, 323, 533–536.

Tian, Y., Xu, Y., Zhang, D., and Li, H. (2020). "Relationship modeling between vehicle induced girder vertical deflection and cable tension by BiLSTM using field monitoring data of a cable stayed bridge." *Structural Control and Health Monitoring*, 28.

Wang, J., Zou, Y., Lei, P., Sherratt, S., and Wang, L. (2020). "Research on recurrent neural network based crack opening prediction of concrete dam." *Journal of Internet Technology*, 21, 1151.

Yang, D., Gu, C., Zhu, Y., Dai, B., Zhang, K., Zhang, Z., and Li, B. (2020). "A concrete dam deformation prediction method based on LSTM with attention mechanism." *IEEE Access*, 8, 185177–185186.