Anomaly Detection Using State-Space Models and Reinforcement Learning

Shervin Khazaeli[*], Luong Ha Nguyen, James-A. Goulet
Department of Civil, Geologic and Mining Engineering
École Polytechnique Montréal, CANADA

March 13, 2021

Abstract

The early detection of anomalies associated with changes in the behavior of structures is important for ensuring their serviceability and safety. Identifying anomalies from monitoring data is prone to false and missed alarms due to the uncertain nature of the infrastructure responses' dependency on external factors such as temperature and loading. Existing anomaly detection strategies typically rely on univariate threshold values and disregard the planning horizon in the context of decision making. This paper proposes an anomaly detection framework that combines the interpretability of existing Bayesian dynamic linear models, a particular form of state-space models, with the long-term planning ability of reinforcement learning. The new framework provides (a) reinforcement learning formalism for anomaly detection in Bayesian dynamic linear models, (b) a method for simulating anomalies with respect to its height, duration, and time of occurrence, and (c) a method for quantifying anomaly detectability. The potential of the new framework is demonstrated on monitoring data collected on a bridge in Canada. The results show that the framework is able to detect real anomalies that were known to have occurred, as well as synthetic anomalies.

Keywords: Anomaly detection, reinforcement learning, Bayesian dynamic linear models, state-space models, decision-making, structural health monitoring, false alarm, bridge

1 Introduction

The deterioration of civil infrastructures such as bridges is responsible for important societal expenses. Despite some improvements in infrastructure condition because of rehabilitation, the average age of our infrastructure is increasing, and in many cases, it is approaching the design limit or operational life-cycle [3, 7]. Using sensors enables monitoring the structural condition as well as the occurrence of anomalies; this task is referred to as *structural health monitoring* (SHM). It provides information for decision makers in order to support the operation, maintenance, and the replacement of civil structures. In this paper, we focus on long-term infrastructure monitoring where an *anomaly* is defined as a change in the

[*] Corresponding author: shervin.khazaeli@gmail.com

structural behavior. The objective of the anomaly detection is to identify these changes early while avoiding false alarms. Despite the fact that sensing technologies have become cheaper and broadly available, interpreting *time series* data remains a challenge. Many methods from the field of statistics and machine learning have been developed, in order to detect anomalies in time series data. Although there is not a unified accepted categories for anomaly detection strategies [29], Yu & Sun [41] classified these strategies into statistical-based and machine learning-based methods. Statistical approaches build a parametric [1, 6] or a non-parametric [21, 39] statistical model from the training dataset, and test the newly collected data against the model in order to identify whether the test dataset fit the model or not. Both types of statistical models requires a predefined distribution representing the normal condition, which is not often available for SHM data. Machine learning-based approaches often rely on building a classification or a clustering model [4,11,19,23]. However, when a model is built, it cannot be continuously adapted as new data are collected. In addition, anomalies in infrastructures are rare events, and yet in many cases, these methods require labels for the structural conditions (e.g. normal and abnormal) [4, 5, 8, 12]. Another common limitation of the above-mentioned approaches is that identifying anomalies from the changes in behavior is prone to outliers, especially for methods relying on the hypothesis testing [9]. A common approach to quantify the performance of the anomaly detection method is to use a metric that quantifies the true and false anomaly detections [29]: a true detection consist in correctly identifying an anomaly when it occurs. In contrast, a false detection, i.e., false alarm, occurs when the anomaly detection method incorrectly identifies the presence of an anomaly. The reader is referred to [14] for a detailed review of the available metrics based on the true detection and false alarm rates. In addition, Ahmad et al. [2] proposed the Numenta Anomaly Benchmark (NAB) scoring system in order to account for the early anomaly detections. In general, an effective anomaly detection method is the one that results in metrics indicating a high early true detection rate, while keeping a low false alarm rate. However, these metrics do not account for the anomaly characteristics such as height, duration and time of occurrence. Therefore, the anomaly detectability method cannot be quantified with respect to the nature of the anomaly.

As pointed by Pimentel et al. [29], another class of anomaly detection strategy relies on the *state-space models* (SSM). These models decompose the observations into the superposition of some *hidden* states that evolve through time. A common SSM for anomaly detection that address the limitations of above-mentioned methods is the *Kalman filter* and its generalization with the *Switching Kalman Filter* (SKF) [16]. In the context of SHM, Goulet [17] and Goulet & Koo [18] employed a particular SSM, called Bayesian Dynamic Linear Models (BDLM). A BDLM consists in *empirical models* allowing to decompose structural responses into the superposition of *reversible* and *irreversible* behaviors. The reversible behaviors are due to external effects such as temperature and loading, whereas the irreversible ones are the result of changes in the structure itself. Nguyen & Goulet [25] have introduced an anomaly detection method based on the combination of the SKF and BDLM. The SKF-based BDLM uses the *probability of regime switching* as an indication of a possible anomaly and does not require labeled training data with respect to the condition of the structure. In this regard, BDLM considers two different dynamics for the normal and abnormal regime of the structure, where the switching between the two dynamics is characterized probabilistically. A key advantage of this approach is that these dynamics explicitly quantify normal and abnormal regime kinematics. Therefore, it is possible to evaluate the performance of the anomaly detection method with respect to the anomaly characteristics. Nguyen & Goulet [26] have extended the BDLM method for real-time anomaly detection using Rao-Blackwellized Particle Filtering (RBPF) enabling to identify the probability of the abnormal regime as new data arrive. The main limitation of making decisions solely based on the probability of the abnormal regime is that it is prone to false or missed alarms, especially for low probability events; it is possible that during the decision making, we face situations for which the probability of abnormal regime is low, while the underlying irreversible responses indicate a possible switch between two regimes, and vice versa. Therefore, there is a need for leveraging information from both irreversible responses and the probability of regime switching for robust SHM anomaly detection. Another limitation is that such a decision making process does not incorporate long-term planning considerations. For instance, there can be a situation for which the initial change in probability at a given time is not due to an anomaly, and the decision maker needs to delay the decision making until more data are available.

Reinforcement Learning (RL) [33, 34] is a sub-field of the Artificial Intelligence (AI) enabling the decision maker, called the *agent*, to make decisions by taking *actions* in an *environment*. RL is formulated via Markov decision process (MDP) for which at each time step, the agent interacts with its environment by taking actions that influence the environment's state. Accordingly, the environment gives a feedback in the form of a *reward*, a signal indicating whether the agent takes beneficial actions. The agent continues learning by interacting with the environment until it learns the optimal actions for the environment's states. The key advantage of RL is that it considers the long term effects of the actions by maximizing the overall discounted reward during its interactions with the environment. Despite this advantage, there are few researchers that have employed RL in the field of anomaly detection, where the agent makes decisions regarding the presence of an anomaly. Huang et al. [20] and Yu & Sun [41] respectively proposed a value-based and policy-based *forward* RL anomaly detection methods based on Deep Reinforcement Learning (DRL), where the objective is to learn the optimal actions, given the reward function values. Oh & Iyengar [28] used an *inverse* RL method for the task of anomaly detection, where the objective is to determine the underlying reward function, and subsequently the agent preference, from a sample of optimal actions. The above-mentioned researchers have showed the advantages of both forward and inverse RL for anomaly detection on various time series. However, the employed time series do not consist in explicit SHM data that are collected on the civil infrastructures. Also, their methods were only applied directly to time series data (i.e., environment) without considering the external effects and the kinematics of the underlying true responses. In addition, the performance metrics for the anomaly detection method do not account for the anomaly characteristics.

To the best of the authors' knowledge, this is the first time that RL is combined with BDLM in order to address the existing limitations of the anomaly detection in SHM time series data by: (i) explicitly detecting anomalies within the underlying irreversible behavior of the structure, (ii) considering the long term effect of detecting anomalies, and

(iii) accounting for the anomaly characteristics to evaluate performance and detectability capacity of the anomaly detection method. In Section 2 we introduce the BDLM theory for building empirical models for both normal and abnormal structural conditions. Section 3 presents the formalism of anomaly detection in the context of RL and the method for simulating the environment. Section 4 presents the *Q-learning* approach [34] that is employed in order to train the agent. Section 5 presents the Anomaly Detection (AD) framework proposed in this paper, along with the methods for quantifying the anomaly detectability capacity. Section 6 illustrates the potential of the framework in order to detect known and unknown anomalies on elongation data recorded on a bridge located in Canada. In addition, this section explores the factors influencing the anomaly detectability capacity of several agents.

2 Empirical model estimation

At time t, the vector of $\mathtt{Y}$ observations $\mathbf{y}_t = [y_1 \; y_2 \; \cdots \; y_{\mathtt{Y}}]^\intercal$ is decomposed into the hidden state vector $\mathbf{x}_t = [x_1 \; x_2 \; \cdots \; x_{\mathtt{X}}]^\intercal$, which contains $\mathtt{X}$ *hidden state* variables representing reversible and irreversible behaviors. Here, we use BDLM in order to model the system dynamics and estimate the hidden state variables, recursively [17, 25]. The evolution of hidden state variables over time is expressed by the linear *transition model*

$$\mathbf{x}_{t+1} = \mathbf{A}_{t+1}\mathbf{x}_t + \mathbf{w}_{t+1}, \;\; \mathbf{w}_{t+1} \sim \mathcal{N}(\mathbf{w}; \mathbf{0}, \mathbf{Q}_{t+1}), \tag{1}$$

in which $\mathbf{A}_{t+1}$ is the transition matrix and $\mathbf{Q}_{t+1}$ is the process noise. The *observation model*

$$\mathbf{y}_{t+1} = \mathbf{C}_{t+1}\mathbf{x}_{t+1} + \mathbf{v}_{t+1}, \;\; \mathbf{v}_{t+1} \sim \mathcal{N}(\mathbf{v}; \mathbf{0}, \mathbf{R}_{t+1}) \tag{2}$$

is described by the observation matrix $\mathbf{C}_{t+1}$ and the observation covariance $\mathbf{R}_{t+1}$. Furthermore, $\mathcal{M} = \{\mathbf{A}_{t+1}, \mathbf{C}_{t+1}, \mathbf{Q}_{t+1}, \mathbf{R}_{t+1}\}$ is a *model matrices* set parametrized by $\mathcal{P}$, a set of unknown parameters that is estimated using the observations. A complete review of BDLM and the choices of the hidden state variables and corresponding model matrices are provided by Goulet [17].

2.1 Stationary and non-stationary regimes

In this study, we employ a local level *baseline* hidden state variable $x^{\mathtt{LL}}$ in order to describe the irreversible response of a structure. The rate of the change in the baseline is the *trend* hidden state variable $x^{\mathtt{LT}}$. We define the *stationary regime* as a regime for which the statistical properties of the trend do not vary over time, i.e. a locally constant trend [17, 18]. In contrast, a *non-stationary regime* refers to the situation in which the trend is not locally constant. We use the rate of the change in the trend, i.e. the *acceleration* hidden state variable $x^{\mathtt{LA}}$, for modelling the non-stationary regimes. Therefore, we can choose two sets of model matrices $\mathcal{M}^{\mathtt{s}}$ and $\mathcal{M}^{\mathtt{ns}}$ which respectively model the stationary ($\mathtt{s}$) and non-stationary ($\mathtt{ns}$) regimes. The BDLM process can be summarized by

$$\left\{\boldsymbol{\mu}_{t+1|t+1}, \boldsymbol{\Sigma}_{t+1|t+1}, \pi^{\mathtt{ns}}_{t+1|t+1}\right\} = \text{BDLM}(\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t}, \pi^{\mathtt{ns}}_{t|t}, \mathbf{y}, \mathcal{M}^{\mathtt{s}}, \mathcal{M}^{\mathtt{ns}}), \tag{3}$$

where, $\boldsymbol{\mu}_{t|t} \equiv \mathbb{E}[\mathbf{x}_t|\mathbf{y}_{1:t}]$ and $\boldsymbol{\Sigma}_{t|t} \equiv \text{cov}[\mathbf{x}_t|\mathbf{y}_{1:t}]$ respectively are the expected value vector and covariance matrix of the hidden states vector at time t, conditional on $\mathbf{y}_{1:t} \equiv \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_t\}$, a set of observations for the times from $t = 1$ up to t. Furthermore, $\pi^{\mathtt{ns}}_{t|t}$ is the probability of the non-stationary regime at time t, given all the data up to time t. Note that the subscript $_{t|t}$ changes to $_{t+1|t+1}$ whenever we are interested in times ranging from 1 up to $t + 1$.

2.2 Anomaly detection

An *anomaly* refers to a change of regime from stationary to non-stationary for which the regime switches from a locally constant trend to a locally constant acceleration. The stationary and non-stationary regimes represent respectively the normal and abnormal state of the structure. Therefore, the objective of *anomaly detection* is to automatically identify the transitions from a stationary regime to a non-stationary one, and vice versa.
Figure 1 presents an example of the application of the BDLM to identify the probability of the non-stationary regime. It illustrates the baseline hidden state variable $x^{\mathtt{LL}}$, i.e. local level, the locally constant trend $x^{\mathtt{LT}}$, locally constant acceleration $x^{\mathtt{LA}}$, and the probability of the non-stationary regime $\pi^{\mathtt{ns}}$ for elongation data (E) collected on a bridge. Data description and the detailed empirical model estimation are presented in §6. The timestamps with high probabilities of regime switching within the year 2011 correctly indicates the presence of an anomaly, which in this case is due to the structural interventions. We can see the effect of the anomaly on the underlying responses during the year 2011. Note that after the intervention, BDLM identifies a return to the stationary regime. However, using only the probability of the non-stationary regime is prone to false alarms: there are situations where the probability of the non-stationary regime is not zero, but we do not know whether the changes in underlying responses correspond to an anomaly or not. For instance, the probabilities of the non-stationary regime during February 2007 and October 2009 are not zero, although the corresponding trends are not as high as the ones occurring between May and October 2011. Therefore, the probability of a non-stationary regime is not a perfect

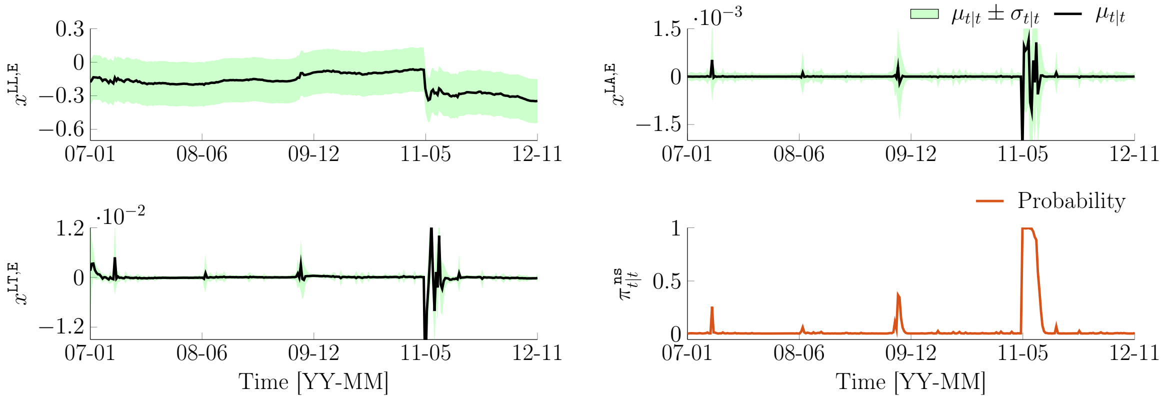


Figure 1: Illustration of the hidden state variable estimation and the probability of non-stationary regime. $x^{\mathtt{LL}}$, $x^{\mathtt{LT}}$, and $x^{\mathtt{LA}}$ respectively are the local level, locally constant trend, and locally constant acceleration hidden state variables corresponding to the elongation measurement on a bridge. $\pi^{\mathtt{ns}}$ is the probability of the non-stationary regime.

indication of an anomaly in these situation. The following sections present how an anomaly detection framework can tackle these situations by training an RL agent in a simulated environment that takes into account the additional information regarding the trend.

3 Anomaly detection in the context of RL

Making decisions regarding the presence of an anomaly requires incorporating the information both from the irreversible responses and the probability of non-stationary regime. For simplicity, in this study we rely on the expected value of the local trend $\mu_t^{\texttt{LT}}$ and the probability of non-stationary regime $\pi_t^{\texttt{ns}}$ as the *environment's state*. This section frames the anomaly detection in the context of RL and presents the simulated environment's state mimicking the underlying behavior of the structure.

3.1 Anomaly detection

At time t, the environment's state vector $\mathbf{s}_t = [\mu_t^{\texttt{LT}}\ \pi_t^{\texttt{ns}}]^\intercal \in \mathcal{S}$ consists in the mean value of the local trend hidden state variable and the probability of the non-stationary regime. The vector $\mathbf{s}_t$ is obtained from the BDLM process and represents the environment that the agent interacts with. Figure 2 shows an example of a continuous *environment's state space* $\mathcal{S}$ for which the solid and dashed lines represent the evolution of the vector $\mathbf{s}_t$ over time. An *episode* $\mathcal{Z} = \{\mathbf{s}_1, \mathbf{s}_2, \cdots\}$ is defined as a set of environment's state vectors for consecutive time steps. Any episode such as the one depicted by the solid line is initialized at $\mathbf{s}_1$, at which time it follows a stationary regime. At time $t = t_{\mathrm{s}}$, when an anomaly happens, the regime switches from stationary to non-stationary. At time $t = t_{\mathrm{e}},\ t_{\mathrm{e}} > t_{\mathrm{s}}$, the regime switches back to stationary. For the agent, anomaly detection refers to making an optimal decision at time t in the form of taking an optimal action $a_t \in \mathcal{A} = \{a^0 \textit{ trigger an alarm}, a^1 \textit{ not trigger an alarm}\}$, according to the possible presence of an anomaly. Therefore, for the non-stationary regime, i.e., $t \in (t_{\mathrm{s}}, t_{\mathrm{e}})$, the agent's optimal action is to only trigger an alarm once, such as the one shown by the red dotted circle, and as close as possible to time $t = t_{\mathrm{s}}$. On the other hand, when the regime is stationary, that is for $t \notin (t_{\mathrm{s}}, t_{\mathrm{e}})$, the agent's optimal action is to not trigger an alarm.

In Figure 2, the reward $r(\mathbf{s}_t, a_t) \in \mathbb{R}$ is a function of the environment's state $\mathbf{s}_t$ and of the agent's action a_t indicating whether the agent takes desirable actions during the agent-environment interactions [27, 38]. Its value depends on whether the action results in a false $(a_t = a^1, \forall t < t_{\mathrm{s}})$ or delayed $(a_t = a^0, \forall t > t_{\mathrm{s}})$ triggering of an alarm, and rightfully not triggering an alarm, i.e., $a_t = a^0, \forall t \notin (t_{\mathrm{s}}, t_{\mathrm{e}})$. In this manuscript, the terms *delayed alarm* and *missed alarm* are interchangeable for times $t > t_{\mathrm{s}}$. A *policy* $\pi : \mathcal{S} \rightarrow \mathcal{A}$ is defined as a map from the environment's state space to the action space describing the action to be taken by the agent given the environment's state. For a given policy π, the value of taking the action a in the environment's state $\mathbf{s}$ is denoted by the action-value function $q_\pi(\mathbf{s}, a)$. The action-value function is formally defined as

$$q_\pi(\mathbf{s}_t, a_t) = \mathbb{E}_\pi\left[R_t | \mathbf{s}_t, a_t\right], \tag{4}$$

where, the *return* $R_t = \sum_{t=1}^{\infty} \gamma^{t-1} r_t(\mathbf{s}_t, a_t)$ depends on the current and future reward values, and $\gamma \in (0, 1]$ is the *discount factor* quantifying how much of the future reward values

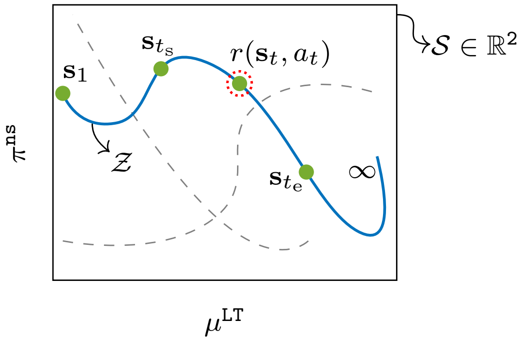


Figure 2: Illustration of the episodes $\mathcal{Z}$ and environment's state vector $\mathbf{s}_t = [\mu_t^{\mathtt{LT}}\ \pi_t^{\mathtt{ns}}]^\intercal \in \mathcal{S}$ for which the agent triggers an alarm at time t during the non-stationary regime. The curves inside the environment represent different realizations. One of them (solid line) is highlighted to explain the nomenclature; $\mathbf{s}_{t_\mathrm{s}}$ and $\mathbf{s}_{t_\mathrm{e}}$ are respectively associated with the start and end time of the non-stationary regime. The red dashed circle indicates the time for which the agent triggers the alarm. The optimal action is corresponding to trigger the alarm as close as possible to the time $t = t_\mathrm{s}$.

are discounted in comparison with the current one. Equation 4 states that the value of a state-action pair $(\mathbf{s}_t, a_t)$ at time t is the expected return of the agent that starts from the environment's state $\mathbf{s}_t$ and takes the action a_t, and takes the future actions by following the policy π [35]. In this setup, the anomaly detection can be regarded as a sequential decision making problem in the context of RL, where the objective is to maximize the expected return and subsequently the action-value function. Therefore, the anomaly detection problem is governed by the execution of the optimal policy π^* defined by

$$\begin{aligned}\pi^*(\mathbf{s}) &= \underset{a_t \in \mathcal{A}}{\arg\max}\ q_{\pi^*}(\mathbf{s}_t, a_t), \\ &= \underset{a_t \in \mathcal{A}}{\arg\max}\ \mathbb{E}_{\pi^*}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t(\mathbf{s}_t, a_t)|\mathbf{s}_t, a_t\right].\end{aligned} \tag{5}$$

The RL-based anomaly detection has two advantages. First, the policy depends on the environment's state $\mathbf{s}$, which consists in the information from both the trend hidden state variable and the probability of the non-stationary regime. This addresses the issues regarding the decision making solely based on the probability of the non-stationary regime as explained in §2.2. Second, in Equation 5, evaluating the action-values depends on the current and future reward values. This implies that the the decision making considers the long-term effect of taking actions. Therefore, maximizing the accumulated discounted rewards implicitly results in maximizing the anomaly detectability, while minimizing the number of false and delayed alarms.

3.2 Simulated environment's state space

For SHM applications, the agent-environment interaction should mimic the underlying infrastructure behavior in the presence of various anomalies. Training the agent to detect anomalies solely based on the observed structural responses cannot address this challenge because anomalies are rare events in the context of infrastructure monitoring. Therefore, we simulate the environment's state using a stochastic empirical model of the structural responses and a stochastic *anomaly function.*

3.2.1 Anomaly Function

We model an anomaly as a switch from a stationary to a non-stationary regime as described in §2.2. In order to simulate such a transition, we define the anomaly function representing a change in the trend using a continuous logistic function

$$f_{\text{a}}(t;\mathcal{J}) = \frac{h_{\text{a}}}{|h_{\text{a}}|} \cdot \frac{|h_{\text{a}}| + 2\epsilon_{\text{a}}}{1 + e^{-k(t-t_{\text{c}})}}, \tag{6}$$

in which the steepness k is governed by

$$k = \frac{2}{w_{\text{a}}} \ln\left(\frac{|h_{\text{a}}|}{\epsilon_{\text{a}}} + 1\right). \tag{7}$$

BDLMs represent the kinematics of time series via a local level, a locally constant trend, and a locally constant acceleration component. Also, the transition from a locally constant trend to a locally constant acceleration (i.e., anomaly) results in a drift on the local level responses over the time. The logistic function is an infinitely differentiable function over its entire domain and have exact closed-form derivatives and integrals. Hence, choosing the logistic function for modelling the changes in the trend allows evaluating its derivative and integral, which determines the kinematics of the anomaly function; this property allows merging the anomaly function with hidden state variables as explained in §3.2.2. In addition, the logistic function parameters are associated with the non-stationary regime characteristics: in Equation 6, $\mathcal{J} = \{h_{\text{a}}, w_{\text{a}}, t_{\text{c}}\}$ is a set of random variables describing the stochastic characteristics of the non-stationary regime. It consists in the height h_{a}, duration w_{a}, and center t_{c} of the anomaly function. The user-defined *anomaly tolerance* ϵ_{a} is the minimum magnitude corresponding to the time for which the non-stationary regime starts, i.e. $f_{\text{a}}(t_{\text{s}}) = \epsilon_{\text{a}}$. Figure 3 illustrates three different anomaly realizations with the same height $h_{\text{a}} > 0$ and different centers and durations such that $w_{\text{a}}^1 < w_{\text{a}}^2 < w_{\text{a}}^3$. Hatched regions on the horizontal axis indicate the out-of-domain values, which are discarded. For the anomaly function shown by a solid line, the non-stationary duration w_{a} is centerd at time t_{c} and is associated with the start time $t_{\text{s}} = t_{\text{c}} - w_{\text{a}}/2$ and end time $t_{\text{e}} = t_{\text{c}} + w_{\text{a}}/2$. Anomaly function values corresponding to the start and end time of the non-stationary duration are ϵ_{a} and $h_{\text{a}} + \epsilon_{\text{a}}$, respectively. The shaded area indicates the anomaly tolerance for which the the regime is considered to be stationary. For the cases where $|h_{\text{a}}| < \epsilon_{\text{a}}$ or $h_{\text{a}} = 0$, we assume that there is no anomaly, i.e., $f_{\text{a}}(t) = 0,\ \forall t$. The anomaly function magnitude $|f_{\text{a}}(t)| \in (0, |h_{\text{a}}| + 2\epsilon_{\text{a}})$ approaches 0 as t tends to $-\infty$. Therefore, at time t_1, the anomaly function introduces an error of magnitude $|f_{\text{a}}(t_1)|$, which must satisfy $|f_{\text{a}}(t_1)| < \varepsilon$. Here, ε is a user-defined tolerance. In order to start episodes with a stationary regime as described in §3, we accept anomaly realizations for which $t_{\text{s}} \geq t_1 + w_0$, where $w_0 > 0$ is the number of initial time steps without anomaly.

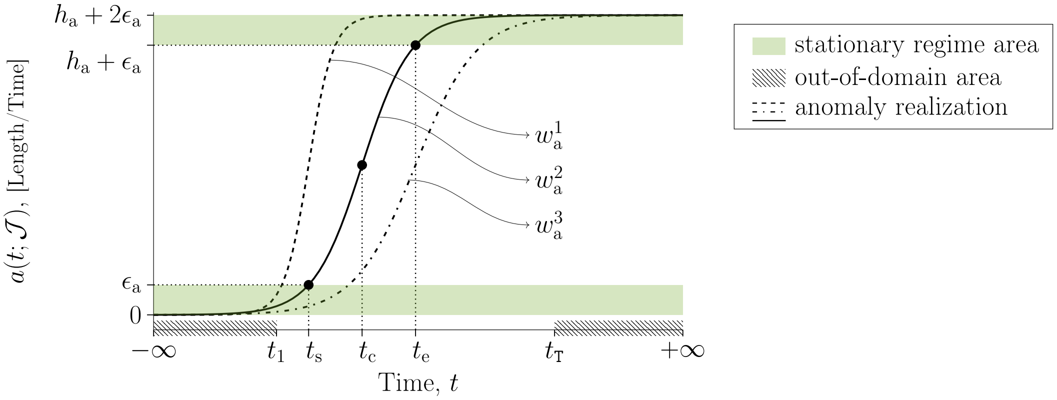


Figure 3: Illustration of an anomaly function $f_a(t;\mathcal{J})$ with $h_a > 0$. The curves represent three anomaly realizations with the same height $h_a^1 = h_a^2 = h_a^3 = h_a$ and anomaly tolerance $\epsilon_a^1 = \epsilon_a^2 = \epsilon_a^3 = \epsilon_a$, and different durations such that $w_a^1 < w_a^2 < w_a^3$. One of them (solid line) is highlighted to explain the nomenclature. The non-stationary regime is centered at time t_c for which $t_{s,e} = t_c \mp w_a/2$.

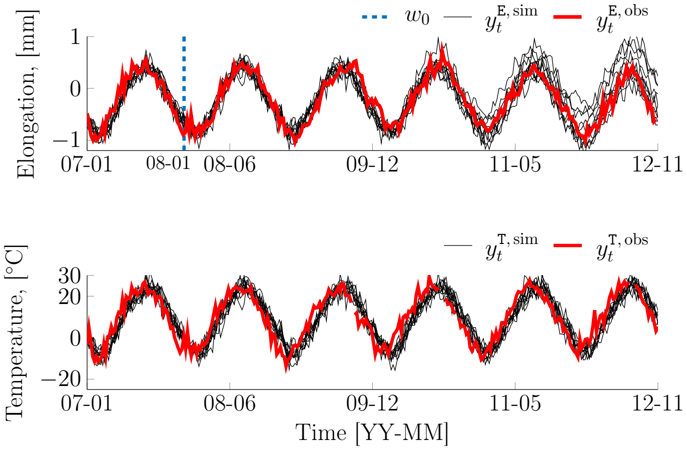
3.2.2 Simulated environment's state

The process of simulating the environment's state involves simulating abnormal structural responses $\mathbf{y}^{\text{sim}}$ and employing them in Equation 3 in order to estimate hidden state variables and regime probabilities. To this end, we add the anomaly function values $f_a(t;\mathcal{J})$ and corresponding derivative $\frac{d}{dt} f_a(t;\mathcal{J})$, and integral $\int f_a(t;\mathcal{J})\,dt$ to the associated hidden states variables' expected values $\boldsymbol{\mu}_{t|t}^{\text{obs}}$ corresponding to the trend, acceleration, and baseline responses. By employing the observation model as described in §3, the simulation of new structural responses is summarized by

$$\mathbf{y}^{\text{sim}} = \text{SIM}(\boldsymbol{\mu}_{t|t}^{\text{obs}}, f_a(t;\mathcal{J}), \boldsymbol{\Sigma}_{t|t}^{\text{obs}}, \mathcal{M}^{\mathbf{s},\text{obs}}), \tag{8}$$

where, the superscripts $^{\text{obs}}$ and $^{\text{sim}}$ refer to the quantities that are respectively obtained from the observed and simulated structural responses. Note that we use the model matrix $\mathcal{M}^{\mathbf{s},\text{obs}}$ associated with the stationary regime to simulate structural responses. Finally, the environment's state is estimated analogously to the BDLM process expressed in Equation 3, but using simulated structural response $\mathbf{y}_t^{\text{sim}}$ and model matrices $\mathcal{M}^{\mathbf{s},\text{obs}}$ and $\mathcal{M}^{\mathbf{ns},\text{obs}}$.

Figure 4 shows ten simulated responses on the bridge as described in §6. In each figure, the solid thick red line is the observations $\mathbf{y}^{\text{obs}}$, while the black lines are the simulated responses $\mathbf{y}^{\text{sim}}$. The vertical dashed line illustrates the number of time steps without anomaly, i.e. w_0. The table in Figure 4 depicts the parameters used to generate systematic anomalies: the height $h_a \sim \mathcal{N}(h_a; 0, \sigma_{h_a}^2)$ is sampled from a normal distribution with mean zero and the standard deviation σ_{h_a}. The duration $w_a \sim \ln\mathcal{N}(w_a; \lambda_{w_a}, \zeta_{w_a}^2) \in \mathbb{R}^+$ is sampled from a log-normal distribution parameterized in the log-space by the mean $\lambda_{w_a} = \ln\mu_{w_a} - \zeta^2/2$ and variance $\zeta^2 = \ln[1 + (\sigma_{w_a}/\mu_{w_a})^2]$. Here, μ_{w_a} and σ_{w_a} respectively are the mean and standard deviation of the non-stationary regime duration. The center of the non-stationary regime corresponds to the time $t_c \sim \mathcal{U}(t_c; t_1, t_T)$, which is sampled from a uniform distribution. In Figure 4, $\epsilon_{\mathrm{a,min}}$ is the minimum required anomaly function value at time t_{s}. Selecting the distributions and corresponding parameters for generating anomalies depend on the requirements of anomaly detection with respect to the prior knowledge about the anomaly heights and durations. Figure 4 is an example of simulated responses, where the anomaly heights and durations follow a normal and log-normal distribution. However, in many cases such prior knowledge does not exist and it is suggested to use uniform distribution to generate anomalies. In addition, the parameters of the distribution depend on the expected anomalies on the structure. For instance, we have a knowledge about a particular anomaly on the bridge as shown in Figure 1 due to interventions. Such a knowledge helps defining the bounds of the uniform distribution for the anomaly function stochastic characteristics $\mathcal{J}$. Table 3 in Appendix B shows the distributions and parameters explored in this work. As it is explained in §6, the bounds for the anomaly heights are defined such that the agent is trained for lower and higher anomaly heights compared with the one obtained from the intervention.

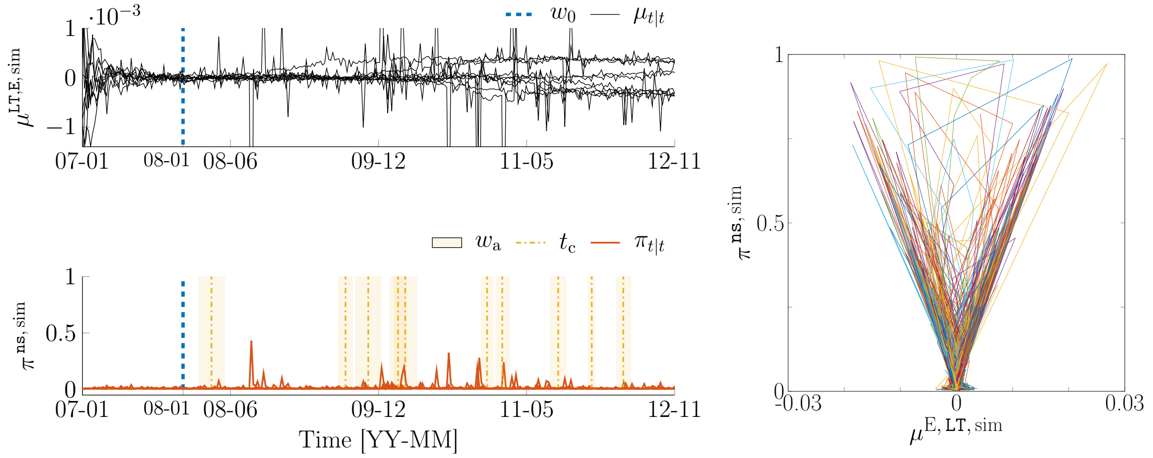


Anomaly function properties

Parameter	Value	Unit
$\sigma_{h_{\mathrm{a}}}$	1×10^{-4}	[mm/day]
$\mu_{w_{\mathrm{a}}}$	60	day
$\sigma_{w_{\mathrm{a}}}$	30	day
$\epsilon_{\mathrm{a,min}}$	1×10^{-4}	[mm/day]
w_0	365	day
ε	1×10^{-5}	[mm/day]

Figure 4: Illustration of the comparison between 10 simulated anomalous structural responses $\mathbf{y}^{\,\mathrm{sim}}$ and observations $\mathbf{y}^{\,\mathrm{obs}}$ for elongation and temperature measurements. The height $h_{\mathrm{a}} \sim \mathcal{N}(h_{\mathrm{a}}; 0, \sigma^2_{h_{\mathrm{a}}})$ is sampled from a normal distribution, the duration $w_{\mathrm{a}} \sim \ln\mathcal{N}(w_{\mathrm{a}}, \lambda_{w_{\mathrm{a}}}, \zeta^2_{w_{\mathrm{a}}})$ is sampled from a log-normal distribution parameterized in the log-space, and w_0 indicates the number of time steps without anomaly during the simulation. The stationary regime model matrix $\mathcal{M}^{\mathbf{s},\mathrm{obs}}$ for generating time series is provided in Appendix A.

Figure 5a illustrates the estimated local trend mean values and the probability of the non-stationary regime for the same ten simulated structural responses as shown in Figure 4. The vertical dash-dotted lines and shaded areas correspond respectively to the center of the anomalies and their duration. Figure 5b shows simulated environment's states consisting in 500 episodes for which 60% of them are abnormal. The anomaly kinematics represent only a random *damage state*, which differs from the normal and abnormal states of the structure as described in §2, because they do not consist in the reversible and irreversible structural responses. The simulation procedure models the normal and abnormal state of the structure for a given anomaly realization considering that the anomaly kinematics interfere with the structural responses kinematics. The result presented in Figure 5a is



(a)

(b)

Figure 5: Simulation of 500 episodes for which 60% of them are anomalous: (a) illustration of 10 selected anomalous episodes for which the locally constant trend and probability of non-stationary regime are shown. The vertical dash-dotted line and shaded area respectively indicates the center and duration of the anomaly, and (b) illustration of the environment's state space $\mathcal{S}$.

an example of such a procedure. However, it shows that relying only on the probability of non-stationary regimes to detect anomalies is prone to false alarms as discussed in §2.2 because there are many timestamps for which the probability of non-stationary regimes are not zero, yet there is no anomaly. In addition, we do not have the prior knowledge about the occurrence of the anomalies. The RL-based anomaly detection copes with this issue by not only incorporating the trend hidden state variable along with the probability of the non-stationary regime, but also it takes the action of triggering the alarm based on the immediate reward and the future accumulated discounted ones; such a RL framework respects the state-of-the-art for sequential decision making [35].

4 Q-Learning

The tabular Q-learning approach [38] allows evaluating the action-values in Equation 5 with a model-free methods from which the agent can learn the optimal policies without having access to the transition probability between the environment's states [22, 27]. This section presents the Q-learning methodology for the *discretized environment's state space.*

4.1 Discretized environment's state space

We use tabular reinforcement learning to train the agent in a discretized environments state space $\hat{\mathcal{S}}$. Let $d_{\mathtt{m}\times\mathtt{n}} : \mathcal{S} \rightarrow \hat{\mathcal{S}}$ be a discretization map from the continuous environment's state space $\mathcal{S}$ into the discretized one $\hat{\mathcal{S}}$ with a grid size $\mathtt{m} \times \mathtt{n}$. Figure 6 shows the same environment's state space depicted in Figure 2 which has been discretized with a grid size of 5×4. The shaded areas indicate the discretized regions visited by the agent during an episode $\mathcal{Z}$, and the diamonds show the centers of these regions. Two successive continuous environment's states $\mathbf{s}_t$ and $\mathbf{s}_{t+1}$ are shown by solid circles. The corresponding discretized states are respectively $\hat{\mathbf{s}}_t = d(\mathbf{s}_t)$ and $\hat{\mathbf{s}}_{t+1} = d(\mathbf{s}_{t+1})$, which are shown by darker shaded areas. In the discretized setup, the task of RL is to find the optimal policy for the

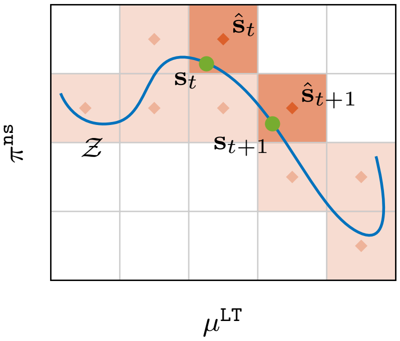


Figure 6: Mapping the continuous environment's state $\mathbf{s}_t = [\mu_t^{\mathtt{LT}}\ \pi_t^{\mathtt{ns}}]^\intercal \in \mathcal{S}$ into discretized one $\hat{\mathbf{s}}_t \in \hat{\mathcal{S}}$ for two successive times t and $t+1$ during an episode $\mathcal{Z}$. The shaded areas and diamonds marks indicate the discretized regions and their centers visited by the agent.

discretized regions in order to obtain the optimal policy for the whole environment. To this end, the agent interacts with the environment through several episodes. Here, each episode includes $\mathtt{T}$ time steps and at the end time of each episode, for that $t = \mathtt{T}$, the agent cannot select further action to determine the next state. Consequently, at time $t = \mathtt{T} - 1$, the agent-environment interaction is terminated and restarted with another episode such as the ones depicted in Figure 2 by dashed lines. In general, the grid size depends on the problem in hand. Using a coarse grid may result in poor policies for the discretized environment's states. Therefore, the grid needs to be fine enough to ensure the convergence of the Q-values [15]. On the other hand, increasing the grid size, increases the number of states and the subsequent memory allocations during the learning; this results in a slower learning as well as the necessity of a larger number of simulated data to train the agent as it is the case for any tabular Q-learning method. In this paper, the grid size is selected via trial-and-error procedure to find a tradeoff between the above-mentioned issues.

4.2 Q-Learning formalism

For a given discretized environment's state space $\hat{\mathcal{S}}$, Q-learning is formalized by the updating equation

$$Q(\hat{\mathbf{s}}_t, a_t) \leftarrow Q(\hat{\mathbf{s}}_t, a_t) + \alpha_t \left[r(\mathbf{s}_t, a_t) + \gamma \max_{a_{t+1}} Q(\hat{\mathbf{s}}_{t+1}, a_{t+1}) - Q(\hat{\mathbf{s}}_t, a_t) \right], \tag{9}$$

where the Q-values $Q(\hat{\mathbf{s}}, a)$ directly approximate the action-values regardless of the policy [35]. At each time step, the optimal action-value $q_{\pi^*}(\hat{\mathbf{s}}) = \max_{a \in \mathcal{A}} Q(\hat{\mathbf{s}}, a)$ is selected from the maximum Q-values associated with the possible actions. In Equation 9, the *learning rate* is defined by

$$\alpha_t \equiv \alpha(\mathtt{N}_{1:t}(\hat{\mathbf{s}}_t, a_t); c_\alpha) = \frac{c_\alpha}{c_\alpha + \mathtt{N}_{1:t}(\hat{\mathbf{s}}_t, a_t)}, \tag{10}$$

where $\mathtt{N}_{1:t}(\hat{\mathbf{s}}_t, a_t)$ is the number of times a state-action pair has been visited up to time t, and $c_\alpha \in \mathbb{R}^+$ is the *learning rate constant.* Note that in Equation 9, unlike the Q-values, the reward function $r(\mathbf{s}_t, a_t)$ value depends on the continuous state $\mathbf{s}_t$. The learning rate function must be formulated such that $\sum_{t=1}^{\infty} \alpha_t = \infty$ and $\sum_{t=1}^{\infty} \alpha_t^2 < \infty$, so the Q-values converge to the optimal action-values when the total number of interactions approaches infinity [33]. Therefore, the optimal policy can be obtained from the converged Q-values as

$$\pi^*(\hat{\mathbf{s}}) = \underset{a \in \mathcal{A}}{\arg\max}\, Q(\hat{\mathbf{s}}, a). \tag{11}$$

Here, the Q-values are considered as converged when the expected Q-values over all the visited states for $\mathtt{K}$ consecutive episodes satisfies the *convergence error ratio*

$$\delta_{z:z+\mathtt{K}} \equiv \frac{\left|\mathbb{E}_{\hat{\mathbf{s}}}[Q_{z+k}(\hat{\mathbf{s}})] - \mathbb{E}_{\hat{\mathbf{s}}}[Q_z(\hat{\mathbf{s}})]\right|}{\left|\mathbb{E}_{\hat{\mathbf{s}}}\left[Q_z(\hat{\mathbf{s}})\right]\right|} \leq \delta_0,\ \forall k = \{1, 2, \cdots, \mathtt{K}\}, \tag{12}$$

where δ_0 is a user-defined convergency tolerance and $Q_z(\hat{\mathbf{s}})$ is calculated at time $\mathtt{T} - 1$ from the episode z. The subscript $\hat{\mathbf{s}}$ indicates that the expected value is computed over all the visited states.

According to Equation 9, the agent can take two possible actions $a_t \in \mathcal{A}$ at each time step t during the training in order to update the corresponding Q-value. In this study, the action selection follows the ϵ-greedy algorithm [40]; at time t, the agent explores the environment by selecting the action randomly with the probability $\epsilon_t \in (0, 1)$ and exploits it by selecting the action from the current policy with the probability $1 - \epsilon_t$. We define an iteration-dependent exploration function as

$$\epsilon_t \equiv \epsilon(\mathtt{N}_{1:t}(\hat{\mathbf{s}}_t); c_\epsilon) = \frac{c_\epsilon}{c_\epsilon + \mathtt{N}_{1:t}(\hat{\mathbf{s}}_t)}, \tag{13}$$

where $\mathtt{N}_{1:t}(\mathbf{s}_t) = \mathtt{N}_{1:t}(\hat{\mathbf{s}}_t, a^0) + \mathtt{N}_{1:t}(\hat{\mathbf{s}}_t, a^1)$ is the number of times a state has been visited up to time t regardless of the selected actions, and $c_\epsilon \in \mathbb{R}^+$ is the *exploration constant.* To select the action a_t at time t, we draw a sample $u_t \sim \mathcal{U}(u_t; 0, 1)$ from a uniform distribution, and compare it with the corresponding exploration probability ϵ_t. The agent action selection is expressed as

$$a_t = \begin{cases} \text{Random selection from } \mathcal{A}, & u_t \sim \mathcal{U}(u_t; 0, 1) \leq \epsilon_t \\ \max_{a \in \mathcal{A}} Q(\hat{\mathbf{s}}_t, a_t), & \text{Otherwise} \end{cases}. \tag{14}$$

4.3 Reward Function

The reward function represents a feedback from the environment determining whether the agent should be encouraged or discouraged for the action it takes [10]. The reward function $r(\mathbf{s}_t, a_t)$ for an episode $\mathcal{Z}$ takes the form of a *confusion matrix* as shown in Table 1. Reward values are based on the comparison between the optimal action and the actions taken by the agent up to time t. Note that for each episode, the optimal action to be taken is known from the anomaly simulation process and corresponding non-stationary regime timestamps as described in §3.2.

In Table 1, the *true positive* reward r^{TP} and *true negative* reward r^{TN} are corresponding to the environment's states where the agent correctly selects the optimal action. On the other hand, the *false positive* (false alarm) reward r^{FP} and *false negative* (delayed or missed alarm) reward r^{FN} are associated with an incorrect action taken by the agent.

Table 1: Reward function's confusion matrix for the actions a^0 and a^1.

Agent action	Optimal action	
	a^0	a^1
a^0	r^{TP}	r^{FP}
a^1	r^{FN}	r^{TN}

In general, true positive and negative reward values are greater than the false positive and negative ones such that $r^{\mathrm{TN}}, r^{\mathrm{TP}} > 0$ and $r^{\mathrm{FP}}, r^{\mathrm{FN}} < 0$. The *delayed* reward $r^{\mathrm{FN}} = -\eta|f_{\mathrm{a}}(t;\mathcal{J}) - f_{\mathrm{a}}(t_{\mathrm{s}};\mathcal{J})|$, $t > t_{\mathrm{s}}$, depends on the anomaly function $f_{\mathrm{a}}(t;\mathcal{J})$ such that the agent is discouraged incrementally as it delays triggering a rightful alarm. η is the *false negative factor* with a $[\eta] = $ [Time/Length] enabling the user to control the reward value and ensuring that the resulting r^{FN} has the same dimension as of the other rewards.

5 Anomaly detection framework

The methods presented in previous sections are assembled into an anomaly detection framework applicable for full-scale infrastructure monitoring. The framework is illustrated in Figure 7 which consists in three stages identified by the dashed boxes. In the first two stages, the AD employs the empirical model of the structure to simulate the episodes consisting in stationary and non-stationary regimes as described in §3.2. In the third stage, the framework uses RL in order to learn the optimal policy as described in §4. In Figure 7, Stage I builds an empirical model from the structural responses $\mathbf{y}^{\mathrm{obs}}$ and the user-defined generic components associated with stationary and non-stationary regimes. The generic components involve model matrices $\mathcal{M}^{\mathtt{s},\mathrm{obs}}$ and $\mathcal{M}^{\mathtt{ns},\mathrm{obs}}$ as described in §2. In this stage, BDLM uses the structural responses and model matrices to estimate the hidden state variables mean vector $\boldsymbol{\mu}^{\mathrm{obs}}_{t|t}$ and covariance matrix $\boldsymbol{\Sigma}^{\mathrm{obs}}_{t|t}$, as well as the model parameters $\mathcal{P}^*$. In Stage II, the goal is to simulate new episodes from simulated structural responses $\mathbf{y}^{\,\mathrm{sim}}$ as described in §3.2. To this end, structural responses are simulated from the stationary model matrix $\mathcal{M}^{\mathtt{s},\mathrm{obs}}$, the stochastic anomaly function $f_{\mathrm{a}}(t;\mathcal{J})$, and estimated hidden state variables mean vector and covariance matrix obtained from the Stage I. Afterwards, BDLM uses both the model matrices and simulated structural responses to establish new environment's states. The goal of the last stage is to train the Q-learning agent through episodes $\mathcal{Z} \in \mathcal{S}$ in order to detect anomalies as explained in §4.2. In Stage III, two termination criteria are defined to restart the agent-environment interaction with a new episode by simulating new structural responses from Stage II. The first criterion, $t < t_{\mathtt{T}} - 1$, is related to end time of each episode as described in §4.1, and the second one, $a_t = a^0$, is related to the action taken by the agent such that whenever it triggers an alarm the episode

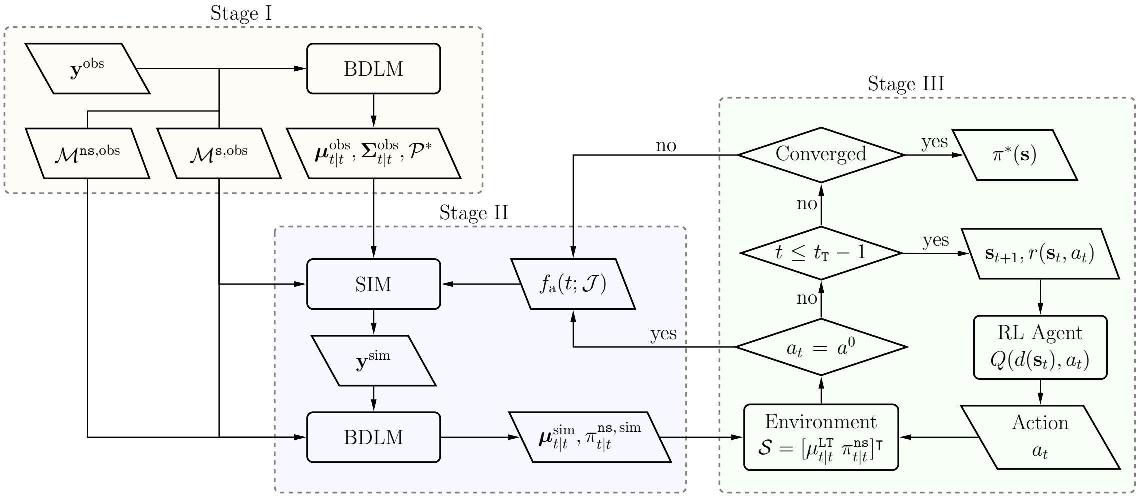

Stage I
$\mathbf{y}^{\text{obs}}$
BDLM
$\mathcal{M}^{\mathbf{ns},\text{obs}}$
$\mathcal{M}^{\mathbf{s},\text{obs}}$
$\boldsymbol{\mu}^{\text{obs}}_{t|t}, \boldsymbol{\Sigma}^{\text{obs}}_{t|t}, \mathcal{P}^*$
Stage II
SIM
$f_{\text{a}}(t;\mathcal{J})$
$\mathbf{y}^{\text{sim}}$
BDLM
$\boldsymbol{\mu}^{\text{sim}}_{t|t}, \pi^{\mathbf{ns},\text{sim}}_{t|t}$
yes
no
Stage III
Converged
yes
$\pi^*(\mathbf{s})$
no
$t \leq t_{\text{T}} - 1$
yes
$\mathbf{s}_{t+1}, r(\mathbf{s}_t, a_t)$
no
$a_t = a^0$
RL Agent
$Q(d(\mathbf{s}_t), a_t)$
Environment
$\mathcal{S} = [\mu^{\text{LT}}_{t|t}\ \pi^{\text{ns}}_{t|t}]^{\intercal}$
Action
a_t

Figure 7: Illustration of the anomaly detection framework in the state-space model, which consists in three stages corresponding to the empirical model estimation (Stage I), systematic anomaly generation (Stage II), and Q-learning (Stage II).

is terminated.

5.1 Anomaly detectability quantification

We quantify the anomaly detectability capacity of the agent within two quantities: (i) the probability of rightfully detecting anomalies, and (ii) the annual rate of false alarms. Table 2 summarizes the four expected situations according to the agent's optimal policy $\pi^*(\mathbf{s}) = a \in \mathcal{A}$ for the environment's state $\mathbf{s} \in \mathcal{S}$ during an episode $\mathcal{Z}$. Only testing an agent only against real structural responses is not representative of the anomaly detectability capacity because anomalies are rare events. To overcome this issue, we simulate a test environment's state space $\mathcal{S}^{\text{test}}$ with labeled anomalies and test the agent against this newly unseen space.

Table 2: Agent detection terminology.

Decision	Agent optimal policy $\pi^*(\mathbf{s})$
true positive (TP)	$a^0,\ t \geq t_{\text{s}}$
false negative (FN)	$a^1,\ t \geq t_{\text{s}}$
true negative (TN)	$a^1,\ t < t_{\text{s}}$
false positive (FP)	$a^0,\ t < t_{\text{s}}$

5.1.1 Probability of true positive detection

The evaluation of the probability of true positive detection is a binary classification problem between TP and FN detections. The covariates are the anomaly height absolute value

$|h_\mathrm{a}|$ and number of time steps after the beginning of the non-stationary regime $\mathtt{N}_{t>t_\mathrm{s}}$, i.e., *anomaly detection duration.* For a test environment's state space $\mathcal{S}^\mathrm{test}$, a set of $\mathtt{D}$ instances is defined by

$$\mathcal{D} = \{\mathcal{D}_x, \mathcal{D}_y\} = \{(\mathbf{x}_i, y_i), \forall i \in \{1 : \mathtt{D}\}\}, \tag{15}$$

where, $\mathbf{x}_i = [|h_\mathrm{a}| \ \mathtt{N}_{t>t_\mathrm{s}}]^\intercal$ and $y_i \in \{-1 : \mathrm{FN}, +1 : \mathrm{TP}\}$, are respectively the covariates vector and binary classes. Note that in establishing $\mathcal{D}$, the policy after triggering an alarm remains triggering an alarm. We employ a *Gaussian Process Classification* (GPC) approach in order to perform the classification [32]. In this regard, the classification for the covariates vector is modelled by a Gaussian process $g : \mathbf{x} \to y$. GPC is parameterized by $\boldsymbol{\theta} = [\sigma_G \ \ell_1 \ \ell_2]^\intercal$, an unknown vector consisting in the process noise σ_G, and square exponential correlation function length-scales ℓ_1 and ℓ_2. Once the unknown vector of parameters is estimated from $\mathcal{D}$, the probability of true positive detection $\Pr(Y = +1 : \mathrm{TP}|\mathbf{x}, \boldsymbol{\theta}^*)$, conditional on the prediction covariates vector $\mathbf{x}_*$ is available. The probability of false negative, conditional on the prediction covariates vector $\Pr(Y = -1 : \mathrm{FN}|\mathbf{x}, \boldsymbol{\theta}^*) = 1 - \Pr(Y = +1 : \mathrm{TP}|\mathbf{x}, \boldsymbol{\theta}^*)$. The reader is referred to Rasmussen & Williams [32] for a future explanations regarding GPC.

5.1.2 Annual false positive detection rate

The evaluation of the annual false positive detection rate R^FP encompasses the testing of the agent against a test environment's state space $\mathcal{S}^\mathrm{test}$ for which all the episodes contains no anomaly, and counting the *number of false positive detections* #FP. For a total number of time steps in the environment's state space *# time steps*, and the average of the number of time steps per year *# time steps/year*, the annual false positive detection rate is obtained by

$$R^\mathrm{FP} = \frac{\#\mathrm{FP} \times \#\ \text{time steps/year}}{\#\ \text{time steps}}. \tag{16}$$

6 Case study

This section applies the AD framework described in §5 on the data collected on a bridge located in Canada. The data are acquired over a period of 6 years (292 time steps) and the observation vector $\mathbf{y}_t^\mathrm{obs} = [y_t^\mathtt{E}, y_t^\mathtt{T}]^\intercal$ consists in the elongation ($\mathtt{E}$) and air temperature ($\mathtt{T}$) data as shown in Figure 8. The average time step length for the data is 7 days. The shaded area indicates the occurrence of an intervention on the structure during the year 2011. We can identify a yearly periodic pattern in elongation dataset, which can be explained by its dependence on the air temperature.

6.1 Empirical model estimation

Following Stage I of the ADD framework, we use two empirical models in order to represent the stationary and non-stationary regimes of the elongation data and their dependency on the temperature. The vector of hidden state variables $\mathbf{x}_t$ for each empirical model is

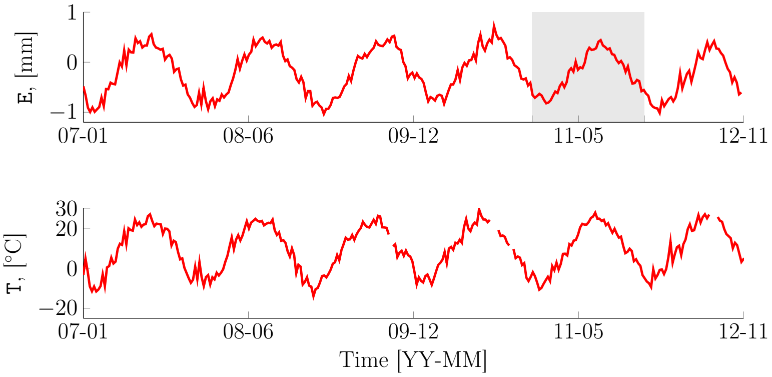


Figure 8: Elongation ($\mathtt{E}$) and air temperature ($\mathtt{T}$) measurements collected on a bridge over a period of 6 years. The shaded area is corresponding to an intervention on the bridge during the year 2011.

identical and follows

$$\mathbf{x}_t = \left[\underbrace{\mathbf{x}_t^{\mathtt{LA,E}}; \mathbf{x}_t^{\mathtt{KR,E}}; \mathbf{x}_t^{\mathtt{AR,E}}}_{\text{elongation components}} ; \underbrace{\mathbf{x}_t^{\mathtt{L,T}}; \mathbf{x}_t^{\mathtt{KR,T}}; \mathbf{x}_t^{\mathtt{AR,T}}}_{\text{temperature components}} \right], \tag{17}$$

where $[\,;]$ indicates the column-wise concatenation of the vectors and the additional superscripts $^{\mathtt{E}}$ and $^{\mathtt{T}}$ indicates that the observations are respectively for elongation and temperature. In Equation 17, the local acceleration vector $\mathbf{x}_t^{\mathtt{LA}} = \left[x_t^{\mathtt{L}}, x_t^{\mathtt{T}}, x_t^{\mathtt{LA}}\right]^{\mathtt{T}}$ describes a locally constant acceleration over a time step. It consists of three components acting jointly: a level ($\mathtt{L}$) component to represent the baseline of the structural behavior, a trend ($\mathtt{T}$) component to represent the rate of change of the baseline, and a local acceleration ($\mathtt{LA}$) component representing the acceleration over a time step. The kernel regression vector $\mathbf{x}_t^{\mathtt{KR}}$ models the reversible periodic patterns for each observation [24]. Here, we use 10 hidden control points for the kernel regression which results in the vector $\mathbf{x}_t^{\mathtt{KR}} = \left[x_{t,0}^{\mathtt{KR}}\ x_{t,1}^{\mathtt{KR}}\ \cdots\ x_{t,10}^{\mathtt{KR}}\right]^{\mathtt{T}}$ consisting in 10 hidden state variables. The auto-regressive component $\mathbf{x}_t^{\mathtt{AR}} = x_t^{\mathtt{AR}}$ models the time-dependent residual term representing the effect of other phenomena that cannot be modelled by the components employed. Note that, in order to differentiate the stationary and non-stationary regimes, the corresponding row and column in the $x_t^{\mathtt{LA}}$ component model matrices are set to be zero for all timestamps [25]. Appendix A provides the detailed model matrices $\mathcal{M}^{\mathtt{s}}$ and $\mathcal{M}^{\mathtt{ns}}$. Moreover, the model parameters are

$$\mathcal{P} = \Bigg\{ \underbrace{z^{\mathrm{s,s}}, z^{\mathrm{ns,ns}}}_{\text{regime transition}}, \underbrace{\sigma^{\mathtt{LT,E}}, \sigma^{\mathtt{LA,E}}, \sigma^{\mathtt{LTT,E}}, \ell^{\mathtt{KR,E}}, \sigma_0^{\mathtt{KR,E}}, \phi^{\mathtt{AR,E}}, \sigma^{\mathtt{AR,E}}, \sigma_v^{\mathtt{E}}}_{\text{elongation}, \mathtt{E}}, \tag{18}$$

$$\underbrace{\beta^{\mathtt{AR,E|T}}, \beta^{\mathtt{KR,E|T}}}_{\text{dependency}, \mathtt{E|T}}, \underbrace{\sigma^{\mathtt{LL,T}}, \ell^{\mathtt{KR,T}}, \sigma_0^{\mathtt{KR,T}}, \phi^{\mathtt{AR,T}}, \sigma^{\mathtt{AR,T}}, \sigma_v^{\mathtt{T}}}_{\text{temperature}, \mathtt{T}} \Bigg\},$$

where $z^{i,i}$ is transition probability among the regimes, $\sigma \in \mathbb{R}^+$ is the standard deviation of the process noise for the corresponding component, $\ell^{\texttt{KR}} \in \mathbb{R}^+$ is the kernel length, $\sigma_0 \in \mathbb{R}^+$ is the standard deviation of the hidden periodic pattern, $\phi^{\texttt{AR}} \in (0, 1]$ is the auto-regressive coefficient, and $\sigma_v \in \mathbb{R}^+$ is the observation standard deviation. Here, we assume that the kernel period and the standard deviation for the hidden control points are respectively 365.2422 days and zero. In Equation 18, $\beta^{\texttt{AR,E|T}} \in \mathbb{R}$ and $\beta^{\texttt{KR,E|T}} \in \mathbb{R}$ are the regression coefficients expressing the dependence between the elongation and the temperature. The hidden state variables estimations and optimization of model parameters $\mathcal{P}^*$ are carried out by using openBDLM toolbox [13], and the corresponding values are reported in Appendix A.

6.2 Anomaly detection

Building the environment's state space and training the RL agent involves using the model matrices obtained from Stage I, realizations of the anomaly function from Stage II, and training the agent in Stage III. The anomaly function's stochastic characteristic set $\mathcal{J}$ for the simulated environment's state space as well as the agent configurations are presented in Appendix B. The environment's state space consists in $\mu_t^{\texttt{LT}} \in (-0.06, 0.06)$ and $\pi_t^{\texttt{ns}} \in (0, 1)$. The space is discretized with the non-uniform grid size of 100×60 and we train the agent for 600000 episodes, 90% of which contains an anomaly. The local trend mean values $\mu_t^{\texttt{LT}}$ are symmetrical with respect to the $\mu_t^{\texttt{LT}} = 0$. Therefore, the training of the agent is carried out only for the absolute value of $|\mu_t^{\texttt{LT}}|$ in order to remove the induced redundancies corresponding to the equivalent state-action pairs [36]. Figure 9 shows the convergence of the Q-values during the training. The black line is the expected optimal Q-values over all visited states $\mathbb{E}_{\hat{\mathbf{s}}}[Q^*(\hat{\mathbf{s}})]$, and the blue line is the mean of the convergence error ratio $\bar{\delta}_{i:i+10}$ for each ten consecutive episodes. The total elapsed time for the training of the agent is approximately 22 hours using 24 Central Processing Unit (CPU) cores. Note that the AD framework had to be implemented from the scratch, and it is currently not yet fully optimized for computational efficiency.

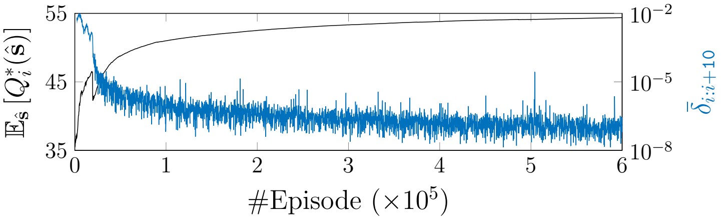


Figure 9: Convergence of the expected Q-values $\mathbb{E}_{\hat{\mathbf{s}}}\left[Q_i^*(\hat{\mathbf{s}})\right]$ corresponding to the visited states, and error ratio $\bar{\delta}_{i:i+10}$ averaged on the each ten consecutive episodes. The agent is trained for 600000 episodes results in the total number of 1.75×10^8 possible agent-environment interactions. The total elapsed time for the training of the agent is approximately 22 hours using 24 CPU cores.

Figure 10 illustrates the agent optimal policy and policy post-processing applied on the optimal policy. The red and blue regions are respectively corresponding to the actions a^0 *trigger an alarm*, and a^1 *not trigger an alarm*. The areas without a dedicated colour indicates the non-visited states. Figure 10a shows the agent optimal policy obtained from the training. As it is seen, the policies in the vicinity of the boundary of the two actions are noisy. Q-values are smoothed and re-evaluated according to Equation 11, in order to reduce the noise. Figure 10b shows the smoothed policy using the *mean filter* [37] with a rectangle kernel size of $(2 \times \mathtt{N}_{\mu^{\mathtt{LT}}} + 1) \times (2 \times \mathtt{N}_{\pi^{\mathtt{ns}}} + 1)$. $\mathtt{N}_{\mu^{\mathtt{LT}}} = 5$ and $\mathtt{N}_{\pi^{\mathtt{ns}}} = 1$, are respectively the number of discrete states in $\mu^{\mathtt{LT}}$ and $\pi^{\mathtt{ns}}$ directions. Moreover, we use *k-nearest neighbour* (k-NN), $k = 10$, in order to assign labels to the non-visited states. The k-NN is directly applied on the policy after the smoothing process. Note that the non-visited states correspond to the states with the large anomaly heights magnitudes. In reality, such magnitudes are associated with extremely rare events such that using the probability of the non-stationary regime is adequate to identify the anomalies. In addition, one can use a variety of approaches for this task, such as Gaussian Process Regression (GPR) [32]. Here, we decided to opt for the simplicity of the k-NN as it does not correspond to a key aspect of the proposed method. Figure 10c shows the final policy of the RL agent. Note that in Figure 10 , the states are shown up to $|\mu_t^{\mathtt{LT}}| \leq 0.04$, and for all the state with $|\mu_t^{\mathtt{LT}}| > 0.04$ the obtained policy is to not trigger the alarm.

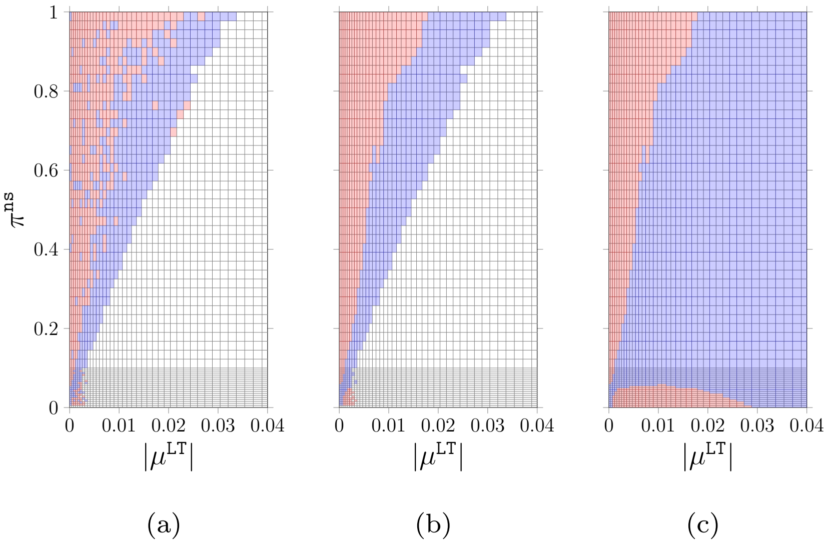
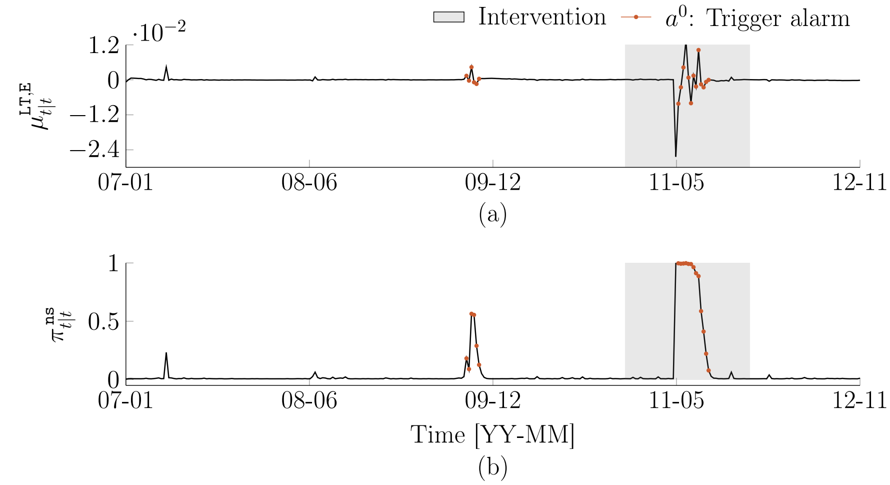


Figure 10: Illustration of the agent's policy, which is mapped on the half of the discretized environment's state space (i.e., $\mu^{\mathtt{LT}} > 0$). The discretized environment states $\bar{\mathcal{S}}$ are shown by the grey grid lines. The red and blue colours respectively are corresponding to the action of triggering the alarm a^0 and not triggering the alarm a^1. The areas without a dedicated colour indicate the non-visited states: (a) the agent's optimal policy obtained from the trained agent according to Equation 5, (b) the agent policy after smoothing the Q-values using the mean filter with a kernel size of 11 × 3 and re-evaluating the optimal policy, (c) the agent policy after performing 10-NN extrapolation to assign labels to the non-visited states.

In this study, we use the policies obtained from the smoothing and k-NN processes as shown in Figure 10c, in order to detect anomalies on the bridge data as depicted in Figure 8. Figure 11 illustrates the anomaly detection of the trained agent overlaid with the mean trend value and probability of the non-stationary regime obtained from BDLM as shown in Figure 1. The solid circles on the time series indicate the time steps when the agent takes the action of triggering an alarm. The shaded area indicates the known anomaly due to an intervention, which is correctly identified by the agent. Moreover, the probability of the non-stationary regime during October 2009 and February 2007 are greater than zero. However, relying on this probability for decision making encounters two issues: (i) there is no criterion available to determine the minimum probability of non-stationary regime for which the changes in underlying responses correspond to an anomaly, and (ii) the long term effects of the decision making is not considered. Therefore, the decision maker does not know whether to trigger the alarm at a particular time step when the probability of non-stationary regime is greater than zero, or to wait until more data becomes available before taking the action of triggering an alarm. In contrast, the RL agent incorporates the mean trend values as the additional information along with the probability of non-stationary regime. In addition, it recognizes the long term effect of the decision making by using the long term accumulated discounted rewards as presented in Equation 5. Therefore, as it is seen in Figure 11, the agent triggers alarms in October 2009, while the action in February 2007 is to not trigger any alarm. Another advantage of training the RL agent with the anomaly function proposed in this paper is that the agent can generalize its anomaly detectability for multiple anomalies. The anomaly function used during the training is a shift in the trend value without returning to the same value when the non-stationary regime ends. However, Figure 11a illustrates an example of a shift in the trend value which returns to the same value after the intervention, and which can be rightfully identified by the agent. In Figure 11, the source of the possible anomaly identified in October 2009 is unknown. In order to examine the robustness of the policy during this time, we introduce two additional artificial anomalies as shown in Figure 12a. Each artificial anomaly $\bar{f}_{\mathrm{a}}(t;\mathcal{J})$ consists in the



(a)

(b)

Figure 11: Illustration of the bridge anomaly detection obtained from the trained agent. For better comparison, the time steps for which the agent triggers the alarm are indicated with the red solid circles and overlaid with the (a) trend mean value and (b) probability of non-stationary regime obtained from BDLM as depicted in Figure 1.

superposition of the two anomaly functions presented in Equation 6, such that the regime returns to the same baseline value. The two artificial anomalies are located before an after the time stamps for which the agent triggers alarms according to Figure 11. The anomaly heights are respectively 2.8×10^{-3} and -4.2×10^{-3} mm/day, and the duration for both is 120 days. The reason for selecting the former height is to mimic the similar hidden state variables and regime switching probability as of the one that occurred on the bridge during October 2009. The latter height is selected to generate anomalies with a non-stationary regime probability $\pi^{\mathtt{ns}}_{t|t} < 1$ in the vicinity of the anomaly that occurred on the bridge in the year 2011. Following the simulation procedure explained in §3.2.2, the simulated structural response, trend mean value, and probability of non-stationary regime are shown in Figures 12b-d. The agent correctly identifies the two additional artificial anomalies apart from the the real ones. This confirms that the proposed AD framework is able to detect low-magnitude of anomalies that are typically associated with small probabilities of a non-stationary regime, such as the ones observed during October 2009 for which the anomaly's peak height is around 1 mm/year.

6.3 Quantifying anomaly detectability

We follow the procedure explained in §5.1 in order to quantify the anomaly detectability of the RL agent. For evaluating the true positive anomaly detections, the test environment's state space $\mathcal{S}^{\text{test}}$ consists in 3000 episodes for which all the episodes contains an anomaly. The anomaly function stochastic characteristics set $\mathcal{J}$ is provided in Appendix B. We use the GPC [31] in order to build a model for $\Pr(Y = +1 : \text{TP}|\mathbf{x}, \boldsymbol{\theta})$. This model employs the *fully independent training conditional* (FITC) approximation method [30] in order to handle the large dataset size. In addition, in order to be able to employ GPC, we sample from the original dataset such that the number of instances close to the time steps for which the agent triggers alarms is larger than other time steps. Figure 13a shows the sampled dataset used to build the classification. The red and blue dots respectively indicate the true positive $\mathcal{D}_y = +1$ and false negative $\mathcal{D}_y = -1$ instances. Figure 13b illustrates the probability of true positive anomaly detection, given the absolute value of anomaly heights $|h_{\text{a}}|$ in mm/year, and the anomaly detection duration $\mathtt{N}_{t>t_{\text{s}}}$ in month. The vertical dashed line indicates the maximal anomaly height $|h^{\text{train}}_{\text{a, max}}| = 0.73$ mm/year used to train the RL agent. We conclude that for the anomaly heights $|h_{\text{a}}| \in (0.5, 9.5)$ mm/year, the current agent is able to rightfully detects anomalies with a probability $\Pr(Y = +1 : \text{TP}|\mathbf{x}, \boldsymbol{\theta}) \geq 0.9$.

In Figure 13b, the hollow circle and diamond indicate the first time step when the agent triggers an alarm for the two artificial anomalies as shown in Figure 12a. They correspond to the true positive anomaly detection probability $\Pr(Y = +1 : \text{TP}|\mathbf{x}_*) = 0.4$. The agent identifies the 1st and 2nd artificial anomalies, respectively with the anomaly detection duration of 67 and 59 days. Furthermore, the annual rate of false positive detection is evaluated from a distinct test environment with 3000 episodes containing no anomaly. The current agent shows a false positive ratio of $R^{\text{FP}} = 0.109$/year. This means that in the context of continuous monitoring, the agent is expected to be able to detect anomalies having the magnitude height $|h_{\text{a}}| \in (0.5, 9.5)$ with the probability ≥ 0.9, and to trigger approximately one false alarm per ten years.

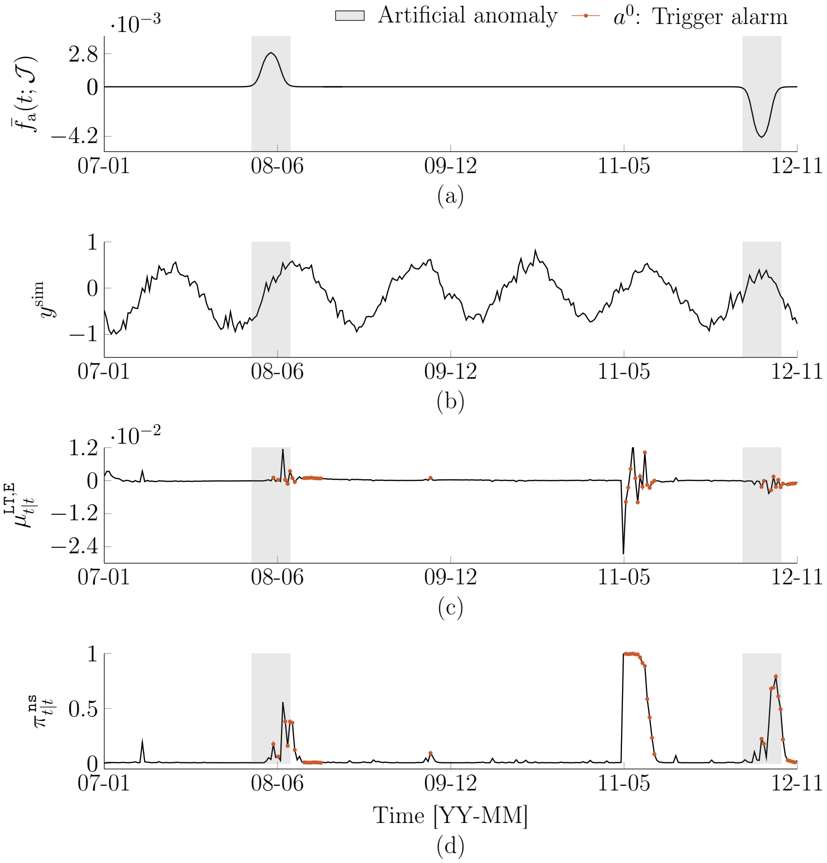


Figure 12: Illustration of the anomaly detection in the presence of artificial anomalies along with the intervention: (a) artificial anomalies with heights 2.8×10^{-3} and -4.2×10^{-3} mm/day, respectively applied before and after the intervention, (b) simulated structural response following the procedure presented in §3.2.2, (c) points of triggering the alarm (solid circles) by the trained agent overlaid with the trend mean value, and (d) points of triggering the alarm (solid circles) by the trained agent overlaid with the probability of non-stationary regime.

We further investigate the anomaly detectability capacity of different agents. Given the same discretized environment and training episodes as employed previously, the main factor affecting agent's anomaly detectability capacity is the reward function values. To this end, five additional agents are trained for different true positive and false positive reward values. Figure 14 illustrates the policies for these agents. Note that, the superscript * in Figure 14d indicates the agent used in order to produce results in Figure 13. Figure 14a-c are corresponding to the agents #1-3 for which we assign a small value for true positive detection reward $r^{\mathrm{TP}} = +2$, while we decrease the penalty for false positive detections $r^{\mathrm{FP}} < 0$. In contrast, Figure 14d-f are associated with the agents #4-6, for which we use a small value for the false positive penalty $r^{\mathrm{FP}} = -2$, while we increase the true positive reward values $r^{\mathrm{TP}} > 0$. In other words, agents #1-3 are penalized more for false

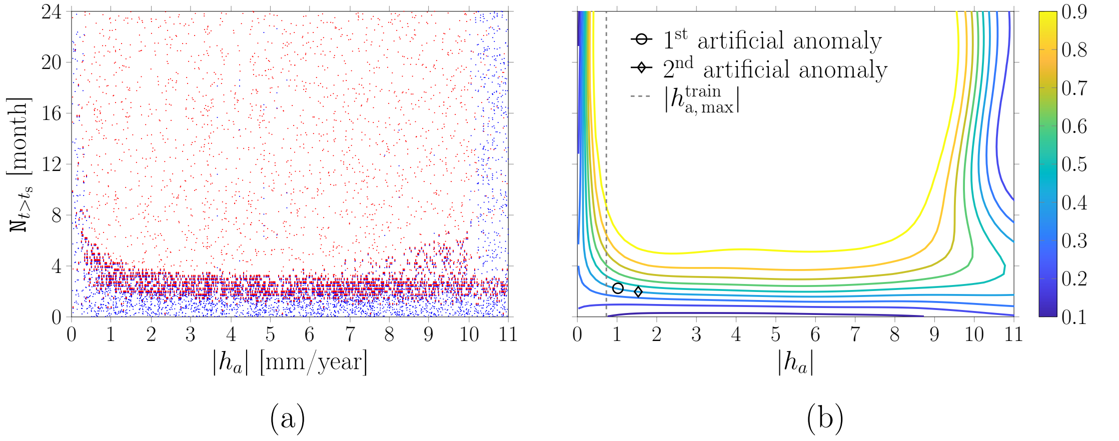
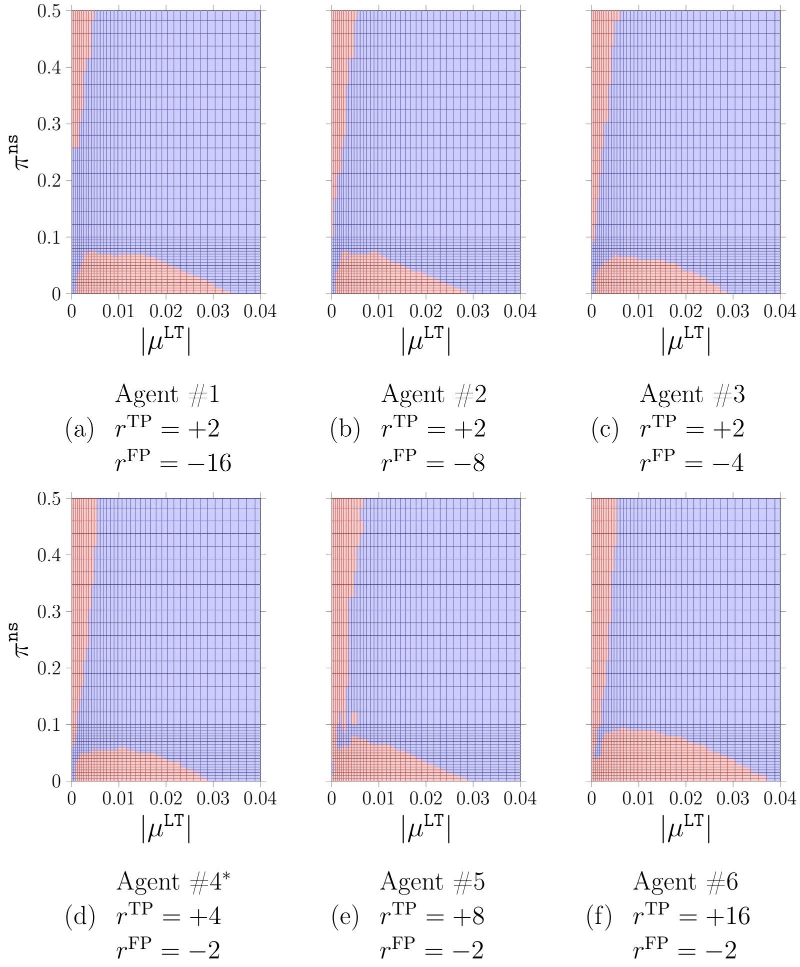


Figure 13: Illustration of the anomaly detectability of the trained agent: (a) the sampled dataset $\mathcal{D}$ for building the classification for which the red and blue dots respectively indicate the true positive $\mathcal{D}_y = +1$ and false negative $\mathcal{D}_y = -1$ samples, and (b) the probability of true positive detection $\Pr(Y = +1 : \text{TP}|\mathbf{x}, \boldsymbol{\theta})$ following the procedure presented in §5.1, given the anomaly heigh and detection duration.

positive detections, while agents #4-6 receive more rewards for true positive detections. Comparing Figures 14a-c reveals that by decreasing the penalty for false positive detections, we increase the agent's preference to trigger alarms for small values of mean trend value and non-stationary regime probabilities. This behavior can be confirmed by the annual rate of false positive R^{FP}, which are respectively 0.005, 0.016, and 0.028 per year for the agents #1-3; as the penalty for the false positive detections reduces, the annual rate of false positive increases. Figures 14d-f display analogous behavior when the agents with higher true positive detection reward prefer to trigger alarms for smaller value of mean trend values and non-stationary regime probabilities. The annual rate of false positive R^{FP} for the agents #5 and #6 are respectively 42 and 49 per year, which correspond to unacceptable false positive ratios.

Decreasing the false positive detection penalty, e.g., Figures 14a-c, or increasing the true positive detection reward, e.g., Figures 14d-f, changes the triggering preference of the agent towards smaller value of non-stationary regime probabilities at the cost of an increase in the annual rate of false positive. Figure 15 illustrates the probability of true positive detection for the anomaly height $|h_\text{a}| \in (0, 2)$ mm/year, corresponding to the agents #1-4. The hollow circles and diamonds correspond to the detection of the two artificial anomalies according to the agent #4 as shown in Figure 13b. The vertical dashed lines indicate the maximum anomaly height used to train the RL agents. As the penalty for false positive detections decreases, the agent tends to trigger alarms *earlier*, after the beginning of the non-stationary regime for $\mathtt{N}_{t>t_s} \leq 5$ months. Note how the probability of true positive detections for the two artificial anomalies increases from 30% to 40% between the agents #1 and #4.



(a) Agent #1
$r^{\text{TP}} = +2$
$r^{\text{FP}} = -16$

(b) Agent #2
$r^{\text{TP}} = +2$
$r^{\text{FP}} = -8$

(c) Agent #3
$r^{\text{TP}} = +2$
$r^{\text{FP}} = -4$

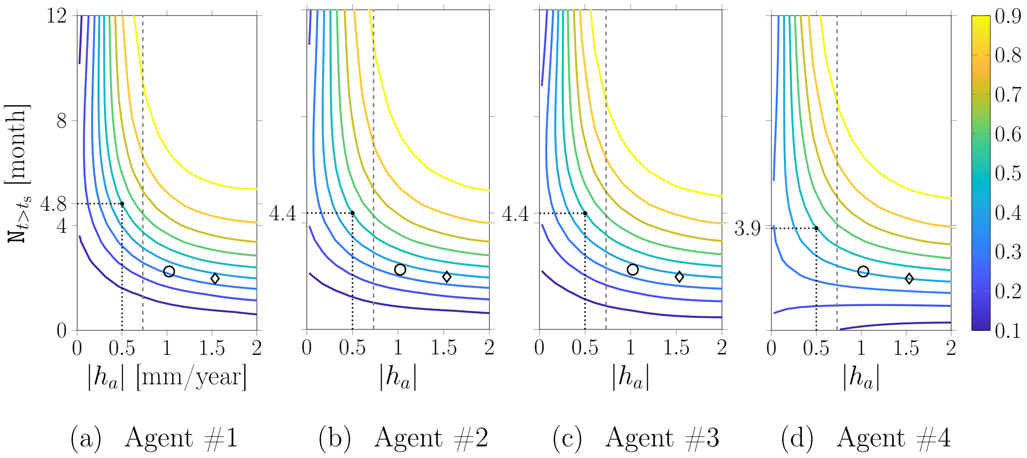
(d) Agent #4*
$r^{\text{TP}} = +4$
$r^{\text{FP}} = -2$

(e) Agent #5
$r^{\text{TP}} = +8$
$r^{\text{FP}} = -2$

(f) Agent #6
$r^{\text{TP}} = +16$
$r^{\text{FP}} = -2$

Figure 14: Policy comparison between different RL agents with respect to the reward function values: (a-c) agents with a small value for true positive detection reward $r^{\text{TP}} = +2$, and an decreasing false positive detections penalties $r^{\text{FP}} < 0$, and (d-f) agents with a small false positive penalty $r^{\text{FP}} = -2$, and an increasing true positive reward values $r^{\text{TP}} > 0$. Note that Agent #4* is the agent used in §6.

Figure 15 shows the number of months after the beginning of the non-stationary regime for which the agent triggers an alarm with the probability 50% for an anomaly heigh $|h_{\text{a}}| = 0.5$ mm/year. Choosing proper reward values for the task of anomaly detection is a tradeoff between how often it rightfully detects anomalies with a desirable anomaly detection duration and probability, as well as the frequency of false positive detections. Figure 16 compares the six agents regarding (i) the number of month required to reach a probability of true positive detection of 50% for an anomaly height $|h_{\text{a}}| = 0.5$ mm/year, and (ii) the annual false positive ratio in logarithmic scale. Choosing among agents #1-4 is a tradeoff between the time to detect anomalies and the false alarm rate. We identify that the agent #1 is responsible for smaller annual false positive ratio compare with the agents #2-4,



(a) Agent #1 (b) Agent #2 (c) Agent #3 (d) Agent #4

Figure 15: Comparison of the probability of true positive detection for the RL agents. The vertical dashed lines indicate the maximum anomaly height used to train the RL agents. The probability of true positive detections for the two artificial anomalies (hollow circles and diamonds) increases from 30% to 40% between the agents #1 and #4.

while resulting in a delay of approximately one month in the anomaly detection duration compared with agent #4. In contrast, the false positive ratios for the agents #5 and #6 are unacceptably high and they cannot detect anomalies within an acceptable time that is useful for SHM. The main reason for the high false positive ratios regarding the agents #5 and #6 is due to the restart criterion during the training, for which the agent-environment interaction stops whenever the action taken by the agent is to trigger an alarm. In other words, agents #5 and #6 collect more negative rewards due to the high annual false positive ratios, which reduces the probability of true positive detections.

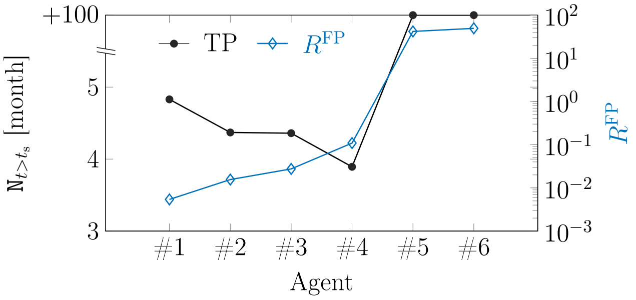


Figure 16: Illustration of the number of month required to reach a probability of true positive detection of 50% for an anomaly height $|h_\mathrm{a}| = 0.5$ mm/year overlaid with the annual false positive ratio. Selecting the agent with respect to the reward function values is a tradeoff between the time to detect anomalies and the false alarm rate: the agent #1 is responsible for smaller annual false positive ratio, while resulting in a delay of nearly one month in the anomaly detection duration.

7 Conclusion

This study proposes an anomaly detection framework that combines the decomposition capacity of BDLM and long-term planning ability of RL. The BDLM is responsible for decomposing structural responses into hidden states, while the RL agent incorporates information from these hidden states in order to make decision based on the expected accumulated discounted rewards. We demonstrate the application of this framework for detecting anomalies on SHM data. Furthermore, the robustness of the framework is examined using artificial anomalies mimicking the real structural responses. The results show that the proposed framework succeeds in detecting anomalies that are small enough to be relevant for real-life applications, while maintaining a tight control on false alarms. This framework allows quantifying the anomaly detectability capacity according to the probability of true positive detection, and the annual rate of false positive detection. The investigation of several agents relying on different reward values reveals that the behavior of the agent depends on the user expectation with respect to the tradeoff between the rightful detection of anomalies and false alarm frequency.

Author Contributions

Shervin Khazaeli: Project Administration, Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing- Original Draft, Visualization. **Luong Ha Nguyen:** Software, Writing- Review & Editing. **James-A. Goulet:** Supervision, Project Administration, Funding acquisition, Resources, Conceptualization, Methodology, Formal analysis, Visualization, Writing- Review & Editing.

Acknowledgements

The authors acknowledge Les Ponts Jacques Cartier et Champlain Incorporée (PJCCI), who funded the project and provided the dataset used in this paper. PJCCI research and applications division works on the application of artificial intelligence methodologies for interpreting data collected on civil infrastructures.

References

[1] Deepak Agarwal. Detecting anomalies in cross-classified streams: A bayesian approach. *Knowl. Inf. Syst.*, 11(1):29–44, December 2006.

[2] Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262:134–147, 2017.

[3] ASCE. 2017 report card for America's infrastructure. Technical report, American Society of Civil Engineers, Washington, 2020.

[4] Onur Avci, Osama Abdeljaber, Serkan Kiranyaz, Mohammed Hussein, Moncef Gabbouj, and Daniel J. Inman. A review of vibration-based damage detection in civil structures:

From traditional methods to machine learning and deep learning applications. *Mech Syst Signal Process*, 147:107077, 2021.

[5] Luciana Balsamo, Raimondo Betti, and Homayoon Beigi. A structural health monitoring strategy using cepstral features. *J Sound Vib*, 333(19):4526 – 4542, 2014.

[6] A. M. Bianco, M. García Ben, E. J. Martínez, and V. J. Yohai. Outlier detection in regression models with arima errors using robust estimates. *J. Forecast.*, 20(8):565–579, 2001.

[7] Ivar Björnsson, Oskar Larsson Ivanov, Dániel Honfi, and John Leander. Decision support framework for bridge condition assessments. *Struct Saf*, 81:101874, 2019.

[8] E. Peter Carden and James M.W. Brownjohn. Arma modelled time-series classification for structural health monitoring of civil infrastructure. *Mech Syst Signal Process*, 22(2):295 – 314, 2008.

[9] Alessio Datteo, Giorgio Busca, Gianluca Quattromani, and Alfredo Cigada. On the use of ar models for shm: A global sensitivity and uncertainty analysis framework. *Reliability Eng. and System Safety*, 170:99–115, 2018.

[10] Nathaniel D. Daw. Chapter 16 - advanced reinforcement learning. In Paul W. Glimcher and Ernst Fehr, editors, *Neuroeconomics (Second Edition)*, pages 299–320. Academic Press, San Diego, second edition edition, 2014.

[11] Charles Farrar and Keith Worden. *Structural Health Monitoring A Machine Learning Perspective*. John Wiley & Sons, 01 2013.

[12] Isaac Farreras-Alcover, Marios K Chryssanthopoulos, and Jacob Egede Andersen. Regression models for structural health monitoring of welded bridge joints based on temperature, traffic and strain measurements. *Struct Health Monit*, 14(6):648–662, 2015.

[13] Ianis Gaudot, Luong Ha Nguyen, Shervin Khazaeli, and James-A. Goulet. Openbdlm, an open-source software for structural health monitoring using bayesian dynamic linear models. In *13th International Conference on Applications of Statistics and Probability in Civil Engineering (ICASP)*, 2019.

[14] M. Gaur, S. Makonin, I. V. Bajić, and A. Majumdar. Performance evaluation of techniques for identifying abnormal energy consumption in buildings. *IEEE Access*, 7:62721–62733, 2019.

[15] Alborz Geramifard, Thomas J. Walsh, Stefanie Tellex, Girish Chowdhary, Jonathan P. How, and Nicholas Roy. A tutorial on linear function approximators for dynamic programming and reinforcement learning. *Foundations and Trends in Machine Learning*, 6(4):375– 454, 2013.

[16] Zoubin Ghahramani and Geoffrey E. Hinton. Variational learning for switching state-space models. *Neural Comput*, 12(4):831–864, 2000.

[17] James-A. Goulet. Bayesian dynamic linear models for structural health monitoring. *Struct Control Health Monit*, 24(12):e2035, 2017.

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

[19] Wei-Hua Hu, De-Hui Tang, Jun Teng, Samir Said, and Rolf Rohrmann. Structural health monitoring of a prestressed concrete bridge based on statistical pattern recognition of continuous dynamic measurements over 14 years. *Sensors*, 18:4117, 11 2018.

[20] Chengqiang Huang, Yulei Wu, Yuan Zuo, K. Pei, and G. Min. Towards experienced anomaly detector through reinforcement learning. In *AAAI*, 2018.

[21] G. R. Kumar, N. Mangathayaru, and G. Narsimha. An approach for intrusion detection using novel gaussian based kernel function. *J. Univers. Comput. Sci.*, 22:589–604, 2016.

[22] Milad Memarzadeh and Matteo Pozzi. Model-free reinforcement learning with model-based safe exploration: Optimizing adaptive recovery process of infrastructure systems. *Struct Saf*, 80:46 – 55, 2019.

[23] Kevin P. Murphy. *Machine learning : a probabilistic perspective.* The MIT Press, 2013.

[24] Luong Ha Nguyen, Ianis Gaudot, Shervin Khazaeli, and James-A. Goulet. A kernel-based method for modeling non-harmonic periodic phenomena in bayesian dynamic linear models. *Front Built Environ*, 5:8, 2019.

[25] Luong Ha Nguyen and James-A. Goulet. Anomaly detection with the switching kalman filter for structural health monitoring. *Struct Control Health Monit*, 25(4):e2136, 2018.

[26] Luong Ha Nguyen and James-A. Goulet. Real-time anomaly detection with bayesian dynamic linear models. *Struct Control Health Monit*, 26(9):e2404, 2019.

[27] Saeed Nozhati, Bruce R. Ellingwood, and Edwin K.P. Chong. Stochastic optimal control methodologies in risk-informed community resilience planning. *Struct Saf*, 84:101920, 2020.

[28] Min-hwan Oh and Garud Iyengar. Sequential anomaly detection using inverse reinforcement learning. In Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis, editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1480–1490. ACM, 2019.

[29] Marco A.F. Pimentel, David A. Clifton, Lei Clifton, and Lionel Tarassenko. A review of novelty detection. *Signal Process*, 99:215 – 249, 2014.

[30] Joaquin Quiñonero Candela and Carl Edward Rasmussen. A unifying view of sparse approximate gaussian process regression. *J Mach Learn Res*, 6:1939–1959, December 2005.

[31] Carl Edward Rasmussen and Hannes Nickisch. Gaussian processes for machine learning (gpml) toolbox. *J Mach Learn Res*, 11:3011–3015, December 2010.

[32] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[33] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, USA, 3 edition, 2009.

[34] Richard S. Sutton. Learning to predict by the methods of temporal differences. *Mach Learn*, 3(1):9–44, August 1988.

[35] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

[36] Elise van der Pol, Daniel E. Worrall, Herke van Hoof, Frans A. Oliehoek, and Max Welling. MDP homomorphic networks: Group symmetries in reinforcement learning. In *NeurIPS*, 2020.

[37] David Vernon. *Machine Vision: Automated Visual Inspection and Robot Vision*. Prentice-Hall, Inc., USA, 1991.

[38] Christopher J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge University, 1989.

[39] Kenji Yamanishi, Jun-Ichi Takeuchi, Graham Williams, and Peter Milne. On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. In *Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 320–324, New York, NY, USA, 2000. ACM.

[40] Gary Yen, Fengming Yang, and Travis Hickey. Coordination of exploration and exploitation in a dynamic environment. *Int J Smart Eng Sysm Design*, 4(3):177–182, 2002.

[41] Mengran Yu and Shiliang Sun. Policy-based reinforcement learning for time series anomaly detection. *Eng. Appl. Artif. Intell.*, 95:103919, 2020.

Appendix A

The model matrices $\{\mathbf{A}_t, \mathbf{C}_t, \mathbf{Q}_t, \mathbf{R}_t\}$ for the stationary and non-stationary model matrices for the bridge case-study are defined following

Stationary model matrices $\mathcal{M}^{\mathbf{s,obs}}$

$$\mathbf{A}_t^{\mathtt{s}} \quad = \quad \text{block diag}\left(\begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & \tilde{\boldsymbol{k}}_t^{\mathtt{KR,E}} \\ \mathbf{0} & \mathbf{I}_{10\times 10} \end{bmatrix}, \phi^{\mathtt{AR,E}}, 1, \begin{bmatrix} 0 & \tilde{\boldsymbol{k}}_t^{\mathtt{KR,T}} \\ \mathbf{0} & \mathbf{I}_{10\times 10} \end{bmatrix}, \phi^{\mathtt{AR,T}}\right)$$

$$\mathbf{C}_t^{\mathtt{s}} \quad = \quad \begin{bmatrix} 1 & 0 & 0 & 1 & \mathbf{0}_{1\times 10} & 1 & 0 & \beta^{\mathtt{KR,E|T}} & \mathbf{0}_{1\times 10} & \beta^{\mathtt{AR,E|T}} \\ 0 & 0 & 0 & 0 & \mathbf{0}_{1\times 10} & 0 & 1 & 1 & \mathbf{0}_{1\times 10} & 1 \end{bmatrix}$$

$$\mathbf{R}_t^{\mathtt{s}} \quad = \quad \text{block diag}\left(\left(\sigma_v^{\mathtt{E}}\right)^2, \left(\sigma_v^{\mathtt{T}}\right)^2\right)$$

$$\begin{aligned}\mathbf{Q}_t^{\mathtt{s(s)}} \quad = \quad \text{block diag}\Bigg(& \left(\sigma_w^{\mathtt{LT}}\right)^2 \begin{bmatrix} \dfrac{\Delta t^3}{3} & \dfrac{\Delta t^2}{2} & 0 \\ \dfrac{\Delta t^2}{2} & \Delta t & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,E}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,E}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,E}}\right)^2, \\ & \left(\sigma_w^{\mathtt{LL}}\right)^2, \begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,T}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,T}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,T}}\right)^2 \Bigg)\end{aligned}$$

$$\begin{aligned}\mathbf{Q}_t^{\mathtt{ns(s)}} \quad = \quad \text{block diag}\Bigg(& \begin{bmatrix} \left(\sigma_w^{\mathtt{LT}}\right)^2 \cdot \dfrac{\Delta t^3}{3} & 0 & 0 \\ 0 & \left(\sigma_w^{\mathtt{LTT}}\right)^2 \cdot \Delta t & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,E}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,E}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,E}}\right)^2, \\ & \left(\sigma_w^{\mathtt{LL}}\right)^2, \begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,T}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,T}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,T}}\right)^2 \Bigg)\end{aligned}$$

Non-stationary model matrices $\mathcal{M}^{\mathbf{ns,obs}}$

$$\mathbf{A}_t^{\mathtt{ns}} = \text{block diag}\left(\begin{bmatrix} 1 & \Delta t & \dfrac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & \tilde{\boldsymbol{k}}_t^{\mathtt{KR,E}} \\ \mathbf{0} & \mathbf{I}_{10\times 10} \end{bmatrix}, \phi^{\mathtt{AR,E}}, 1, \begin{bmatrix} 0 & \tilde{\boldsymbol{k}}_t^{\mathtt{KR,T}} \\ \mathbf{0} & \mathbf{I}_{10\times 10} \end{bmatrix}, \phi^{\mathtt{AR,T}}\right)$$

$$\mathbf{C}_t^{\mathtt{ns}} = \begin{bmatrix} 1 & 0 & 0 & 1 & \mathbf{0}_{1\times 10} & 1 & 0 & \beta^{\mathtt{KR,E|T}} & \mathbf{0}_{1\times 10} & \beta^{\mathtt{AR,E|T}} \\ 0 & 0 & 0 & 0 & \mathbf{0}_{1\times 10} & 0 & 1 & 1 & \mathbf{0}_{1\times 10} & 1 \end{bmatrix}$$

$$\mathbf{R}_t^{\mathtt{ns}} = \text{block diag}\left(\left(\sigma_v^{\mathtt{E}}\right)^2, \left(\sigma_v^{\mathtt{T}}\right)^2\right)$$

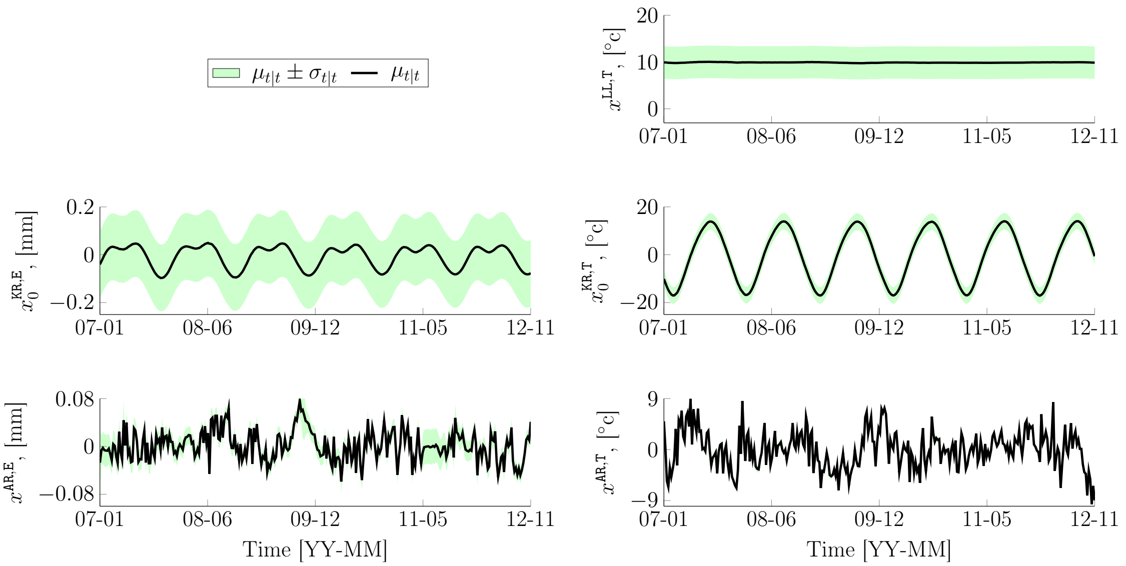
$$\begin{aligned}\mathbf{Q}_t^{\mathtt{s(ns)}} = \text{block diag}&\left(\begin{bmatrix} \left(\sigma_w^{\mathtt{LA}}\right)^2 \cdot \dfrac{\Delta t^5}{20} & 0 & 0 \\ 0 & \left(\sigma_w^{\mathtt{LTT}}\right)^2 \cdot \dfrac{\Delta t^3}{3} & 0 \\ 0 & 0 & \left(\sigma_w^{\mathtt{LA}}\right)^2 \cdot \Delta t \end{bmatrix},\right. \\ &\begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,E}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,E}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,E}}\right)^2, \left(\sigma_w^{\mathtt{LL}}\right)^2, \begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,T}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,T}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,T}}\right)\end{aligned}$$

$$\begin{aligned}\mathbf{Q}_t^{\mathtt{ns(ns)}} = \text{block diag}&\left(\left(\sigma_w^{\mathtt{LA}}\right)^2 \begin{bmatrix} \dfrac{\Delta t^5}{20} & \dfrac{\Delta t^4}{8} & \dfrac{\Delta t^3}{6} \\ \dfrac{\Delta t^4}{8} & \dfrac{\Delta t^3}{3} & \dfrac{\Delta t^2}{2} \\ \dfrac{\Delta t^3}{6} & \dfrac{\Delta t^2}{2} & \Delta t \end{bmatrix}, \begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,E}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,E}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,E}}\right)^2,\right. \\ &\left.\left(\sigma_w^{\mathtt{LL}}\right)^2, \begin{bmatrix} \left(\sigma_{w,0}^{\mathtt{KR,T}}\right)^2 & \mathbf{0} \\ \mathbf{0} & \left(\sigma_{w,1}^{\mathtt{KR,T}}\right)^2 \cdot \mathbf{I}_{10\times 10} \end{bmatrix}, \left(\sigma^{\mathtt{AR,T}}\right)^2\right)\end{aligned}$$

where $\tilde{\boldsymbol{k}}_t^{\mathtt{KR}} = [\tilde{k}_{t,1}^{\mathtt{KR}}, \tilde{k}_{t,2}^{\mathtt{KR}}, \cdots, \tilde{k}_{t,10}^{\mathtt{KR}}]$ is the normalized kernel values and Δt is the time step at the time t.

$$\mathcal{P}^* = \left\{\underbrace{0.99590, 0.99912}_{\text{regime transition}}, \underbrace{0, 7.33\times 10^{-4}, 0, 1.00, 0, 3.93\times 10^{-1}, 2.67\times 10^{-2}, 6.46\times 10^{-5}}_{\text{elongation}, \mathtt{E}}, \underbrace{3.66\times 10^{-2}, 4.14\times 10^{-2}}_{\text{dependency}, \mathtt{E|T}}, \underbrace{0, 6.54\times 10^{-1}, 0, 3.88\times 10^{-1}, 2.58, 0.16}_{\text{temperature}, \mathtt{T}}\right\}$$

Figure 17 illustrates the hidden state variable estimation using BDLM.



(a) Elongation hidden states

(b) Temperature hidden states

Figure 17: Hidden state variables estimation for elongation and temperature observations: (a) the kernel regression $x_0^{\mathtt{KR}}$ and auto-regressive $x^{\mathtt{AR}}$ hidden state variables for the elongation observations, and (b) the local level $x^{\mathtt{LL}}$, kernel regression $x_0^{\mathtt{KR}}$, and auto-regressive $x^{\mathtt{AR}}$ hidden state variables for the temperature observations.

Appendix B

Table 3 shows the distributions and the parameters of the anomaly function stochastic characteristic set $\mathcal{J}$ for training the agent. Here, we simulate the environment's state for the positive anomaly heights for the sake of computational efficiency. In Table 3, the upper bound for the anomaly height during anomaly detectability quantification is 4×10^{-2}.

Table 4 depicts the agents' configuration during the training. Agent #4 is the one used in this paper for the bridge anomaly detection, and Q_0 is the initial Q-value selected for all the environment's states.

Table 3: Environment's state space simulation parameters.

Parameter	Value		Unit
	a	b	
$h_{\mathrm{a}} \sim \mathcal{U}(h_{\mathrm{a}}; a, b)$	1×10^{-5}	2×10^{-3}	[mm/day]
$w_{\mathrm{a}} \sim \mathcal{U}(w_{\mathrm{a}}; a, b)$	60	180	[day]
$t_{\mathrm{c}} \sim \mathcal{U}(t_{\mathrm{c}}; a, b)$	Jan 1, 2007	Nov 11, 2012	[date]
w_0	constant:	365	[day]
$\epsilon_{\mathrm{a,min}}$	constant:	1×10^{-6}	[mm/day]
ε	constant:	1×10^{-7}	[mm/day]

Table 4: Agent #4 configuration.

Parameter	Value
r^{TP}	+4
r^{FP}	-2
r^{TN}	+2
η	2
c_α	5
c_ϵ	5
Q_0	35
γ	0.97