Title

A variational Bayesian neural network for structural health monitoring and cost-informed decision-making in miter gates

Journal

Structural Health Monitoring, 21(1)

ISSN

1475-9217

Authors

Vega, Manuel A
Todd, Michael D

Publication Date

2022

DOI

10.1177/1475921720904543

Peer reviewed

A Variational Bayesian Neural Network for Structural Health Monitoring and Cost-Informed Decision-Making in Miter Gates

Manuel A. Vega[1] and Michael D. Todd[1*]

*[1]Dept. of Structural Engineering, Univ. of California San Diego, 9500 Gilman Dr. La Jolla, California, USA. Email: mdtodd@eng.ucsd.edu**

ABSTRACT

Many physics-based and surrogate models used in structural health monitoring (SHM) are affected by different sources of uncertainty such as model approximations and simplified assumptions. Optimal SHM and prognostics are only possible with uncertainty quantification that leads to an informed course of action. In this paper, a Bayesian Neural Network (BNN) using variational inference is applied to learn a damage feature from a high-fidelity finite element model. BNNs can learn from small and noisy datasets and are more robust to overfitting than Artificial Neural Networks (ANNs), which make it very suitable for applications such as SHM. Also, uncertainty estimates obtained from a trained BNN model are used to build a cost-informed decision-making process. To demonstrate the applicability of BNNs, an example of this approach applied to miter gates is presented. In this example, a degradation model based on real inspection data is used to simulate the damage evolution.

KEYWORDS: Miter gates, artificial neural networks, surrogate model, finite element, inverse model

1. INTRODUCTION

An ANN is a machine learning algorithm widely used in many areas in science and engineering. They are attractive alternatives to physics-based modeling, particularly for complex structures with unknown failure modes or highly variable operational and environmental inputs [1]. Most of the applications in civil engineering are in pattern recognition problems. The first journal article on neural network application in civil/structural engineering was published by Adeli and Yeh [2], which was an ANN algorithm trained to inform if a particular engineering design was acceptable or not. Hajela and Berke [3] applied ANN algorithms for structural optimization. Theocaris and Panagiotopoulos [4] used ANN algorithms to learn the parameter identification problem in fracture mechanics. For SHM applications, Wu et al. [5] and Feng and Bahng [6] trained an ANN algorithm to detect structural damage in the form of reduction in member stiffness on a multistory shear building and reinforced concrete bridge columns respectively. Other researchers used changes in modally-derived features such as mode shapes, eigenvectors and Ritz vectors, from numerical and experimental samples, to train an ANN to diagnose damage [7,8].

Various other SHM algorithms for civil engineering infrastructure, such as bridges [9–11] and buildings [12–14], have been implemented based on ANN architectures. Waszczyszyn and Ziemianski [15] and Adeli [16] reviewed several more application in civil engineering including the use of neural networks in analysis and design of structures, system identification, structural control, finite

element (FE) mesh generation and other disciplines in civil engineering. Some researchers have used ANNs as surrogate models, using validated FE models to generate data to train the network [10,11,17,18]. Many of these researchers have used ANNs as emulators of computationally expensive high-fidelity finite element model runs.

In general, ANN algorithms are trained using optimization techniques such as gradient descent [19]. Therefore, ANN models are generally used to build point prediction models. Recently, Bayesian prediction models have started to be more attractive for damage assessment, especially in civil engineering because the limited amount of data available to build a reliable deterministic point prediction model. Many researchers use Gaussian Process (GP) regression to build Bayesian prediction models for civil engineering structures [20]. However, GP models are computationally challenging for high-dimensional spaces or otherwise "large" data sets. Due to the scalability limitations in GP models [21–23], BNN [24–26] models have started to be more practical model when dealing with high-dimensional space. In the case of SHM, this space depends on the number of spatially-distributed sensors and their collection (monitoring) frequency. BNNs are preferred over deterministic mathematical models such as neural networks because they account for uncertainty in their parameters (i.e., weights and biases) and propagate this into their predictions. Such uncertainty management is critical to support decision-making, which is the necessary outcome of an SHM process [27]. BNNs are more robust against overfitting because a posterior distribution of the parameters is considered instead of using deterministic parameters that minimize the empirical risk during training. Also, BNN may be trained using limited and noisy data, while ANNs typically tend to require more and lower-noise training data for equivalent performance.

In this paper, a BNN is trained with a FE model due to such highly-limited data availability. Therefore, this probabilistic prediction models also serve as a surrogate model of a validated high-fidelity FE model, which sometimes are unable to be used efficiently to make fast predictions. Specifically, the BNN model is trained to assess the condition of quoin blocks in miter gates, which are essential civil structures for navigation system in rivers. In this work, a degradation model based on real inspection data of miter gates is used to simulate the damage evolution. Additionally, a cost function is introduced to improve prioritization of maintenance events of components of miter gates. The added value of using SHM in miter gates is evaluated in term of maintenance cost savings. The ultimate goal of the authors is to set up a SHM workflow that allows further optimization in term of cost savings, and this paper presents one realization of this goal within a civil structural monitoring application.

2. BAYESIAN NEURAL NETWORK

2.1. ARTIFICIAL NEURAL NETWORK

Two of the main problems in machine learning are classification (for discrete classes) and regression (for continuous processes). An ANN is a powerful supervised learning algorithm that can be used to solve classification and non-linear regression problems. In the context of an SHM problem, an ANN can be trained to learn the relationship between sensor values or features derived from sensor values and damage classes or parameters. Figure 1 shows the ANN architecture used in this paper and the non-linear functions (i.e. sigmoid and softplus) that are used to learn the relationship between sensor information and structural damage targets.

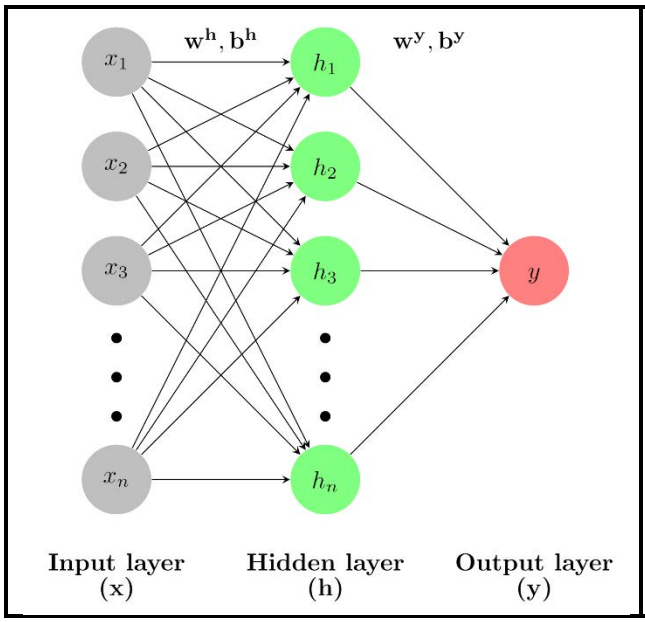


Figure 1: ANN architecture and definitions

Generally, gradient descent algorithms are used to train neural networks. These algorithms objective is to find the ANN parameters (i.e. weights and biases) that minimize an error or loss function that depends on the ANN outputs (y) and the true (training) output values. The training error is known in the machine learning community as the empirical risk. Commonly, regularization is used to avoid overfitting, i.e., substantially degrading network performance when it is presented any data set other than the training set. Another way to train an ANN to be robust against overfitting is to use a Bayesian approach to find the parameters of the network. The uncertainty in these weights and biases can be propagated into network predictions, which is useful in the context SHM problems that involves (cost-informed) decisions.

2.2. BAYESIAN NEURAL NETWORK

BNNs are essentially neural networks with a prior distribution on their network parameters [28]. The joint posterior distribution $p(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma} \mid \mathbf{Y}, \mathbf{X})$ of the network parameters—including the covariance matrix, $\boldsymbol{\Sigma}$, of the assumed zero mean error—after observing a set of training data $\mathbf{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$ may be expressed as

$$p(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma} \mid \mathbf{Y}, \mathbf{X}) = \frac{p(\mathbf{Y} \mid \mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}, \mathbf{X}) p(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma} \mid \mathbf{X})}{p(\mathbf{Y} \mid \mathbf{X})} = \frac{p(\mathbf{Y} \mid \mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}, \mathbf{X}) p(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma})}{\iiint_{\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}} p(\mathbf{Y} \mid \mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}, \mathbf{X}) p(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}) \, d\mathbf{w} d\mathbf{b} d\boldsymbol{\Sigma}}, \quad (1)$$

where $\mathbf{w}$ and $\mathbf{b}$ represents the weights and biases of the BNN, respectively. Also, $\mathbf{X}$ and $\mathbf{Y}$ are defined as $\mathbf{X}=\left[\mathbf{x}_1^T,\mathbf{x}_2^T,...,\mathbf{x}_N^T\right]^T \in^{\sim}{}^{(N\times\dim(\mathbf{x}))\times 1}$ and $\mathbf{Y}=\left[\mathbf{y}_1^T,\mathbf{y}_2^T,...,\mathbf{y}_N^T\right]^T \in^{\sim}{}^{(N\times\dim(\mathbf{y}))\times 1}$ respectively.

The conditional distribution, $p(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}\,|\,\mathbf{X})$ is equivalent to $p(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma})$ due to conditional independence. The marginal probability density function, $p(\mathbf{Y}\,|\,\mathbf{X})$, can be obtained by integrating $\iiint_{\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}} p(\mathbf{Y}\,|\,\mathbf{w},\mathbf{b},\boldsymbol{\Sigma},\mathbf{X})\,p(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma})\,d\mathbf{w}d\mathbf{b}d\boldsymbol{\Sigma}$, which it is generally mathematically intractable. Furthermore, the likelihood function $p(\mathbf{Y}\,|\,\mathbf{w},\mathbf{b},\boldsymbol{\Sigma},\mathbf{X})$ can be expressed as

$$p(\mathbf{Y}\,|\,\mathbf{w},\mathbf{b},\boldsymbol{\Sigma},\mathbf{X})=\prod_{i=1}^{N} p(\mathbf{y}_i\,|\,\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}_i,\mathbf{x}_i). \tag{2}$$

In this paper, the following measurement model/equation is assumed

$$\mathbf{y}_i = NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})+\boldsymbol{\varepsilon}_i, \text{ where } \boldsymbol{\varepsilon}_i \sim \mathcal{N}\left(\mathbf{0},\underbrace{\begin{bmatrix}\sigma_{11}^2 & \sigma_{12}^2 & \dots & \sigma_{1k}^2\\ \sigma_{21}^2 & \sigma_{22}^2 & \dots & \sigma_{2k}^2\\ \vdots & \vdots & \ddots & \vdots\\ \sigma_{k1}^2 & \sigma_{k2}^2 & \dots & \sigma_{kk}^2\end{bmatrix}}_{\boldsymbol{\Sigma}_i}\right), k=\dim(\mathbf{y}_i), \tag{3}$$

which is equivalent the following

$$\begin{aligned} p(\mathbf{y}_i\,|\,\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}_i,\mathbf{x}_i) &\sim \mathcal{N}(NN(\mathbf{x}_i;\mathbf{w};\mathbf{b}),\boldsymbol{\Sigma}_i)\\ &= \frac{1}{(2\pi)^{k/2}}\left|\boldsymbol{\Sigma}_i\right|^{-1/2}\exp\left\{-\frac{1}{2}\left[\mathbf{y}_i-NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})\right]\boldsymbol{\Sigma}_i^{-1}\left[\mathbf{y}_i-NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})\right]^T\right\}, \end{aligned} \tag{4}$$

where $\boldsymbol{\Sigma}_i$ is the covariance of the measurement error between observation $\mathbf{y}_i$ and model prediction $NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})$ with weights $\mathbf{w}$, biases $\mathbf{b}$, and input $\mathbf{x}_i$. $\mathcal{N}(\ )$ represent a normal distribution parametrized by its mean and variance, and $k=\dim(\mathbf{y}_i)$.

For the prior $p(\mathbf{w},\mathbf{b},\boldsymbol{\sigma})$, it is assumed that $\mathbf{w}$, $\mathbf{b}$ and $\boldsymbol{\Sigma}$ are statistically independent. The joint prior can be expressed as follows:

$$p(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma})=p(\mathbf{w})\times p(\mathbf{b})\times\prod_{i=1}^{N} p(\boldsymbol{\Sigma}_i), \tag{5}$$

where $p(\mathbf{w}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $p(\mathbf{b}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and $p(\mathbf{\Sigma}_i) \sim \text{Lognormal}(2\mathbf{I}, \mathbf{I})$ are the priors used in this paper.

The posterior predictive distribution of $\mathbf{Y}_{\text{test}}$ for a set of observed points $\mathbf{X}_{\text{test}}$ is then

$$p(\mathbf{Y}_{\text{test}} \mid \mathbf{X}_{\text{test}}, \mathbf{Y}_{\text{train}}, \mathbf{X}_{\text{train}}) = \iiint_{\mathbf{w},\mathbf{b},\mathbf{\Sigma}} p(\mathbf{Y}_{\text{test}} \mid \mathbf{w}, \mathbf{b}, \mathbf{\Sigma}, \mathbf{X}_{\text{test}}) p(\mathbf{w}, \mathbf{b}, \mathbf{\Sigma} \mid \mathbf{Y}_{\text{train}}, \mathbf{X}_{\text{train}}) d\mathbf{w} d\mathbf{b} d\mathbf{\Sigma}, \quad (6)$$

where the joint posterior distribution, $p(\mathbf{w}, \mathbf{b}, \mathbf{\Sigma} \mid \mathbf{Y}_{\text{train}}, \mathbf{X}_{\text{train}})$, is estimated after observing a set of training data $(\mathbf{X}_{\text{train}}, \mathbf{Y}_{\text{train}})$. The covariance matrix ($\mathbf{\Sigma}$) of the assumed zero mean error is also treated as an unknown parameter during training, and its uncertainty is also accounted in the BNN predictions.

2.3. VARIATIONAL INFERENCE

In order to obtain a trained BNN for predictions, the key part is to calculate the posterior distribution of the parameters after observing the (training) data. The posterior distribution of the parameters is typically mathematically intractable due to the normalization term (see Eq. (1)), which is a high-dimensional integral. The two most popular approximation methods to obtain the posterior distribution are Markov chain Monte Carlo (MCMC) and variational inference (VI). For BNNs, there many parameters to be inferred making this a high-dimensional problem. MCMC is proven to approximate very well to the true posterior. MCMC algorithms involve sampling-based methods, and it is very challenging to sample a high-dimensional posterior [29]. The Gibbs sampler is one MCMC algorithm that can work on high-dimensional space; however, it still can be computationally expensive [30]. Therefore, VI is a more practical approach in this case, which is becoming popular in BNN designs [31]. In this paper, VI is employed to infer the high-dimensional space of the parameters of the BNN that serves as a mathematical model of a nonlinear mapping between inputs and targets.

2.3.1. VARIATIONAL INFERENCE FOR A BAYESIAN NEURAL NETWORK

The idea of VI is to postulate a family of distributions, Q, and to find the closest member, $q^*(\mathbf{w}, \mathbf{b}, \mathbf{\Sigma})$, from the family of distributions that approximates to the posterior distribution, $p(\mathbf{w}, \mathbf{b}, \mathbf{\Sigma} \mid \mathbf{Y}, \mathbf{X})$, using Kullback–Leibler (KL) divergence to maximize the evidence lower bound (ELBO). For simplification purposes the parameter, $\boldsymbol{\theta}$, would represent the parameters $\mathbf{w}$, $\mathbf{b}$, and $\mathbf{\Sigma}$ as follows

$$q^*(\boldsymbol{\theta}) = \arg\min_{q(\boldsymbol{\theta}) \in Q} \text{KL}(q(\boldsymbol{\theta}) \,||\, p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})), \quad (7)$$

where the KL divergence between the variational distribution $q(\boldsymbol{\theta})$ and posterior distribution $p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})$ is given by

$$
\begin{aligned}
&\mathrm{KL}(q(\boldsymbol{\theta}) \,||\, p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})) \\
&= \int_{-\infty}^{\infty} q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})} = \mathrm{E}_{q(\boldsymbol{\theta})}[\log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})}] \\
&= \mathrm{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})] - \mathrm{E}_{q(\boldsymbol{\theta})}[\log p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})] \\
&= \mathrm{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})] - \mathrm{E}_{q(\boldsymbol{\theta})}[\log \frac{p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{X}) p(\boldsymbol{\theta})}{p(\mathbf{Y} \mid \mathbf{X})}] \\
&= \mathrm{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})] - \mathrm{E}_{q(\boldsymbol{\theta})}[\log(p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{X}) p(\boldsymbol{\theta}))] + \mathrm{E}_{q(\boldsymbol{\theta})}[\log p(\mathbf{Y} \mid \mathbf{X})].
\end{aligned} \tag{8}
$$

The $\mathrm{E}_{q(\boldsymbol{\theta})}[\log p(\mathbf{Y} \mid \mathbf{X})]$ term is constant because the normalization term $p(\mathbf{Y} \mid \mathbf{X})$ is a constant. Next, the ELBO is defined as

$$
\mathrm{ELBO}(q(\boldsymbol{\theta})) = \mathrm{E}_{q(\boldsymbol{\theta})}[\log(p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{X}) p(\boldsymbol{\theta}))] - \mathrm{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})]. \tag{9}
$$

Substituting Eq. (9) into Eq. (7) yields

$$
\mathrm{KL}(q(\boldsymbol{\theta}) \,||\, p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})) = -\mathrm{ELBO}(q(\boldsymbol{\theta})) + \mathrm{E}[\log p(\mathbf{Y} \mid \mathbf{X})]. \tag{10}
$$

Now, the closest member, $q^*(\boldsymbol{\theta})$, from the family of distributions that approximates to the posterior distribution can be found by maximizing the ELBO:

$$
q^*(\boldsymbol{\theta}) = \arg\min_{q(\boldsymbol{\theta}) \in Q} \mathrm{KL}(q(\boldsymbol{\theta}) \,||\, p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{X})) = \arg\max_{q(\boldsymbol{\theta}) \in Q} \mathrm{ELBO}(q(\boldsymbol{\theta})) \tag{11}
$$

So now, the inference of the posterior distribution can be seen as an optimization problem. For comparison purposes with loss functions used for training an ANN model, the following loss function, L, is defined:

$$
\begin{aligned}
L(q(\boldsymbol{\theta})) &= -\mathrm{ELBO}(q(\boldsymbol{\theta})) = -\mathrm{E}_{q(\boldsymbol{\theta})}[\log(p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{X}) p(\boldsymbol{\theta}))] + \mathrm{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})] \\
&= -\mathrm{E}_{q(\boldsymbol{\theta})}[\log p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{X})] - \mathrm{E}_{q(\boldsymbol{\theta})}[\log p(\boldsymbol{\theta})] + \mathrm{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})] \\
&= -\mathrm{E}_{q(\boldsymbol{\theta})}[\log p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{X})] + \mathrm{E}_{q(\boldsymbol{\theta})}[\log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta})}] \\
&= -\mathrm{E}_{q(\boldsymbol{\theta})}[\log p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{X})] + \mathrm{KL}(q(\boldsymbol{\theta}) \,||\, p(\boldsymbol{\theta}))
\end{aligned} \tag{12}
$$

Substituting Eq (2) into Eq (12), the following relation is obtained

$$
\begin{aligned}
L(q(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma})) &= -\mathrm{E}_{q(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma})}[\log\left(\prod_{i=1}^{N} p(\mathbf{y}_i \mid \mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}_i, \mathbf{x}_i)\right)] + \mathrm{KL}(q(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}) \,||\, p(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma})) \\
&= -\sum_{i=1}^{N} \mathrm{E}_{q(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma})}[\log p(\mathbf{y}_i \mid \mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}_i, \mathbf{x}_i)] + \mathrm{KL}(q(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}) \,||\, p(\mathbf{w}, \mathbf{b}, \boldsymbol{\Sigma}))
\end{aligned} . \tag{13}
$$

Now, the simplified relation is obtained after substituting Eq (4) into Eq (13)

$$
\begin{aligned}
L(q(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma})) &= -\sum_{i=1}^{N} \mathrm{E}_{q(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma})}[\log p(\mathbf{y}_i \mid \mathbf{w},\mathbf{b},\boldsymbol{\Sigma}_i,\mathbf{x}_i)] + \mathrm{KL}(q(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}) \,||\, p(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma})) \\
&= -\sum_{i=1}^{N} \mathrm{E}_{q(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma})}[\log(\frac{1}{(2\pi)^{k/2}}|\boldsymbol{\Sigma}_i|^{-1/2}) \\
&\quad + \left\{-\frac{1}{2}\left[\mathbf{y}_i - NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})\right]\boldsymbol{\Sigma}_i^{-1}\left[\mathbf{y}_i - NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})\right]^T\right\})] \\
&\quad + \mathrm{KL}(q(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}) \,||\, p(\mathbf{w},\mathbf{b},\boldsymbol{\Sigma}))
\end{aligned}
\quad . \qquad (14)
$$

If the following covariance matrix is assumed

$$
\boldsymbol{\Sigma}_i = \begin{bmatrix} \sigma^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2 \end{bmatrix} = \sigma^2\mathbf{I} = \lambda^{-1}\mathbf{I}, \qquad (15)
$$

and it is assumed that the $\boldsymbol{\Sigma}_i$'s are statistically independent between observations $\mathbf{y}_i$, then the loss function can be expressed as

$$
\begin{aligned}
L(q(\mathbf{w},\mathbf{b},\lambda)) &= -\sum_{i=1}^{N} \mathrm{E}_{q(\mathbf{w},\mathbf{b},\lambda)}[\frac{k}{2}(2\pi) + \frac{k}{2}\log\lambda^{-1} + \left\{-\frac{\lambda}{2}\left\|\mathbf{y}_i - NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})\right\|^2\right\})] \\
&\quad + \mathrm{KL}(q(\mathbf{w},\mathbf{b},\lambda) \,||\, p(\mathbf{w},\mathbf{b},\lambda))
\end{aligned}
. \qquad (16)
$$

The first term in Eq. (16) is known as the *average likelihood*, which can be minimized when the model prediction $NN(\mathbf{x}_i;\mathbf{w};\mathbf{b})$ explains the observed data $\mathbf{y}_i$. This term is also minimized when the variational distribution $q(\mathbf{w},\mathbf{b},\lambda)$ is optimally selected. The second term is the KL divergence between the variational distribution $q(\mathbf{w},\mathbf{b},\lambda)$ and the prior $p(\mathbf{w},\mathbf{b},\lambda)$, which minimizes the loss when the variational distribution is close to the prior. Therefore, this loss function balances the variational distribution $q(\mathbf{w},\mathbf{b},\lambda)$ with the likelihood $\prod_{i=1}^{N} p(\mathbf{y}_i \mid \mathbf{w},\mathbf{b},\lambda,\mathbf{x}_i)$ and the prior $p(\mathbf{w},\mathbf{b},\lambda)$. For further details on how to compute these terms given different types of variational distributions and how to derive their respective gradients of the loss function can be found here [32,33]. There are several gradient based methods to calculate the optimal $q(\mathbf{w},\mathbf{b},\lambda)$ that minimize this loss function (or maximize the ELBO). A stochastic gradient descent strategy has been used for this application using the Edward [34] probabilistic framework, which is based on Tensorflow [35], a widely used programming language for deep learning neural networks. This gradient descent method consists in calculating a noisy gradient from Monte Carlo samples of the ELBO distribution.

3. DAMAGE DETECTION USING A BNN SURROGATE MODEL

In structural health monitoring (SHM), damage classification (or regression) is generally an inverse problem, e.g., damage (input) causes change in an observable (output). Therefore, a Bayesian approach using a finite element model (or a surrogate model) is generally used to infer the inverse problem. In this paper, a BNN surrogate model is used to directly learn the inverse problem, where the output data (e.g., strain measurements) and input data (i.e. the damage, to be defined below) of a validated FE model become the input data and output data, respectively, for the BNN surrogate model. Figure 2 illustrates the data used to validate the FE model, the data generated to train the BNN model, and introduces the need for a degradation model (i.e. damage evolution) to allow cost-informed decisions, which is explain in section 4.

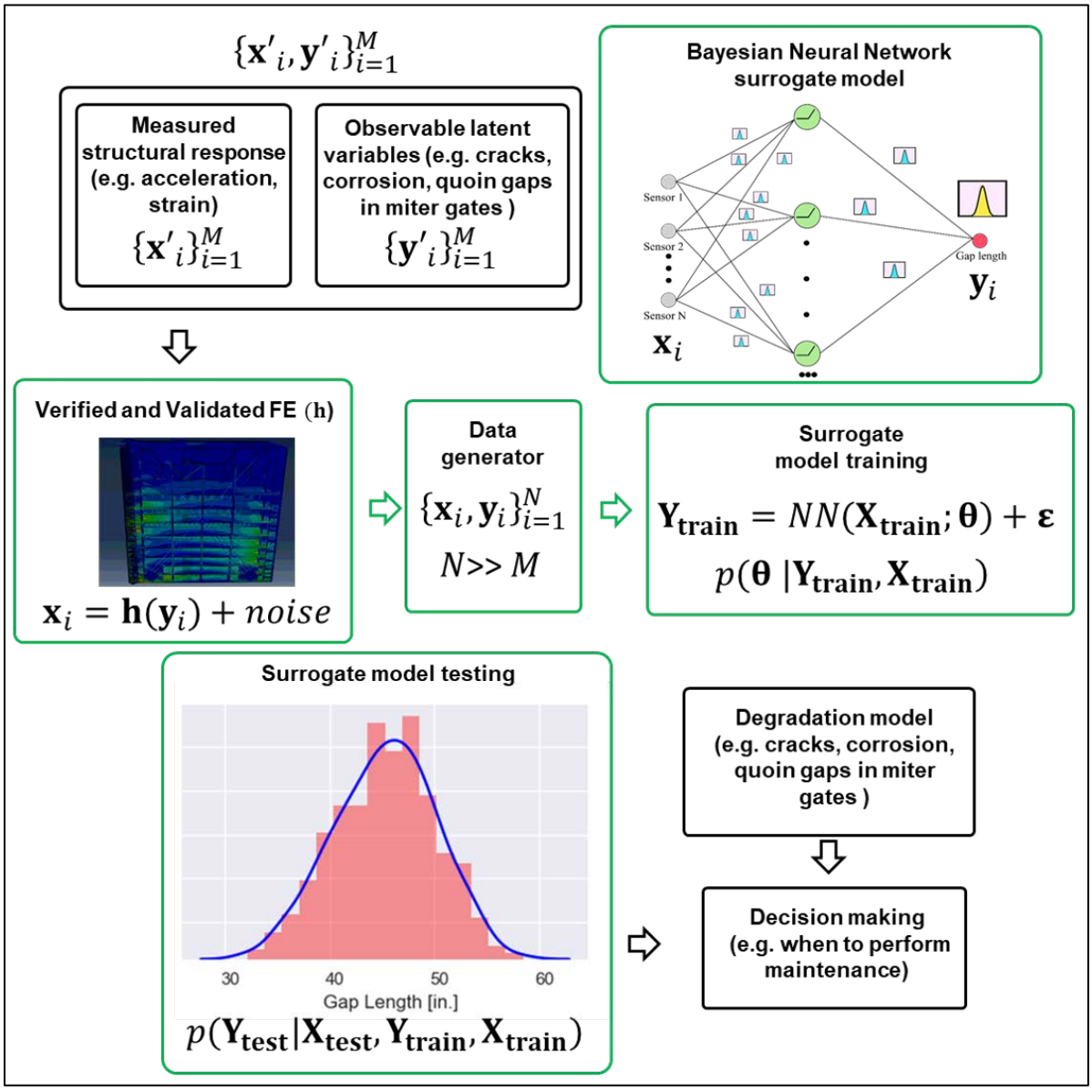


Figure 2: Decision flow based on BNN model for damage detection

3.1. HIGH FIDELITY FINITE ELEMENT MODEL OF MITER GATES

It is imperative in the cargo ship navigation to avoid unexpected closures, which can cause considerable economical loss to the marine cargo and associated industries. In the United States, the U.S. Army Corps of Engineers (USACE) owns and operates 236 locks at 191 sites [36]. According to a report published by USACE in 2017, more than half of these assets are older than their economic design life of 50 years and need a prudent structural health monitoring solution to ensure their safe and reliable operation [37]. SHM of miter gates of navigation locks, as shown in Figure 3, are a good case study on which to demonstrate feasibility of using BNNs as a damage detection solution for a real-world problem.

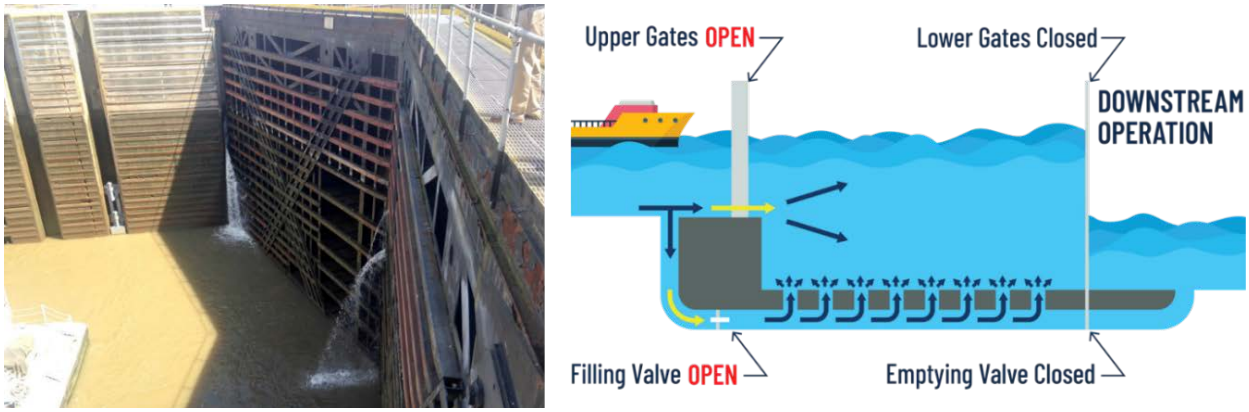


Figure 3: Navigation in Miter Gates

Some lock operators and experienced engineers from USACE [37] have stated that the condition of the quoin and gaps between the lock wall and the quoin block is one of the primary concerns within the inspection, maintenance, and repair cycle. A "gap" is somewhat generically referred to as the loss of bearing contact between the quoin attached to the gate and the lock wall. Such a gap in the quoin block changes the load path in the miter gate, leading often to higher stresses at some places in the lock gate (e.g., the pintle), which in turn can lead to operational and/or structural failure. Some miter gates owned by USACE are currently instrumented with strain gauges for in-situ data acquisition [38]. The fundamental inference is made that changes in the gap contact state will lead to observable changes in the measured gate strain field.

FE models could be used to map the strain gauges data to a specific gap condition (usually quantified by size) in an inverse analysis. However, these models are computationally expensive to run, and sometimes they are not feasible for real-time health monitoring or for monitoring fluctuating environmental effects. Consequently, a surrogate model with fast predictions of the target damage (e.g., the gap) can be employed. Figure 4 shows the ABAQUS FE model for the Greenup miter gate located in the Ohio River in Kentucky, USA. The FE model has been previously validated [39] with the available strain gage readings from the Greenup miter gate. The Greenup gate is a brand-new gate where a negligible gap was assumed for validation purposes. All the element in the gate are 3D linear shells elements to reduce the computational cost of such a large model.

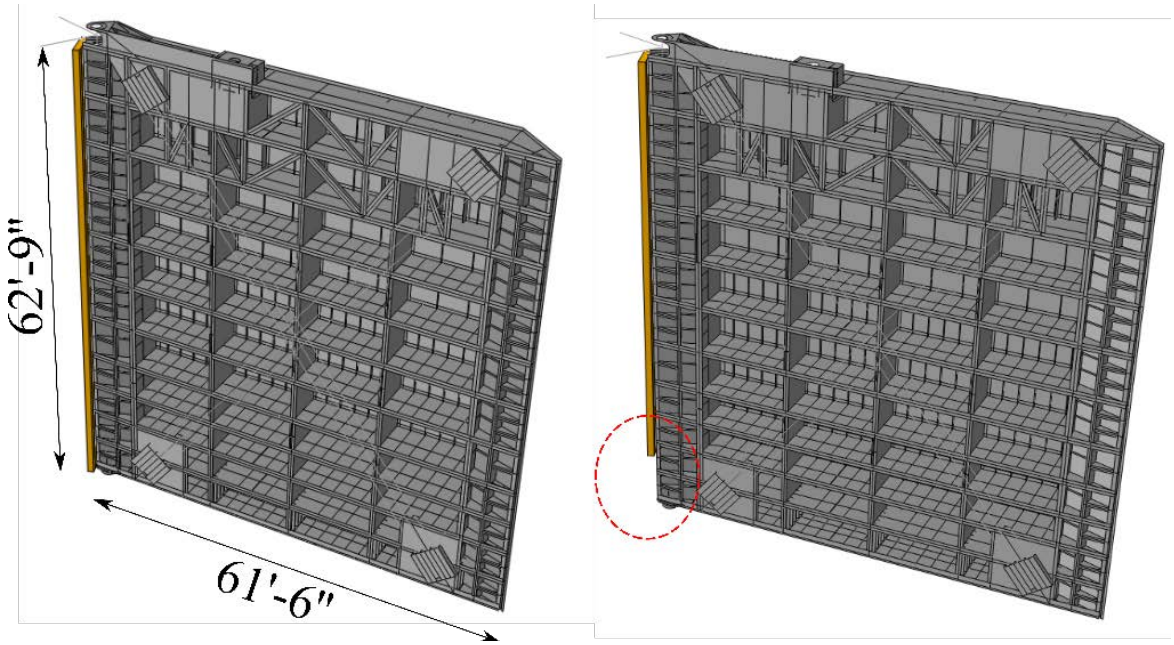


Figure 4: Gap modeling (Left: No gap, Right: Schematic gap)

A contact-type constraint is used between the lock wall (denoted in yellow) and the gate (denoted in gray), making this a nonlinear problem. To impose the contact constraint the Lagrange multiplier method was employed. The strain gauge locations are far from the contact area, mostly due to physical constraints in the miter gate, but this far-field location also mitigates errors due to the method employed to enforce the contact constraint. The opposite side of the lock wall uses fixed boundary conditions, and symmetry boundary conditions are used at the right end (i.e., miter) of the gate to simulate the right leaf.

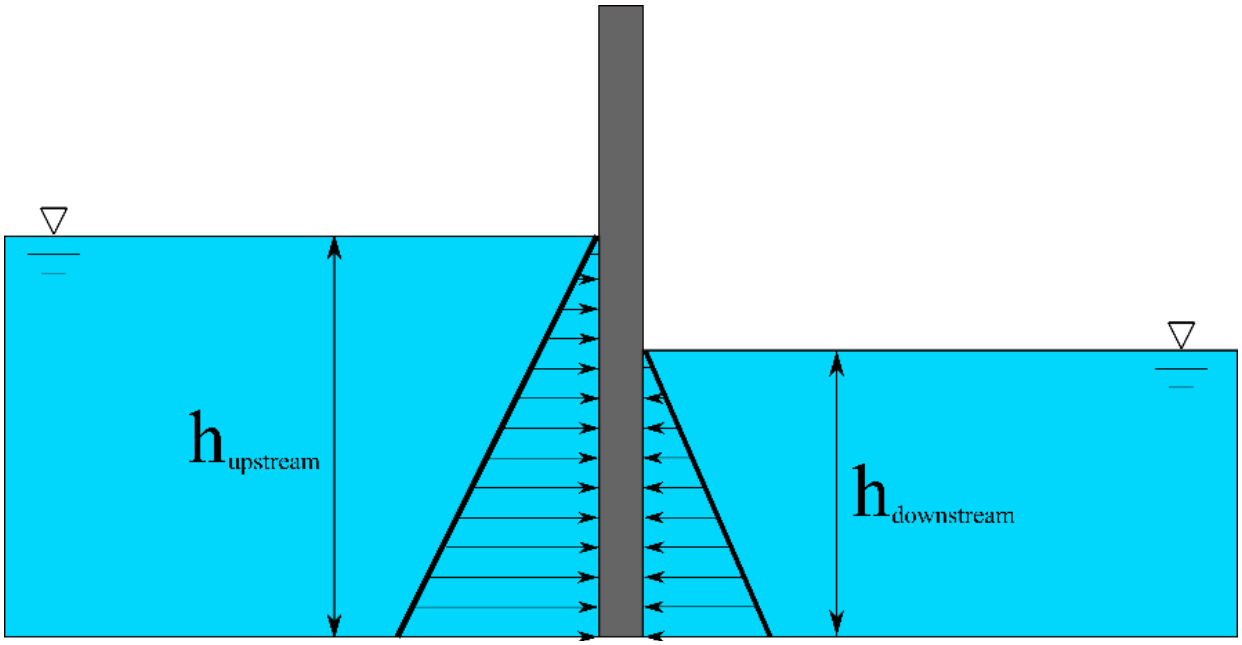


Figure 5: Hydrostatic loading on miter gates

Figure 5 shows the upstream and downstream hydrostatic loading that the miter gate experiences. Also, the environmental temperature, which will add thermal strain effects, is defined as the values recorded by thermometers located at the actual Greenup gate.

3.2. BNN ARCHITECTURE AND RESULTS

The architecture used in this paper contains 2 layers with 50 neurons in its hidden layer and biases at each hidden neuron. A parametric studied using a 2-layer network was carried using an ANN model with different numbers of neurons in its hidden layer. It was found that using 50 neurons yielded a higher testing accuracy than the other architectures used. Ten different (10, 20, …., 90 and 100 neurons) architectures were considered to arrive to this architecture, which each architecture took from 2 to 5 minutes for training and testing using a single CPU processor. Regularization was considered to penalize architectures with more parameters (i.e. neurons). A 2-layer architecture (i.e., 1 hidden layer) was selected because it can learn any continuous mathematical function [40]. Further studies could be carried using a deeper neural network architecture, but the simplest universal approximator was considered most desirable.

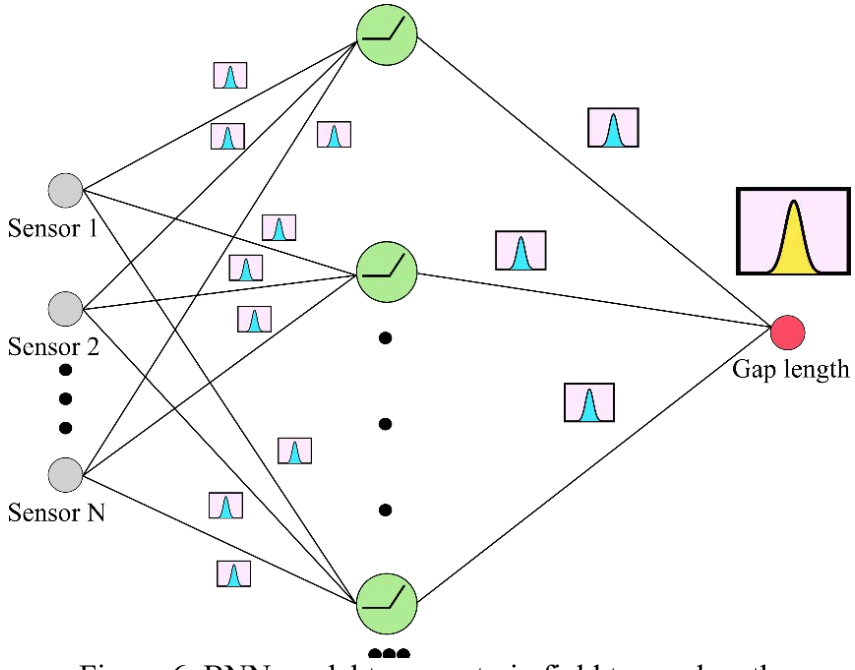


Figure 6: BNN model to map strain field to gap length

The hidden layers use activation functions (e.g. sigmoid) to make the BNN learn any nonlinearity between the strain values and the gap length. For the output layer, the softplus activation function was employed to impose physical constraints (e.g., the gap length cannot be negative) in the BNN. As expressed mathematically earlier, a BNN is a neural network with a prior in its weight and biases as shown in Figure 6.

3.2.1. TRAINING DATA AND TESTING RESULTS

The gap length is assumed to be a random number between zero and 180 in. [39] under random loading scenarios defined by two normal distributions for upstream (h_{up}) and downstream (h_{down}) hydrostatic pressure as shown in Table 1. For training and testing data, 3000 data points were obtained using the ABAQUS FE model of the Greenup gate by varying the value of each random variable for training (2000 for training) and testing purposes. This data took one week using a 4-cpu desktop to be generated. Thermal effects are also considering. The temperature (T_{surf} & T_{uw}) are defined as a normal distributions with mean at a temperature, *T*, which is defined as a random number based on the lowest (T_{min}) and highest (T_{max}) temperature values recorded by thermometers (underwater and surface) in an actual miter gate at different times of the year. Figure 7 shows how these distributions are propagated to 46 strain values, whose location are based on what is installed in the Greenup gate. Note that the parameters presented in Table 1 are the input variables to the FE model, whose output is the strain, as would be detected by an installed SHM system.

Table 1: Random variables used to generate training/testing data

Parameter	**Distribution**	**Unit**
Gap length	$Gap \sim Uniform\ (0,\ 180)$	Inches
h_{down}	$h_{down} \sim Normal(\mu = 168,\ \sigma = 20)$	Inches
h_{up}	$h_{up} \sim Normal(\mu = 552,\ \sigma = 10)$	Inches
T	$T \sim Uniform\ (T_{min} = 29.4, T_{max} = 47.4)$	Celsius
T_{surf}	$T_{surf} \sim Normal\ (T,\ \sigma_{surf} = 10.3)$	Celsius
T_{uw}	$T_{uw} \sim Normal(T, \sigma_{uw} = 5.37)$	Celsius

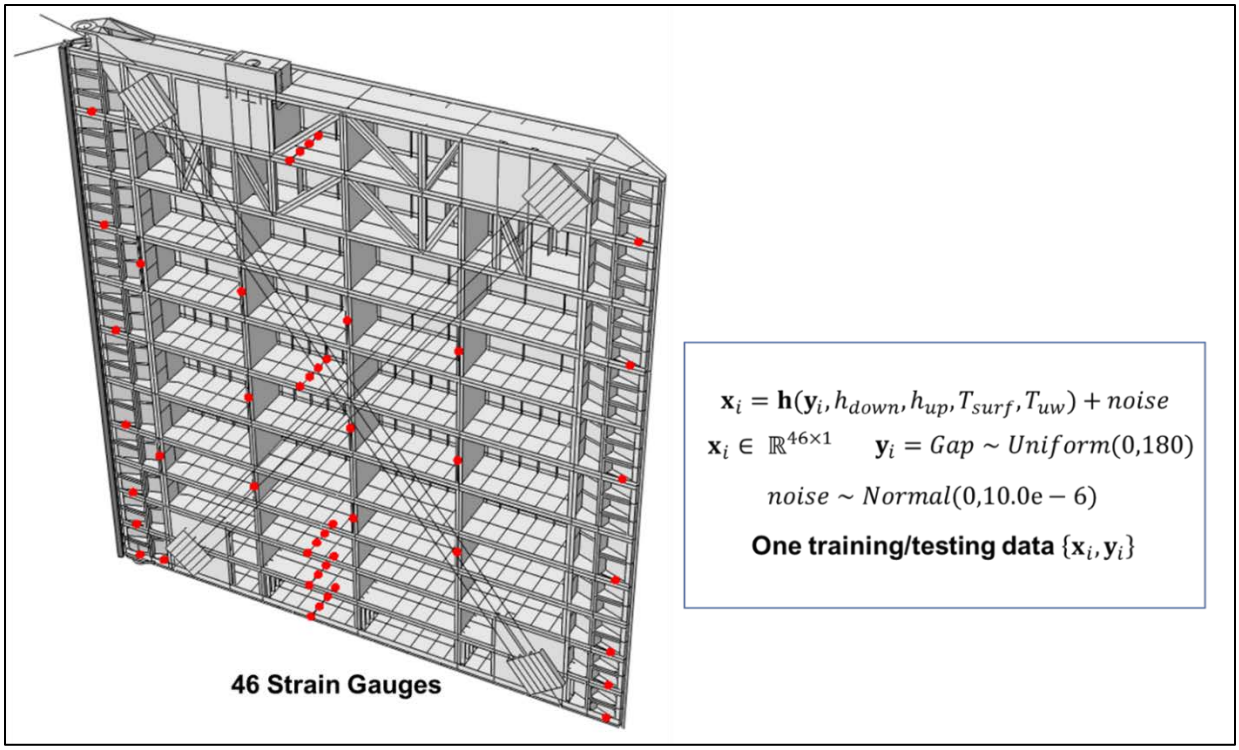


Figure 7: Training and testing data generation

The posterior predictive distribution is calculated using Eq. (6), and 2000 samples from this distribution at each testing data point are shown in Figure 8. It takes around 20 minutes to train this BNN model.

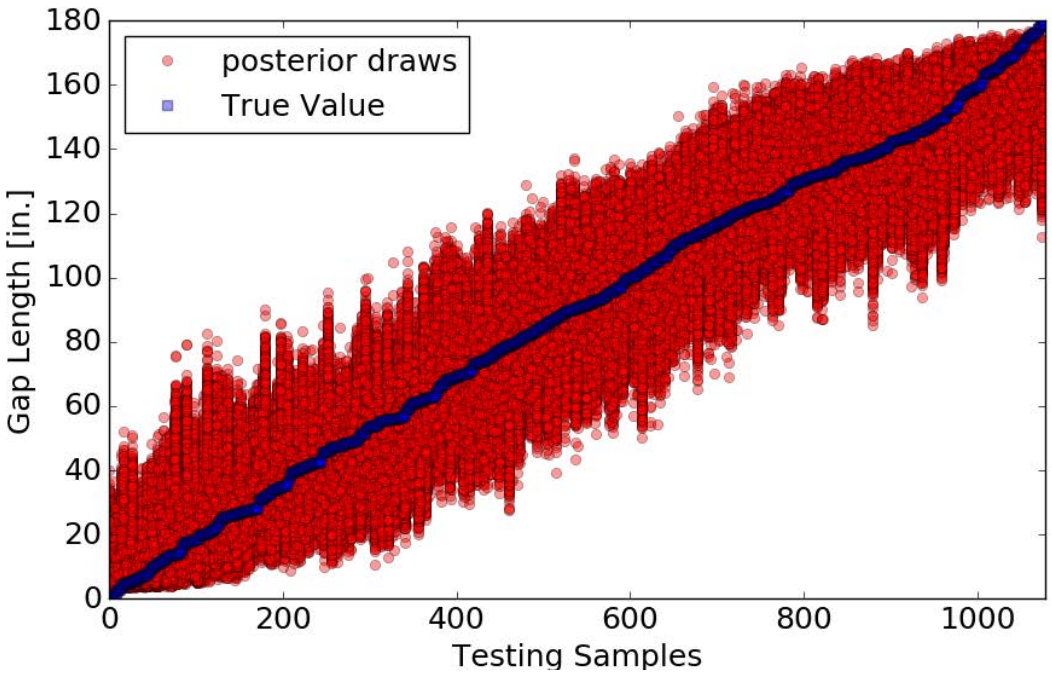


Figure 8: Posterior Distribution of gap length using 1000+ testing samples

The median value of the posterior distribution, as shown in Figure 9, may be used to calculate the mean square error (MSE) to compare the predicted gap length with the true gap length testing value as shown in Table 2, in order to evaluate a set of point predictions.

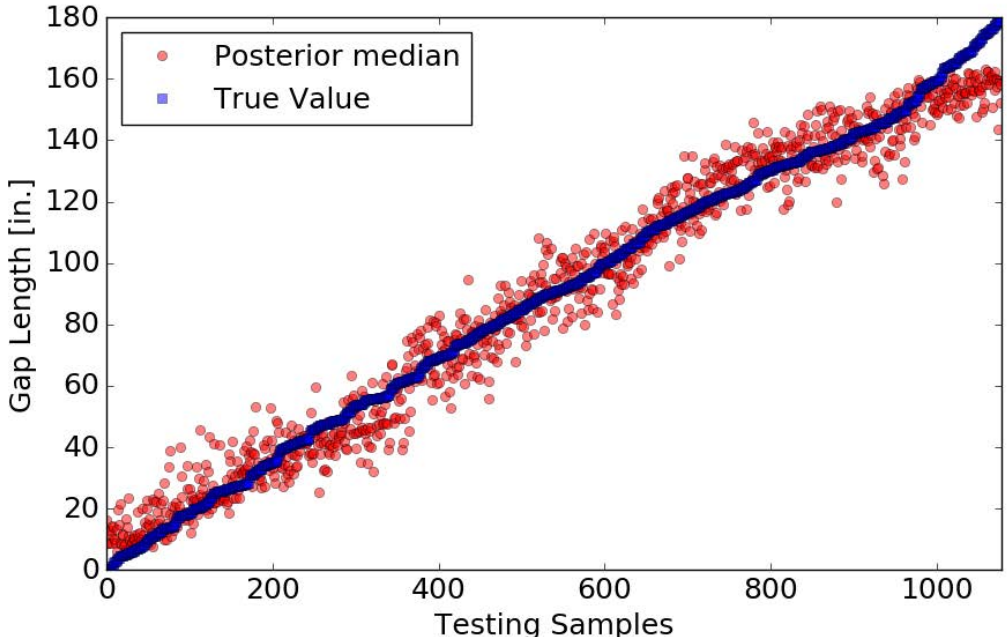


Figure 9: Median values of prediction of gap length

The median is a more useful metric to summarize central tendencies in cases where the distributions may potentially be highly skewed or asymmetric. Moreover, the median is more robust than the mean to outliers that can bias the central tendency.

Table 2: Testing accuracy

BNN Accuracy	RMSE (in.)
Median	8.34
Mean	8.39

One other advantage of using a BNN over an ANN is that the gap length using a given set of strain measurements, $\mathbf{x}_i$, may be expressed as a distribution rather than a single point estimate. Therefore, the probability of exceeding certain critical gap length may be calculated to facilitate the decision-making process for preventive maintenance actions. Figure 10 shows a representative prediction distribution at 4 different testing points, which qualitatively appear somewhat Gaussian for each of these cases. Note that in some other applications, the posterior predictive distribution, obtained from VI, can be multi-modal, which makes order statistics (important for decision-making) less interpretable.

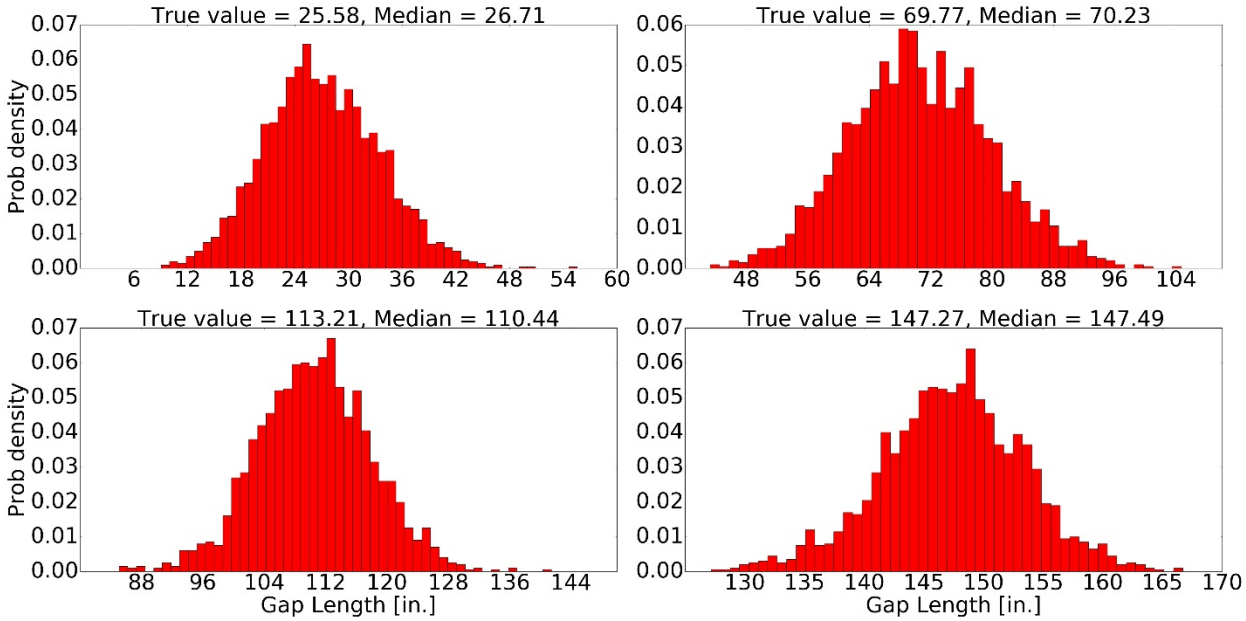


Figure 10: Posterior distribution of gap length using 4 different test samples

In general, a BNN surrogate that is trained by data generated by computer simulation may not be able to capture the behavior of the real structure due to modelling error. To ensure that a trained BNN is reliable for SHM, the modelling error between the FE model and the real world should be accounted for and modeled. In this paper, modeling error was introduced to the computer simulations by varying the hydrostatic and temperature load and treating these loads as unknown quantities. Alternatively, two surrogate models could be used: one to learn the function that defines the FE simulations, and one to learn the modeling error between the FE simulation and the real world. As new data arrive from continuous monitoring, both surrogate models should be updated regularly. Particularly, the surrogate that fits the modelling error as modelling error extrapolation may be not be very accurate if it not updated regularly.

4. VALUE OF IMPLEMENTING SHM USING BNN SURROGATE MODEL

Recently, theoretical and applied approaches to quantify the value of deploying a structural health monitoring have been studied by researchers such as Konakli and Faber [41] and Thons [42]. These and related studies are among the first to tie decisions that SHM informs to decision costs; this is the critical step that connects SHM to the business case for investing in and deploying an SHM system. Within the context of using the uncertainty-quantified BNN SHM "system" developed in the first part of this paper, the BNN outputs will be matched with functions representing the consequence costs of (good and bad) decisions. This framework will be used to compare the relative merits of the BNN SHM approach to current engineering inspection data to arrive at conclusions regarding the relative "value" of such an approach.

4.1. OPTIMAL DECISIONS USING INSPECTION DATA ONLY

In the specific use case presented in this paper, the USACE Asset Management team oversees the Operational Condition Assessment (OCA) process to assess structural component deficiencies by giving a category rating based on a condition and performance criteria. These ratings are performed by an inspector, who base the evaluation on engineering knowledge and information of preexisting inspections Figure 11 summarizes the OCA criteria currently used by USACE.

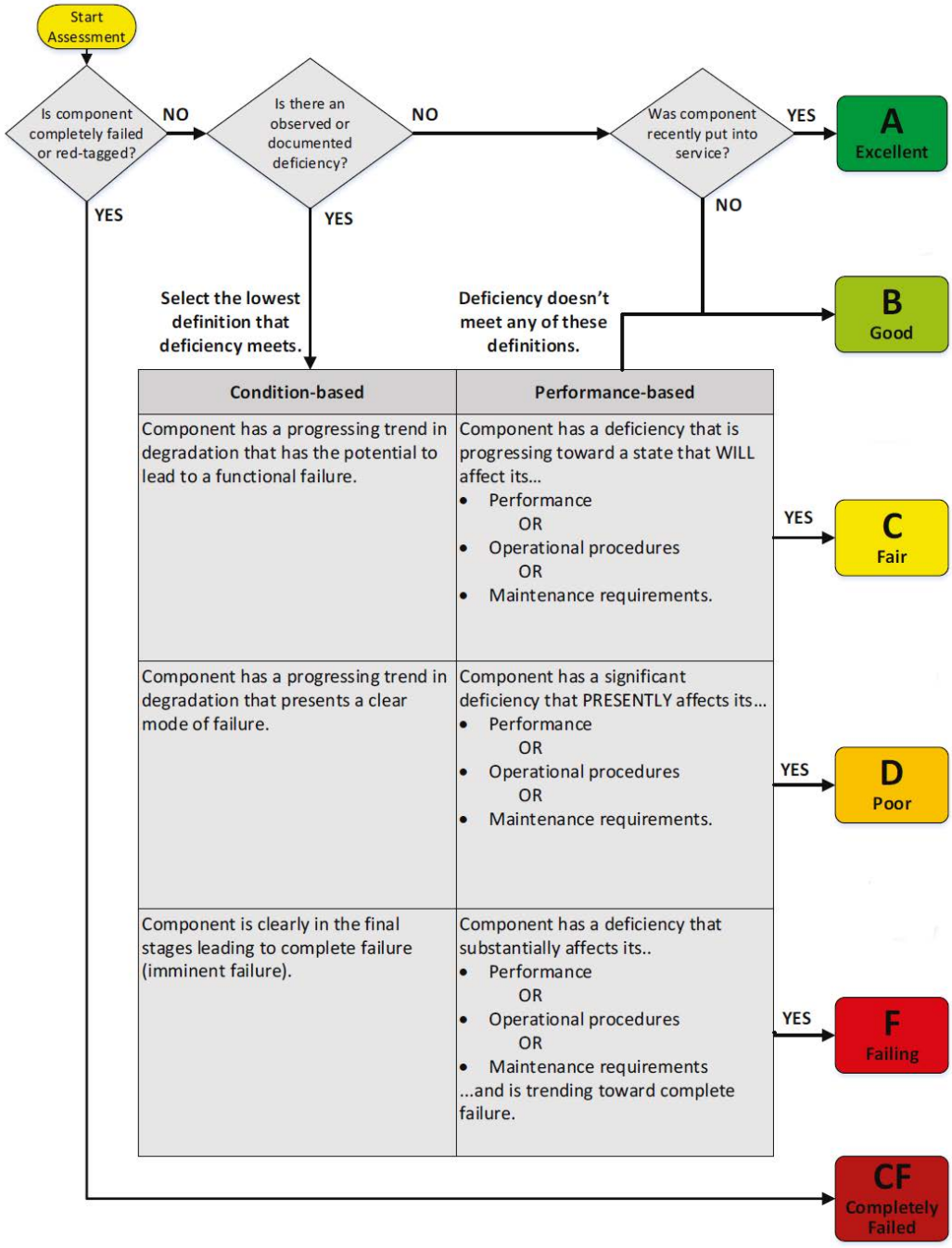
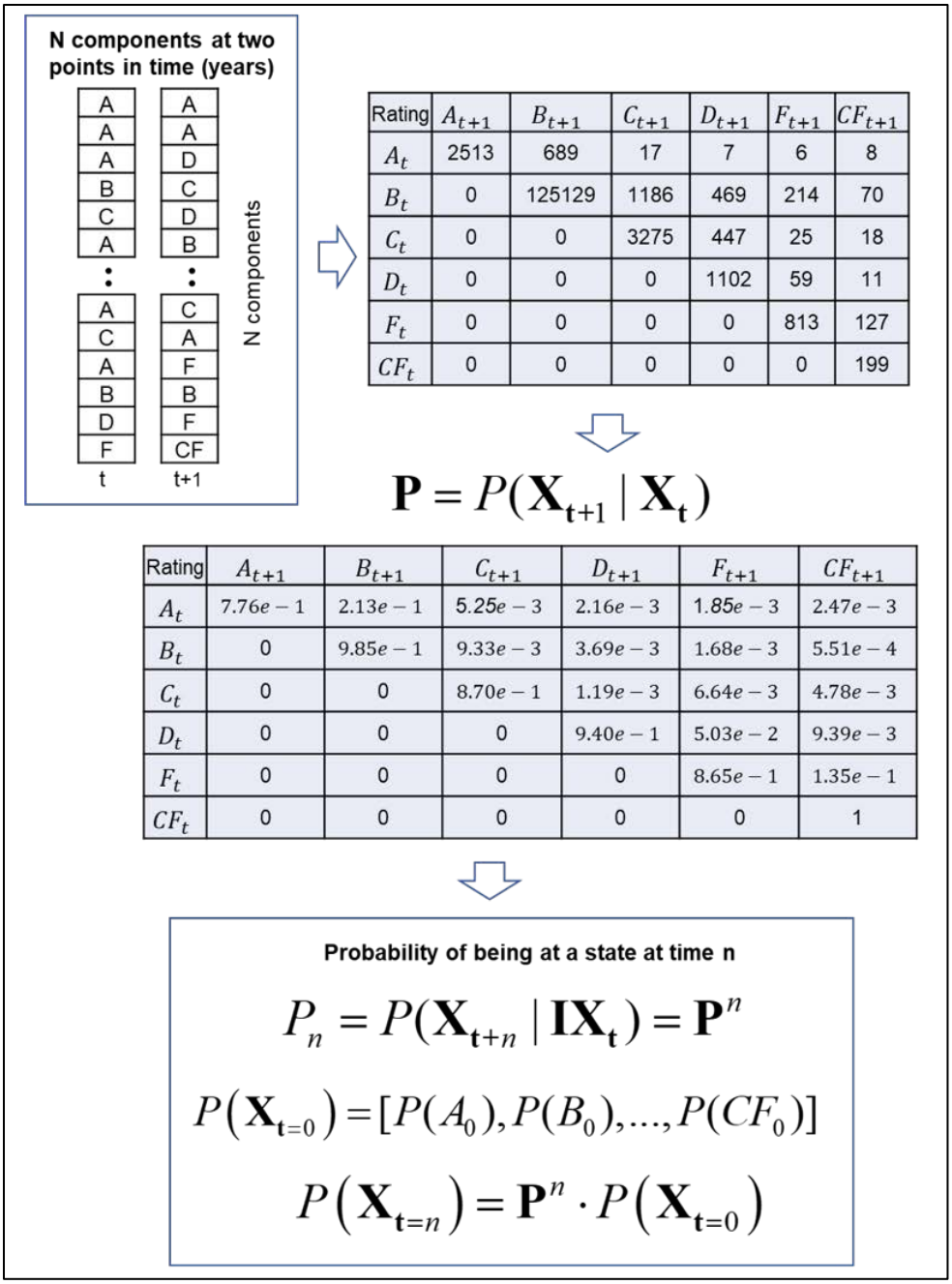


Figure 11: Current OCA rating criteria [43]

4.1.1. TRANSITION PROBABILITY DERIVATION

A transition matrix is defined as a square matrix with nonnegative values that represents how some process "transitions" from one state to the next. Based on an OCA database, the number of times that a component transitioned from one rating category (by engineering inspection expert judgment) to another in a given year was determined to generate a "condition" transition matrix. Thus, in this application, each value in the transition matrix represents a probability, and the sum of each row equals unity. Only the upper triangle components were considered to simulate component deterioration; the lower triangle would represent improvements or repairs, and for the purposes of this analysis, they were ignored. This "condition" transition matrix was found by normalizing the counts in each row as shown in Figure 12.



Rating	A_{t+1}	B_{t+1}	C_{t+1}	D_{t+1}	F_{t+1}	CF_{t+1}
A_t	2513	689	17	7	6	8
B_t	0	125129	1186	469	214	70
C_t	0	0	3275	447	25	18
D_t	0	0	0	1102	59	11
F_t	0	0	0	0	813	127
CF_t	0	0	0	0	0	199

$$\mathbf{P} = P(\mathbf{X_{t+1}} \mid \mathbf{X_t})$$

Rating	A_{t+1}	B_{t+1}	C_{t+1}	D_{t+1}	F_{t+1}	CF_{t+1}
A_t	$7.76e-1$	$2.13e-1$	$5.25e-3$	$2.16e-3$	$1.85e-3$	$2.47e-3$
B_t	0	$9.85e-1$	$9.33e-3$	$3.69e-3$	$1.68e-3$	$5.51e-4$
C_t	0	0	$8.70e-1$	$1.19e-3$	$6.64e-3$	$4.78e-3$
D_t	0	0	0	$9.40e-1$	$5.03e-2$	$9.39e-3$
F_t	0	0	0	0	$8.65e-1$	$1.35e-1$
CF_t	0	0	0	0	0	1

$$P_n = P(\mathbf{X_{t+n}} \mid \mathbf{IX_t}) = \mathbf{P}^n$$

$$P\left(\mathbf{X_{t=0}}\right) = [P(A_0), P(B_0), \ldots, P(CF_0)]$$

$$P\left(\mathbf{X_{t=n}}\right) = \mathbf{P}^n \cdot P\left(\mathbf{X_{t=0}}\right)$$

Figure 12: Deriving 1-step (1 year) transition matrix for quoin block components

Transition matrices, known also as stochastic matrices, have been broadly used in different fields such as probability theory, control, economics, and meteorology [44–46].

4.1.2. FAILURE RATE OF COMPONENT AND COST FUNCTION

A degradation model built from the transition matrix is used to generate a failure cumulative mass function, which can approximate the unreliability function, as described in detail in [47]. Figure 13

shows the unreliability function of the quoin block component with the component age in years as the random variable.

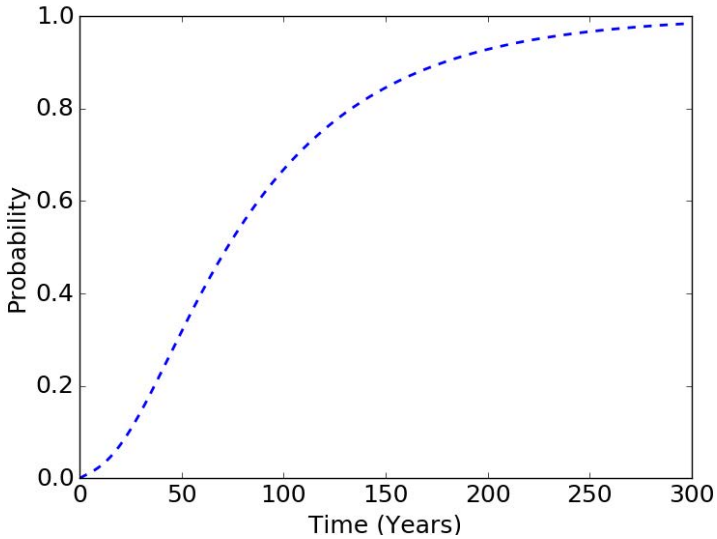


Figure 13: Unreliability function of quoin block component

Eq. (17) shows a cost function proposed by [48] to find the cost per unit of time (CPUT) of performing preventive maintenance at a time *t* in years.

$$CPUT(t) = \frac{C_P[1-F(t)] + C_U[F(t)]}{\int_0^t [1-F(s)]\,ds}, \tag{17}$$

where $F(t)$ is the unreliability function, C_P is the preventive action cost, and C_U is the unplanned action cost. The unreliability function presented in Figure 13 was used with Eq. (17) to find the CPUT for different values of *t* as shown in Figure 14. This plot suggests that the optimal time to perform preventive maintenance is every 48 years when only considering the deterioration of quoin blocks and the data available from OCA inspections and the cost ratio is equal to 5. In other words, the "model" of the engineering inspection via the OCA database proposes a cost-minimized optimal inspection time of 48 years. The corresponding cost ratio (i.e. C_U/C_P) values depend on the structure and site. The values C_P can be defined as

$$C_P = \begin{matrix}\text{scheduled}\\ \text{maintenance}\\ \text{cost}\end{matrix} + t_P * \begin{matrix}\text{daily economic}\\ \text{cost due to}\\ \text{downtime}\end{matrix}, \tag{18}$$

where t_P is the downtime (in days) that takes to perform normal maintenance. The maintenance cost associated is definitely lower when this is planned ahead. The value of C_U is correspondingly

$$C_U = \begin{matrix}\text{maintenance}\\ \text{cost due to}\\ \text{failure}\end{matrix} + t_U * \begin{matrix}\text{daily economic}\\ \text{cost due to}\\ \text{downtime}\end{matrix}, \quad (19)$$

where t_U is the downtime (in days) required to replace the component that failed, measured from the time of failure. For miter gates, $t_U >> t_P$ essentially due to two reasons: 1) availability to start maintenance in a short period of time after failure occurrence and 2) maintenance takes longer when a component fails because it can affect other components or systems. The maintenance cost associated is definitely higher when maintenance needs to start as soon as possible. Therefore, the unplanned cost is higher than the preventive action cost (i.e. $C_U / C_P >> 1$).

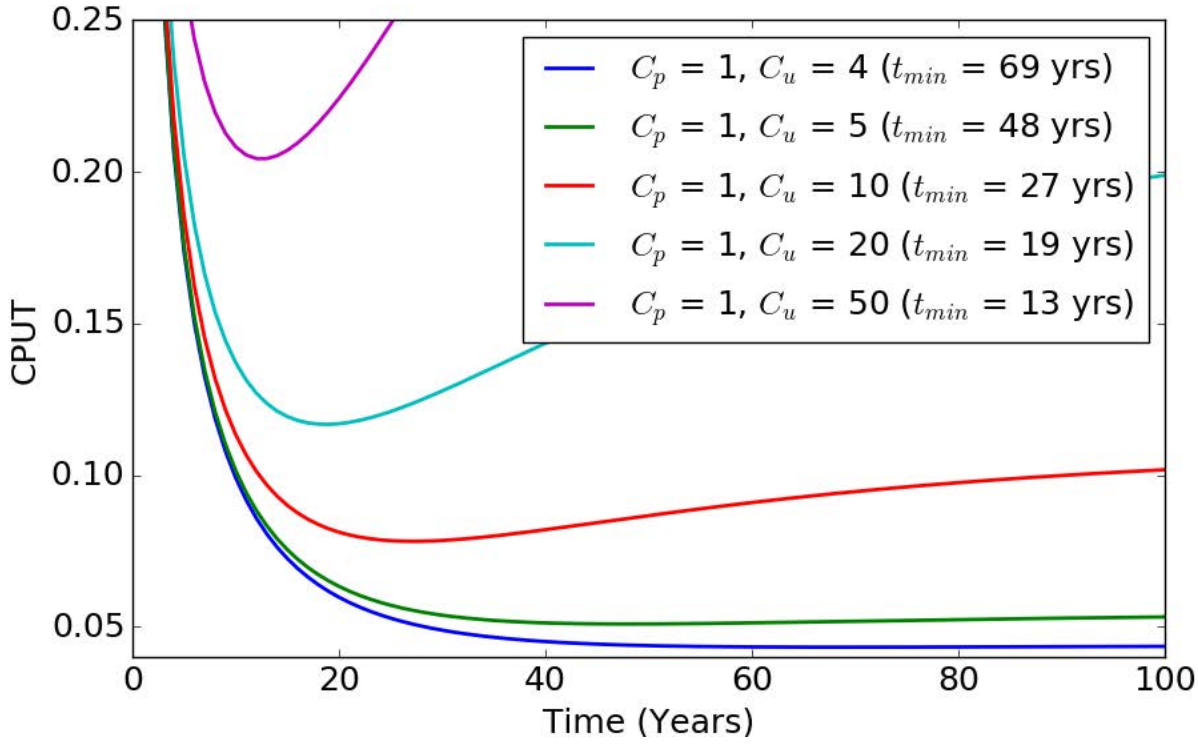


Figure 14: Cost per unit of time as a function of component age

4.2. OPTIMAL DECISIONS USING BNN SURROGATE MODEL AND VALUE OF SHM APPLICATION

Earlier, a cost function was defined, and the optimal maintenance time that corresponded to the minimum CPUT was calculated using the information provided by a model of the visual inspection process over time, presented in section 4.1 and reviewed in more detail in [47]. Now, the BNN SHM approach will be used instead to build the unreliability function in order to compare the relative "value" of using the BNN SHM approach to the visual inspection approach for monitoring. To do that, the results shown in Figure 8 can be used to calculate the probability of exceedance of a certain gap length threshold as shown in Figure 15. This figure shows empirical cumulative mass

functions for five different failure thresholds (i.e., gap lengths that correspond to criticality). Next, a mapping between the true gap length value and the component age can be used to find the corresponding unreliability functions. Eq. (11) can be applied to find the corresponding gap length value that minimizes the cost function as shown in Figure 16. Finally, the mapping from the true gap length value to the component age is used to find the optimal maintenance time as shown in Table 3. For these realizations, different cost ratios (i.e., values of C_P and C_U) and different failure threshold were used. It is important to note that specific results obviously depend on the choice of these values, but that the methodology shown in this section is independent of the actual values of the cost ratios and the failure threshold. After comparing different values of maintenance costs for miter gates at a specific site, the authors suggest that the corresponding cost ratio (i.e. C_U/C_P) is close to 5 based on communications with USACE personnel [49].

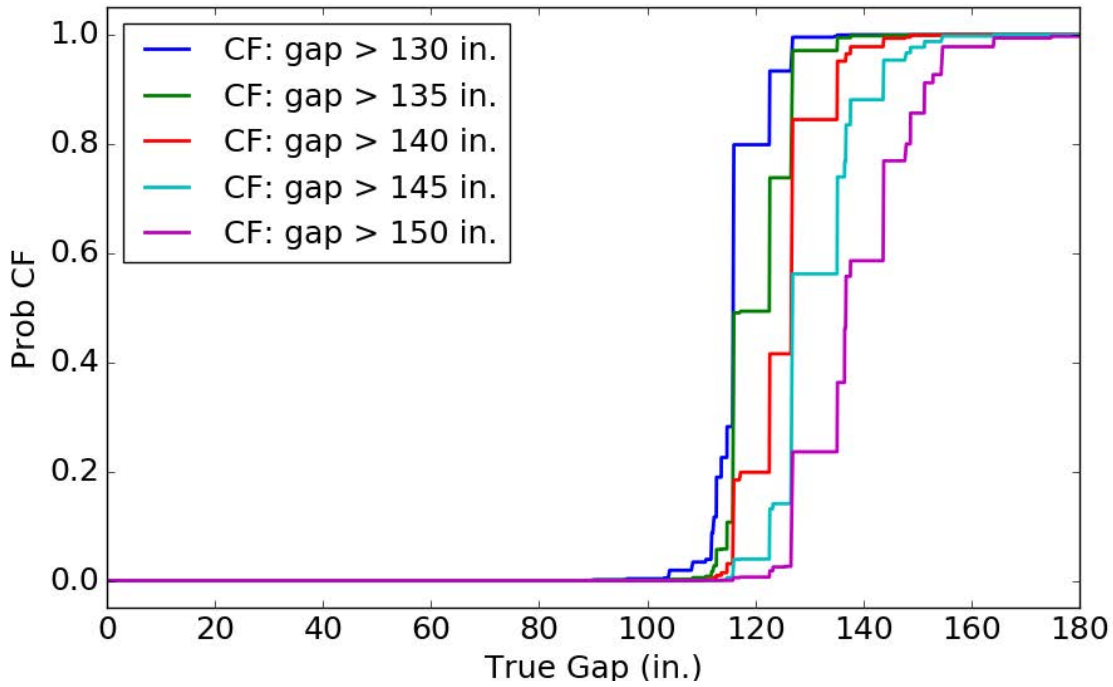


Figure 15: Empirical (Failure) cumulative mass function with $C_P = 1$ and C_U=5

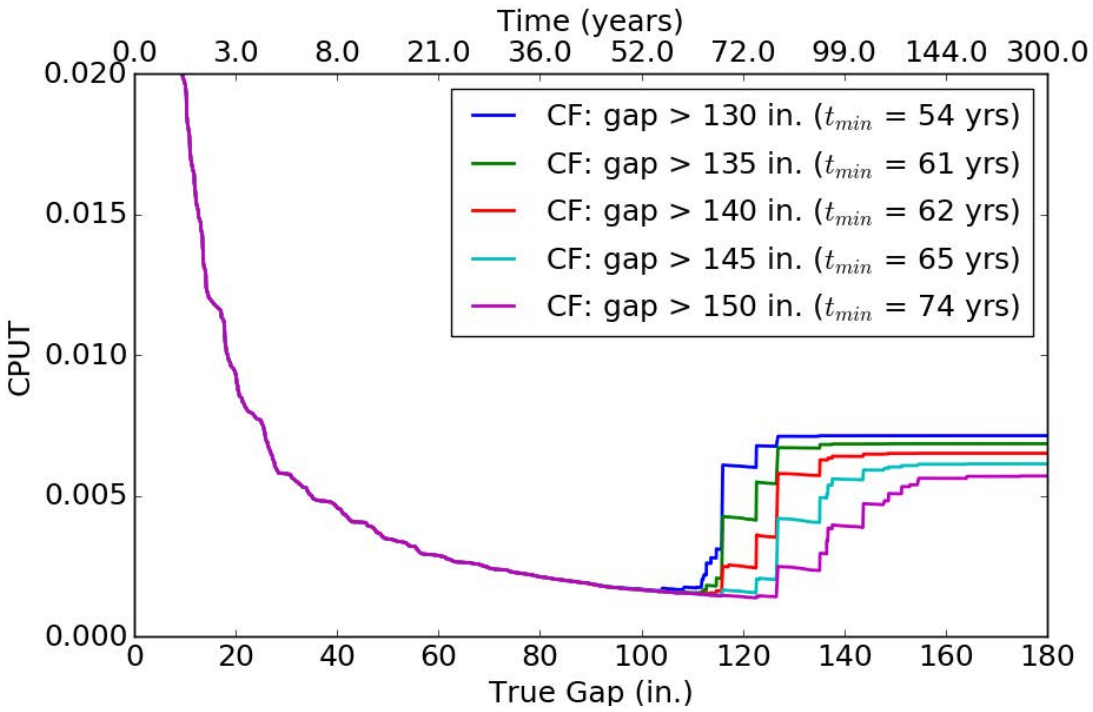


Figure 16: Cost per unit of time as a function of true testing gap length with $C_P = 1$ and C_U=5

Table 3: Optimal maintenance time using BNN model with $C_P = 1$ and C_U=5

Critical gap length (in.)	Optimal time (years)	Cost reduction (%)
130+	54	11.1
135+	61	21.3
140+	62	22.6
145+	65	26.2
150+	74	35.1

Different miter gate sites may have different values for the cost ratio (i.e. C_U/C_P); Figure 17 shows the variation in CPUT when a different value of cost ratio is used. For the realizations in this figure, the critical gap length threshold is assumed to be equal to 140 in. This figure shows less sensitivity to the cost ratio than Figure 14. The main reason why it is so is because there is less uncertainty when using the BNN SHM model. Of course, in an absolute judgment sense, it is important to note that the BNN model assumes that the training data generated from the FEM model is ground truth. As with any such model, its representative predictive value is only as good as its validation with regard to the real structure that it is modeling. In this case, the FEM was previously validated to the Greenup miter gate in the undamaged condition, as mentioned earlier, but the modeling of the damage itself couldn't be validated on actual data from the gate in a known damaged condition, so modeling bias error in the damage state could creep into the process, as was discussed in section 3. That doesn't change or otherwise invalidate the demonstration of the proposed approach or its utility, but rather it provides caution on interpreting the specific results for this case beyond demonstration of the overall approach.

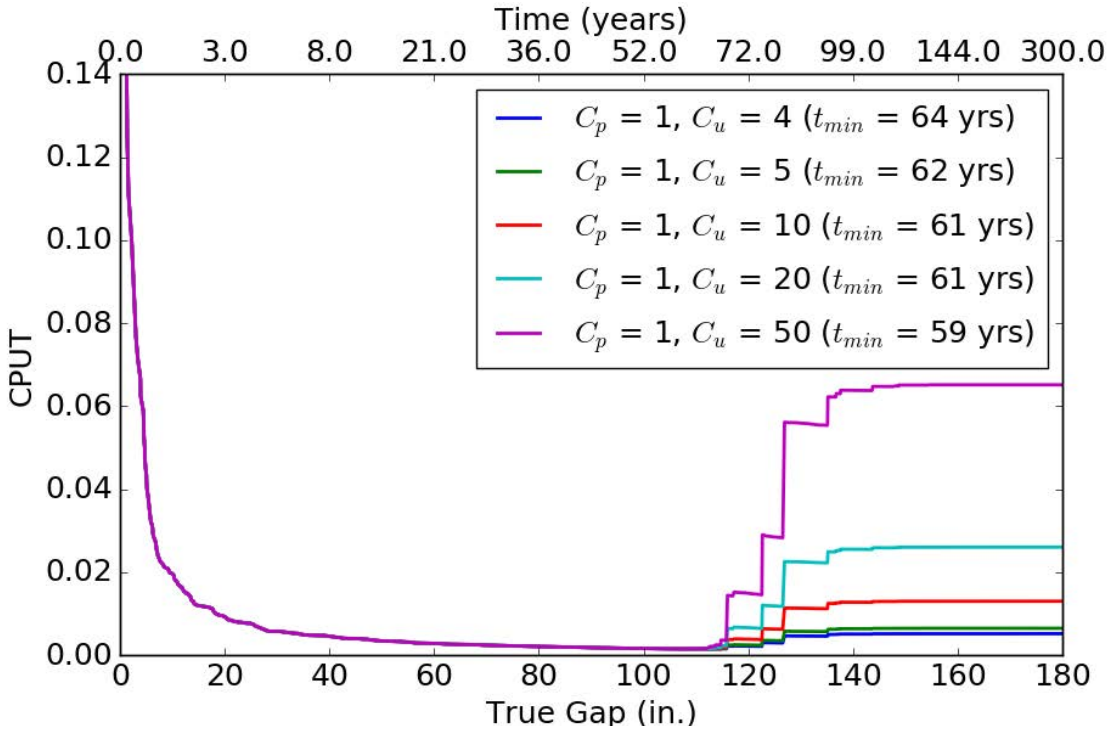


Figure 17: Cost per unit of time as a function of true testing gap length

5. CONCLUSIONS AND DISCUSSION

The added value of using SHM in miter gates is evaluated in term of maintenance cost savings. A cost function is presented to improve prioritization of maintenance events of components of miter gates by evaluating the performance of a trained BNN model. This model is trained to assess the condition of quoin blocks in miter gates with a FE model due to such highly-limited data availability. In this work, a degradation model based on real inspection data of miter gates is used to simulate the damage evolution. A SHM workflow is set up to allows further optimization in term of cost savings within a civil structural monitoring application. As presented in this paper, continuous monitoring via this BNN SHM "system" can lead to more economical decisions regarding maintenance policies than only using the data from visual inspection (e.g. OCA ratings). From the results shown in the previous sections, there is an 11.1% to 35.1% of maintenance cost reduction when the OCA ratings are used with a surrogate model based on a physical based model.

It is important to know that the degradation modes presented in this paper was built from real inspection data. However, this data can still be bias to human error or insufficient information due to the difficulty to assess OCA ratings when a component is underwater, and it is not visibly available. Other degradation models can be considered when a larger historical data set is available. This paper only focuses on the degradation of a single component. Further analysis can be carried out by considering more critical components (e.g. cracks in pintle, corrosion in the gate, etc.). Also, there are sources of uncertainty that need to be further analyzed that will lead to changes in the optimal maintenance time, such as measurement uncertainties or model uncertainties from both the BNN and FE model; the latter of these could be quantified via a sensitivity analysis of all the parameters in the FE and BNN models. Another potentially fruitful avenue for improvement is

consideration of how many strain sensors are used and where they are placed. Different such sensor designs could lead to different SHM assessment statistical performance, which in turn affects decision costs, and the sensor design itself directly influences procurement, installation, and sensor maintenance costs. Both of these could compete in a cost-minimized formulation.

ACKNOWLEDGEMENTS

Funding for this work was provided by the United States Army Corps of Engineers through the U.S. Army Engineer Research and Development Center Research Cooperative Agreement W912HZ-17-2-0024.

REFERENCES

1. Stoffel M, Bamer F, Markert B. Artificial neural networks and intelligent finite elements in non-linear structural mechanics. *Thin-Walled Struct* 2018; 131: 102–106.
2. Adeli H, Yeh C. Perceptron Learning in Engineering Design. *Comput Civ Infrastruct Eng* 1989; 4: 247–256.
3. Hajela P, Berke L. Neurobiological computational models in structural analysis and design. *Comput Struct* 1991; 41: 657–667.
4. Theocaris PS, Panagiotopoulos PD. Neural networks for computing in fracture mechanics. Methods and prospects of applications. *Comput Methods Appl Mech Eng* 1993; 106: 213–228.
5. Wu X, Ghaboussi J, Garrett JH. Use of neural networks in detection of structural damage. *Comput Struct* 1992; 42: 649–659.
6. Feng MQ, Bahng EY. Damage Assessment of Jacketed RC Columns Using Vibration Tests. *J Struct Eng* 1999; 125: 265–271.
7. Elkordy MF, Chang KC, Lee GC. A Structural Damage Neural Network Monitoring System. *Comput Civ Infrastruct Eng* 1994; 9: 83–96.
8. Lam H-F, Yuen K-V, Beck JL. Structural Health Monitoring via Measured Ritz Vectors Utilizing Artificial Neural Networks. *Comput Civ Infrastruct Eng* 2006; 21: 232–241.
9. Mikami I, Tanaka S, Hiwatashi T. Neural Network System for Reasoning Residual Axial Forces of High-Strength Bolts in Steel Bridges. *Comput Civ Infrastruct Eng* 1998; 13: 237–246.
10. Xu H, Humar J. Damage Detection in a Girder Bridge by Artificial Neural Network Technique. *Comput Civ Infrastruct Eng* 2006; 21: 450–464.
11. Shu J, Zhang Z, Gonzalez I, et al. The application of a damage detection method using Artificial Neural Network and train-induced vibrations on a simplified railway bridge model. *Eng Struct* 2013; 52: 408–421.
12. Masri SF, Nakamura M, Chassiakos AG, et al. Neural Network Approach to Detection of Changes in Structural Parameters. *J Eng Mech* 1996; 122: 350–360.
13. Kim S-H, Yoon C, Kim B-J. Structural Monitoring System Based on Sensitivity Analysis and a Neural Network. *Comput Civ Infrastruct Eng* 2000; 15: 189–195.
14. Wang N, Zhao Q, Li S, et al. Damage Classification for Masonry Historic Structures Using Convolutional Neural Networks Based on Still Images. *Comput Civ Infrastruct Eng* 2018; 33: 1073–1089.
15. Waszczyszyn Z, Ziemiański L. Neural Networks in the Identification Analysis of

Structural Mechanics Problems. In: *Parameter Identification of Materials and Structures*. Vienna: Springer Vienna, pp. 265–340.
16. Adeli H. Neural Networks in Civil Engineering: 1989–2000. *Comput Civ Infrastruct Eng* 2001; 16: 126–142.
17. Rocchetta R, Broggi M, Huchet Q, et al. On-line Bayesian model updating for structural health monitoring. *Mech Syst Signal Process* 2018; 103: 174–195.
18. Tan ZX, Thambiratnam DP, Chan THT, et al. Detecting damage in steel beams using modal strain energy based damage index and Artificial Neural Network. *Eng Fail Anal* 2017; 79: 253–262.
19. Ruder S. An overview of gradient descent optimization algorithms. 2016; 1–14.
20. Parno M, O'Connor D, Smith M. High dimensional inference for the structural health monitoring of lock gates. 2018; 1–29.
21. Liu H, Ong Y-S, Shen X, et al. When Gaussian Process Meets Big Data: A Review of Scalable GPs. 2018; 1–20.
22. Shi J, Khan ME, Zhu J. Scalable Training of Inference Networks for Gaussian-Process Models, http://arxiv.org/abs/1905.10969 (2019).
23. Flam-shepherd D, Requeima J, Duvenaud D. Mapping Gaussian Process Priors to Bayesian Neural Networks. In: *31st Conference on Neural Information Processing Systems*. Long Beach, CA, USA, 2017, pp. 1–8.
24. Yin T, Zhu H. Probabilistic Damage Detection of a Steel Truss Bridge Model by Optimally Designed Bayesian Neural Network. *Sensors* 2018; 18: 3371.
25. Chua CG, Goh ATC. Estimating wall deflections in deep excavations using Bayesian neural networks. *Tunn Undergr Sp Technol* 2005; 20: 400–409.
26. Arangio S, Bontempi F. Structural health monitoring of a cable-stayed bridge with Bayesian neural networks. *Struct Infrastruct Eng* 2015; 11: 575–587.
27. Todd MD, Flynn EB. A Bayesian Experimental Design Approach for Structural Health Monitoring. In: *14th International Symposium on Dynamic Problems of Mechanics (DINAME 2011)*. Sao Sebastio, Brazil, 2011.
28. Neal RM. *Bayesian Learning for Neural Networks*. New York, NY: Springer New York. Epub ahead of print 1996. DOI: 10.1007/978-1-4612-0745-0.
29. Morzfeld M, Tong XT, Marzouk YM. Localization for MCMC: sampling high-dimensional posterior distributions with local structure. *J Comput Phys* 2017; 380: 1–28.
30. Geman S, Geman D. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Trans Pattern Anal Mach Intell* 1984; PAMI-6: 721–741.
31. Sun S, Zhang G, Shi J, et al. Functional Variational Bayesian Neural Networks. 2019; 1–22.
32. Blei DM, Kucukelbir A, McAuliffe JD. Variational Inference: A Review for Statisticians. *J Am Stat Assoc* 2017; 112: 859–877.
33. Ostwald D. Variational Inference. In: *Probabilistic models for functional neuroimaging*, pp. 163–171.
34. Tran D, Hoffman MD, Saurous RA, et al. Deep Probabilistic Programming. 2017; 1–18.
35. Abadi M, Barham P, Chen J, et al. TensorFlow: A system for large-scale machine learning, http://arxiv.org/abs/1605.08695 (2016).
36. U.S. Army Corps of Engineers Headquarters. Navigation, http://www.usace.army.mil/Missions/CivilWorks/Navigation.aspx (2018, accessed 1 August 2018).

37. Foltz SD. *Investigation of Mechanical Breakdowns Leading to Lock Closures*. Champaign, IL, 2017.
38. U.S. Army Corps of Engineers Headquarters. SMART GATE, https://www.erdc.usace.army.mil/Media/Fact-Sheets/Fact-Sheet-Article-View/Article/476668/smart-gate/ (accessed 1 August 2018).
39. Eick BA, Treece ZR, Spencer BF, et al. Automated damage detection in miter gates of navigation locks. *Struct Control Heal Monit* 2018; 25: 1–18.
40. Csáji B. *Approximation with artificial neural networks*. Eötvös Loránd University, Hungary. Epub ahead of print 2001. DOI: 10.1.1.101.2647.
41. Konakli K, Faber MH. Value of Information Analysis in Structural Safety. In: *Vulnerability, Uncertainty, and Risk*. Reston, VA: American Society of Civil Engineers, pp. 1605–1614.
42. Thöns S. On the Value of Monitoring Information for the Structural Integrity and Risk Management. *Comput Civ Infrastruct Eng* 2018; 33: 79–94.
43. Allen JP, Foltz SD, Werth MH. *Sustainment management system dams inspection module: Department of Defense dams inventory and inspection template*, https://erdc-library.erdc.dren.mil/xmlui/handle/11681/27351 (2018).
44. Degroot MH. Reaching a Consensus. *J Am Stat Assoc* 1974; 69: 118–121.
45. Caldarelli G, Cristelli M, Gabrielli A, et al. A Network Analysis of Countries' Export Flows: Firm Grounds for the Building Blocks of the Economy. *PLoS One* 2012; 7: e47278.
46. Schoof JT, Pryor SC. On the Proper Order of Markov Chain Model for Daily Precipitation Occurrence in the Contiguous United States. *J Appl Meteorol Climatol* 2008; 47: 2477–2486.
47. Vega MA, Madarshahian R, Fillmore TB, et al. Optimal Maintenance Decision for Deteriorating Components in Miter Gates Using Markov Chain Prediction Model. In: *Structural Health Monitoring 2019: Enabling Intelligent Life-cycle Health Management for Industry Internet of Things (IIOT)*. Lancaster, PA: DEStech Publications, Inc., pp. 1471–1478.
48. Barlow R, Hunter L. Optimum Preventive Maintenance Policies. *Oper Res* 1960; 8: 90–100.
49. Schultz MT. Personal communication, Sept. 18, 2018.