Reliability Engineering and System Safety 239 (2023) 109496

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Bridging POMDPs and Bayesian decision making for robust maintenance planning under model uncertainty: An application to railway systems

Giacomo Arcieri [a,*], Cyprien Hoelzl [a], Oliver Schwery [b], Daniel Straub [c], Konstantinos G. Papakonstantinou [d], Eleni Chatzi [a]

[a] Institute of Structural Engineering, ETH Zürich, 8093 Zürich, Switzerland
[b] Swiss Federal Railways SBB, 3000 Bern, Switzerland
[c] Engineering Risk Analysis Group, Technical University of Munich, 80333 Munich, Germany
[d] Department of Civil and Environmental Engineering, Pennsylvania State Univ., University Park, PA 16802, USA

ARTICLE INFO

Keywords:
Partially observable Markov decision processes
Bayesian inference
Optimal maintenance planning
Model uncertainty
Hidden Markov models
Dynamic Programming

ABSTRACT

Structural Health Monitoring (SHM) describes a process for inferring quantifiable metrics of structural condition, which can serve as input to support decisions on the operation and maintenance of infrastructure assets. Given the long lifespan of critical structures, this problem can be cast as a sequential decision making problem over prescribed horizons. Partially Observable Markov Decision Processes (POMDPs) offer a formal framework to solve the underlying optimal planning task. However, two issues can undermine the POMDP solutions. Firstly, the need for a model that can adequately describe the evolution of the structural condition under deterioration or corrective actions and, secondly, the non-trivial task of recovery of the observation process parameters from available monitoring data. Despite these potential challenges, the adopted POMDP models do not typically account for uncertainty on model parameters, leading to solutions which can be unrealistically confident. In this work, we address both key issues. We present a framework to estimate POMDP transition and observation model parameters directly from available data, via Markov Chain Monte Carlo (MCMC) sampling of a Hidden Markov Model (HMM) conditioned on actions. The MCMC inference estimates distributions of the involved model parameters. We then form and solve the POMDP problem by exploiting the inferred distributions, to derive solutions that are robust to model uncertainty. We successfully apply our approach on maintenance planning for railway track assets on the basis of a "fractal value" indicator, which is computed from actual railway monitoring data.

1. Introduction

Engineering infrastructures are subject to deterioration processes, which undermine a safe utilization and incur economic and environmental costs. Maintenance policies aim to extend the operating life-cycle, by seeking a trade-off between compromise in structural condition and the costs associated to repair and intervention actions. Structural Health Monitoring (SHM) contributes toward this goal by delivering data-driven indicators of structural condition, and/or by allowing to update and refine predictive models of operating engineered systems [1]. The extracted information can support maintenance planning to achieve the long-term objectives of cost and risk minimization throughout the structural life-cycle. To this end, a probabilistic risk-based decision framework for SHM is outlined in [2]. Linked to SHM is the concept of Value of Information (VoI) or Value of Structural Health Monitoring [3–6], which quantifies the cost benefits associated with adoption of monitoring tools.

Cost efficient maintenance is crucial for effective management of extended infrastructure networks, as represented for instance in the case of railway systems. As a characteristic example, Switzerland's railway network usage and load have increased by roughly 40% and 70%, respectively, in the last 30 years, while the amount of traffic per km of track is the highest worldwide [7]. This increased backlog demand has led to higher life-cycle costs and an increase in disruptive events. However, infrastructure asset management has to obey budgetary, availability, and further constraints. As a result, new, efficient approaches for maintenance scheduling are needed to address modern challenges. In formalizing the approach to maintenance planning, it is possible to cast this as a sequential decision-making problem with a long horizon cost minimization objective [8]. Current decisions will

* Corresponding author.
E-mail addresses: giacomo.arcieri@ibk.baug.ethz.ch (G. Arcieri), hoelzl@ibk.baug.ethz.ch (C. Hoelzl), oliver.schwery@sbb.ch (O. Schwery), straub@tum.de (D. Straub), kpapakon@psu.edu (K.G. Papakonstantinou), chatzi@ibk.baug.ethz.ch (E. Chatzi).

https://doi.org/10.1016/j.ress.2023.109496
Received 22 December 2022; Received in revised form 30 June 2023; Accepted 4 July 2023
Available online 7 July 2023

bear an impact on the system's future condition, which – in absence of intervention – tends to stochastically evolve according to a degradation process. There is, however, significant uncertainty associated to the estimate of a system's condition, both at present and in the future. SHM offers a tool for more reliably tracking the system's state (condition), thus reducing the associated uncertainty. However, monitoring measurements come in the form of noise-corrupt information, which only approximate the actual structural state. This problem admits representation in the form of a Partially Observable Markov Decision Process (POMDP). The POMDP framework utilizes the uncertain available information along with a (transition) model of the stochastic evolution of the system, to derive solutions with mathematically sound optimality properties [9]. POMDPs have already been successfully implemented for solving optimal maintenance planning problems of corroding reinforced concrete structures [10], interstate highway pavements [11], wind turbines [12], deteriorating bridges [13], regenerative air heater in power plants [14], or oil and gas pipelines [15]. While POMDP solutions have long been limited to small-scale problems, it has recently been shown that the framework can be efficiently extended to more complex problems [10].

Nevertheless, there is currently scarce adoption and available literature of POMDP solutions for real-world applications. The framework requires knowledge of the stochastic transition dynamics of the structure as well as of the observation generating process. Such models are rarely available in the framework of infrastructure maintenance planning, but could be estimated from available data. However, the recovery of the involved transition dynamics and the associated observation model can be quite complex, while only scarce literature is available on best practices, as stated in [10]. As one of few examples, Papakonstantinou et al. [10] exploit a physical model, described in detail in [16], in order to recover the state transition probability matrix for the deterioration process (i.e., action do-nothing, as explained in Section 4). However, the transition matrices for the repair actions, as well as the observation model, have not been derived from actual data. The authors themselves stress the need for further studies on recovering observation models and transition models for maintenance actions. Song et al. [17] infer the time-dependent deterioration transition matrices by assuming different models, whose parameters estimate via a maximum likelihood approach. However, the methods assume knowledge of the hidden states to then compute the transitions. In addition, in their work the transition matrices for maintenance actions are not inferred, while the inference of the observation function is restricted to the discrete case. Wari et al. [15] infer the deterioration transition matrix from actual data by first computing transition intensities, then forming the matrix by means of a Markov pure birth process. Such an approach does not offer a quantification of the uncertainty over the inferred parameters. Here as well, the inference of the transition matrices for repair actions is not similarly considered. Guo et al. [18] propose the use of the Baum–Welch algorithm for the POMDP model parameter estimation, subsequently exploited to optimize the timing of the inspections. However, the proposed methods do not involve any form of model uncertainty quantification. In general, the majority of applications of POMDPs on infrastructure maintenance planning concern illustrative examples, often of simplified nature. Albeit these works are valuable, this reflects a lack of applications on real-world data, which would often necessitate inferring the transition dynamics relative to the deterioration process and maintenance actions, along with the associated observation model, entirely from data. This creates a gap between development of effective solution algorithms and their actual deployment to real-world applications.

A main contribution of this work is to cover the aforementioned gap by formulating a framework that estimates directly and entirely from real-world data both the POMDP transition and observation models, via MCMC sampling from a Hidden Markov Model (HMM) conditioned on actions. We demonstrate the implementation of our approach based on a real-world problem of optimal maintenance planning for a railway network. Our inference technique can recover the full distributions of parameters, which represent all plausible values the model can assume under the available data. To this end, we exploit the "fractal values" indicator, collected across Switzerland's railway network and described in detail in Section 3. While we focus on this specific application, the presented methods are general and applicable across a broader suite of problems. To the best of our knowledge, no other works demonstrate inference of the complete POMDP model entirely from real-world data.

A further critical point that prevents broad adoption in real-world applications, which is however only secondary to the inference of the complete POMDP model, is that POMDP solutions do not usually account for epistemic uncertainty [19]. Indeed, POMDP solutions are globally optimal for an assumed a-priori model structure, but this is unlikely to coincide with the actual environment (ground truth). As a result, POMDP solutions can be insufficiently robust against model uncertainty, often causing concerns when deployed on real-world applications. The work in [20] casts POMDPs into a fully Bayesian framework, but there is scarce literature on considering epistemic uncertainty in POMDP applications, with [12,21,22] comprising few exceptions. Here, we build on these prior works to further bridge POMDPs and Bayesian decision making by considering model parameter distributions that are inferred by MCMC sampling. As a result, the computed POMDP solutions are not optimized for specific parameters but for all plausible values and are thus robust over model uncertainty.

The remainder of this paper is organized as follows. The next section provides the POMDP theoretical background. Section 3 describes the "fractal values" indicator, namely the data used in this paper to recover transition and observation models, while Section 4 explains how the problem of railway maintenance planning can be cast into the POMDP framework. Section 5 illustrates the inference of the underlying transition and observation models, Section 6 presents the algorithms employed to derive policies that are robust to epistemic uncertainty and summarizes the results, and finally, Section 7 concludes this work.

2. Background and fundamentals

2.1. Markov decision process

A Markov Decision Process (MDP) provides the mathematical framework for modeling a sequential decision making problem within a stochastic control setting. A MDP is defined by the tuple $\langle S, A, R, T, H, \gamma \rangle$, where:

- S is the finite set of states that the environment can assume.
- A is the finite set of actions that the decision maker (or agent) can pick.
- $R : S \times A \to \mathbb{R}$ is the reward function that assigns the reward $r_t = R(s_t, a_t)$ for assuming an action a_t at state s_t.
- $T : S \times S \times A \to [0, 1]$ is the transition dynamics model that consists of the probability $p(s_{t+1}|s_t, a_t)$.
- H is the considered planning horizon of the problem.
- γ is the discount factor.

The objective of the MDP is to determine the optimal policy $\pi^* : S \to A$, which maps states to actions such that the expected sum of rewards is maximized:

$$J(\pi^*) = \max_{\pi} \mathbb{E}\left[\sum_{t=0}^{H} \gamma^t r_t\right] \tag{1}$$

where $r_t = R(s_t, \pi(s_t))$ and $\mathbb{E}[\cdot]$ is the expectation operator.

An MDP can be represented as a special case of influence diagrams [23,24]; which form a class of probabilistic graphical models. Fig. 1 illustrates the graphical model for a general MDP. Circles, rectangles and diamonds correspond to random, decision and utility variables, respectively [25]. Shaded shapes denote observed variables, while edges indicate dependencies among variables.

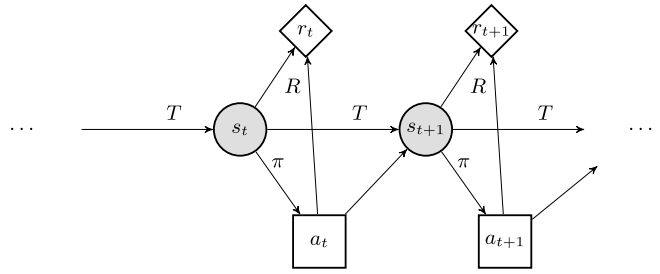


Fig. 1. Probabilistic graphical model of a MDP.

An MDP is assumed to satisfy the Markov property [26], i.e., $p(s_{t+1}|s_t, a_t, \dots, s_0, a_0) = p(s_{t+1}|s_t, a_t)$. If a process does not satisfy the Markov property, the problem may still be modeled as an MDP by state augmentation [9]. In such an approach, the state vector s_t is augmented to further include previous information so that the Markov property is satisfied. Likewise, time can be encoded in the state, allowing to model non-stationary problems and to transform finite horizon problems into infinite ones.

The MDP problem can be solved via Dynamic Programming techniques [27] and the introduction of the value function $V^{\pi} : S \to \mathbb{R}$, which represents the expected sum of rewards of policy π from a certain state. The optimal policy π^* can be computed through Bellman's optimality equation:

$$V_n^{\pi^*}(s_t) = \max_{a_t \in A} \left[R(s_t, a_t) + \gamma \sum_{s_{t+1} \in S} p(s_{t+1}|s_t, a_t) V_{n-1}^{\pi^*}(s_{t+1}) \right] \tag{2}$$

Eq. (2) can be solved with the value iteration algorithm [28]. For the finite horizon problem, n is the number of remaining steps to reach horizon H, i.e., the algorithm operates backwards, initiating at the last time step and identifying the optimal actions for all preceding steps. For the infinite horizon case, n represents the iteration step of the algorithm. If the transition probabilities in Eq. (2) are not known, state-values (and, hence, an optimal policy) can be learned with reinforcement learning via temporal difference methods [29].

Bellman's equation can alternatively be written in terms of the Q-value function [30]:

$$Q_n^{\pi^*}(s_t, a_t) = R(s_t, a_t) + \gamma \sum_{s_{t+1} \in S} p(s_{t+1}|s_t, a_t) V_{n-1}^{\pi^*}(s_{t+1}) \tag{3}$$

which outputs the state-value for taking action a_t at state s_t and then following the optimal policy π^*. The state-value function can then be obtained by choosing the action that maximizes the Q-value function:

$$V_n^{\pi^*}(s_t) = \max_{a_t \in A} \left[Q_n^{\pi^*}(s_t, a_t) \right] \tag{4}$$

2.2. Partially observable Markov decision process

A POMDP extends the MDP framework by incorporating uncertainty into the observations. The states are now hidden variables, which generate observations that provide partial and/or noisy information about the actual state of the system. A POMDP is thus defined by the tuple $\langle S, A, Z, R, T, O, b_0, H, \gamma \rangle$, where the newly introduced variables are:

- Z is the set of possible observations.
- $O : S \times A \times Z \to \mathbb{R}$ is the observation generating process that defines the emission probability $p(z_t|s_t, a_{t-1})$.
- b_0 is the initial belief on the state of the system s_0, with the belief variable defined in what follows.

Given the partial information that the observations provide, the agent should take actions based on the full observation history, which would violate the Markov property. As such, a new variable is introduced in the POMDP setting: the belief state b. The belief is a probability distribution over S, which maps the discrete finite set of states into a continuous $|S| - 1$ dimensional simplex [9]. The belief over the state of the system is updated every time the agent receives a new observation according to Bayes' rule:

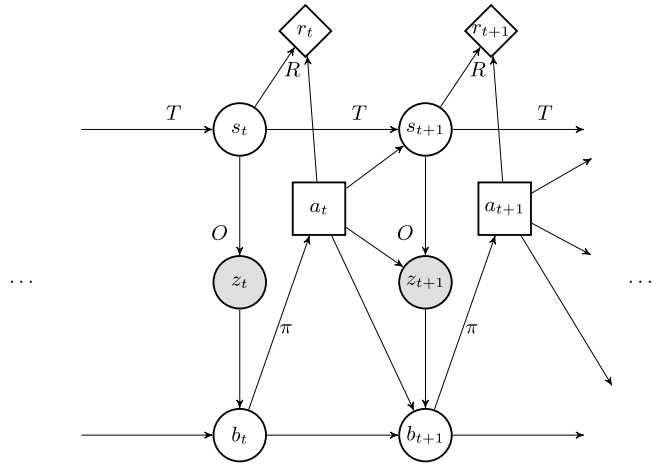


Fig. 2. Probabilistic graphical model of a POMDP.

$$b(s_{t+1}) = \frac{p(z_{t+1}|s_{t+1}, a_t)}{p(z_{t+1}|\mathbf{b}, a_t)} \sum_{s_t \in S} p(s_{t+1}|s_t, a_t) b(s_t) \tag{5}$$

where the denominator is the usual normalizing factor:

$$p(z_{t+1}|\mathbf{b}, a_t) = \sum_{s_{t+1} \in S} p(z_{t+1}|s_{t+1}, a_t) \sum_{s_t \in S} p(s_{t+1}|s_t, a_t) b(s_t) \tag{6}$$

The belief over the state of the system at time t offers sufficient statistics of the full history of actions and observations, namely it provides the decision maker with the same amount of information. The decision maker can then follow a policy $\pi(\mathbf{b})$, which depends on the computed belief, and the POMDP framework thus satisfies the Markov property. The probabilistic graphical model of the POMDP is provided in Fig. 2, whereby state variables are no longer observed, but are hidden variables.

The previously defined Bellman equation changes accordingly to:

$$V_n^{\pi^*}(\mathbf{b}) = \max_{a_t \in A} \left[\sum_{s_t \in S} b(s_t) R(s_t, a_t) + \gamma \sum_{z_{t+1} \in Z} p(z_{t+1}|\mathbf{b}, a_t) V_{n-1}^{\pi^*}(\mathbf{b}') \right] \tag{7}$$

where $\mathbf{b}'$ is the updated belief, which is computed according to Eq. (5).

Solving a POMDP is thus equivalent to solving a continuous state MDP defined over the belief space. While it is still possible to provide optimality convergence properties of the value iteration algorithm thanks to the piecewise linear convex property [9], the exact solution is generally intractable except for very low-dimensional problems. As such, in the literature POMDP solution methods have been relying on approximations. The advent of point-based value iteration algorithms allowed to efficiently solve large scale POMDP problems with good approximation, although they generally require S, A, and Z to be finite. An introduction to these methods is provided in [31,32].

2.3. Bayesian decision making

In the previous sections, we introduced the transition dynamics T and the observation generating process O. These models generally depend on some parameters θ. In existing literature, these are typically treated as fixed. However, in many applications these parameters can be subject to uncertainty, often due to the limited amount of data used for learning, leading to epistemic uncertainty. To tackle this,

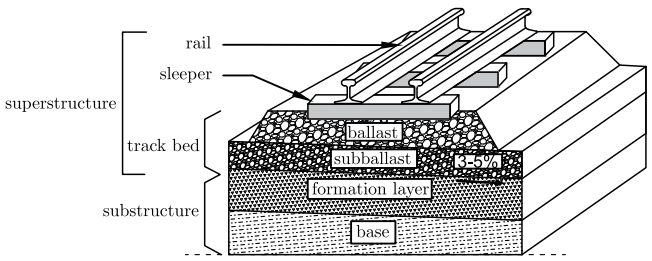


Fig. 3. Structure of the railway track.

a number of works [12,19,20] cast the sequential decision-making problem of a POMDP into a fully Bayesian framework. Indeed, while the POMDP framework is inherently Bayesian, due to the update of the belief variable through Bayes theorem, the scheme is not generally treated as a fully Bayesian framework, since POMDP parameters are not considered as random variables $p(\theta)$, thus failing to incorporate model uncertainty into the solution.

In Bayesian decision theory [33] the concept of utility function $U(\theta, a)$ is introduced, which maps possible outcomes to their utility given the parameters θ and some decision a. The Bayesian optimal action is the one which maximizes the expected utility:

$$a^* = \arg\max_{a \in A} \mathbb{E}_{\theta \sim p(\theta)} [U(\theta, a)] \tag{8}$$

In the MDP framework, the concept of utility is associated with the Q-values. We denote $Q^{\pi}_{\theta}(s, a)$ as the Q-value for action a when the model parameters are $\theta \sim p(\theta)$. In this fully Bayesian context, the optimal action thus maximizes the expected Q-value function over the model parameter distribution [12]:

$$a^* = \arg\max_{a \in A} \mathbb{E}_{\theta \sim p(\theta)} \left[Q^{\pi^*}_{\theta}(s, a) \right] \tag{9}$$

In this setting, the optimal policy may be sub-optimal for a specific value θ, while maximizing the expected value with respect to the entire model parameter distribution. As a result, the policy is robust over epistemic uncertainty.

3. Data description

Although our suggested techniques are generally applicable, the focus application in this work is related to maintenance planning for railway track infrastructure. The latter forms an assembly of multiple components (rails, sleepers, ballast, switches, etc.), as illustrated in Fig. 3, which are exposed to harsh environments and high loads, leading to accelerated degradation. The durability of the railway track, as well as its renewal costs are strongly dependent on the condition of certain components, such as the substructure. The substructure plays an essential role in the degradation process of the track, as the substructure material sustains cyclic loading from the superstructure, acts as a filter that blocks the uprising of fine particles into the ballast, and facilitates water drainage. A weakened substructure will typically result in distortions of the track geometry. Tamping, a maintenance action involving the usage of compacting devices to pack the ballast under the railway track, is often applied when the substructure condition is deemed moderately deteriorated. When only the superstructure is degraded (ballast fouling) the preferred maintenance measures are ballast cleaning or replacement. If the substructure is in poor condition (intrusion of clay or mud, water clogging, etc.), tamping or superstructure maintenance can only provide a short-term remedy, leaving replacement of the superstructure and substructure as the most appropriate long-term solution. Clearly, the optimization of maintenance decisions for such critical infrastructure components would benefit from information that is additional to scheduled inspection. Such additional information can be delivered from monitoring data derived by diagnostic vehicles. In this work, we specifically exploit the *fractal values*, a substructure condition indicator derived from diagnostic vehicle measurements to guide decisions for substructure renewal.

Such diagnostic vehicles form part of modern practice in the management of infrastructure assets. In the domain of railway infrastructure predictive or reactive maintenance and renewal decisions are increasingly guided by data-supported decision tools, such as the SwissTamp platform of the Swiss Federal Railways [7]. Periodic inspection is carried out by means of diagnostic measurement vehicles that are equipped with a multitude of sensors (cameras, accelerometers, laser-distometers, etc.). Amongst the diverse portfolio of collected information, the track geometry measurements, in particular, deliver condition indicators that are readily exploited for the network-wide estimation of the ballast and substructure condition [34].

A specific set of such condition indicators are the so-called fractal values, which are derived from the longitudinal level measurement. The longitudinal level represents the vertical smoothness of the rail and is measured via a diagnostic vehicle as the deviation of the running surface of the rail from the smoothed vertical position [35]. Fractal values are the outcome of fractal analysis. The fractal dimension corresponds to the ratio between the change in the details in a pattern with respect to the change in the measurement scale. For railway tracks the fractal dimension corresponds to the degree of "roughness" at varying wavelength scales. For the interested reader, the detailed steps of the fractal value computation are reported in Algorithm 3 of the Appendix, which was devised by Matthias Landgraf [36]. The fractal values are now used in practice by the Austrian and Swiss railways to detect ballast and substructure damage [36]. Mid-wave (3–25 m) fractal values have been shown to have a higher correlation to ballast degradation, while long-wave (25–70 m) fractal values are more related to substructure damages [37].

A visual example is offered in Fig. 4, which displays a highly deteriorated portion of a track in 2014. The area shows presence of clay, fine material (fouling), and water intrusion, which represent characteristic problems of ballast and substructure damages. The figure also reports fractal value data that has been collected over the same area from 2012 and 2015. As a result of the deterioration of the track, fractal values decrease over time. In the damaged area (km 25.5 in figure), the fractal values have dropped considerably in the examined time-frame, suggesting the severe degradation confirmed by the inspection.

In this work, we use actual track geometry measurements, carried out by the SBB (the Swiss Federal Railways) between 2008 and 2018, across Switzerland's railway network, for tracks whose superstructure or substructure were subsequently maintained in 2019 [37]. The track geometry measurements were collected at least twice a year for the tracks under investigation. The fractal values are computed every 2.5 m from the measured longitudinal level. The performed maintenance actions have been additionally logged for the analyzed tracks. These logs contain information on the maintenance, repair, or renewal actions taken on a section of the network at a specific date. We propose herein a unique POMDP scheme, which relies on diagnostic vehicle measurements of long-wave fractal values to predict an optimal maintenance policy.

4. POMDP modeling

Within our application of railway maintenance planning, the POMDP problem is defined by the following variables:

- *Hidden states*, which represent the health condition of the track. We assume 4 hidden states: s_0, s_1, s_2, and s_3, which can be seen as *very good*, *good*, *bad*, and *very bad* track conditions, respectively. The choice of the number of hidden states is eventually arbitrary, since ground truth is not available. However, we adopted a pragmatic approach for determining the dimension of hidden states, by assuming this as a hyperparameter and repeating the inference

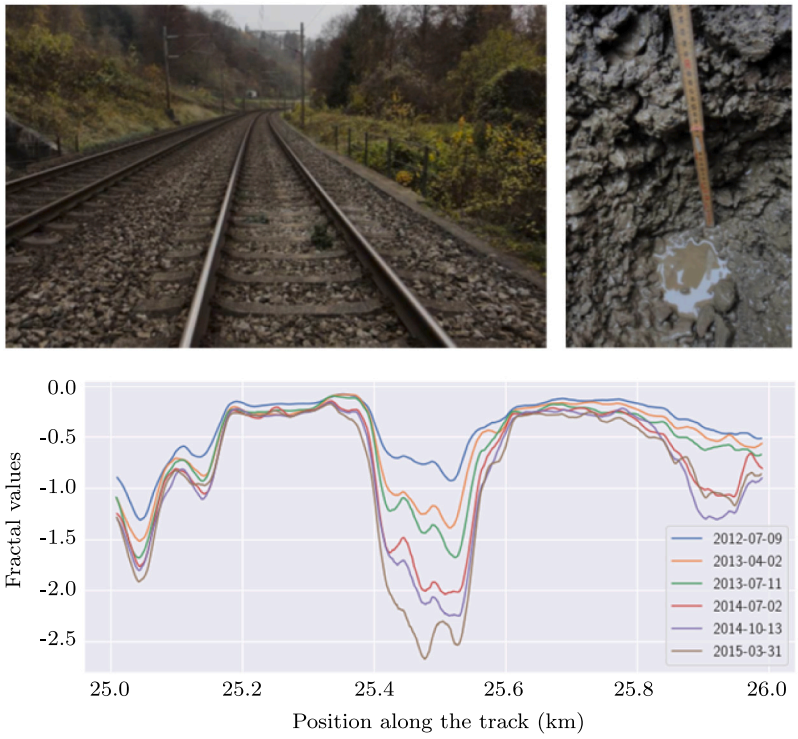


Fig. 4. A highly deteriorated track from an inspection in 2014. The upper left image shows an overview of the track at the deteriorated location. The right image shows a severely degraded portion of the track at km 25.5, with presence of fouling, clay, and water intrusion. The bottom left image shows the associated long-wave fractal values. The decreasing values around km 25.5 indicate the deterioration of the area over time.

of the model presented in the next section for 3, 4 and 5 hidden states. The model with 4 hidden states yielded improved convergence and better-defined distributions. In addition, 4 discrete condition states are assumed in similar works [10].

- *Actions*, represented by the possible maintenance actions. We focus on 3 possible actions among the ones recorded in our available data. a_0 represents the do-nothing action, i.e., the agent chooses not to take any maintenance at this decision step. The effect of a_0 is governed by the degradation process. a_1 is a low cost "tamping" action, which is often conducted as part of standard ballasted track maintenance. Tamping vehicles are commonly used to restore the geometry of ballasted tracks in a nearly automatic fashion [38]. Finally, a_2 is a more costly repair action, which involves the renewal of the substructure plus maintenance similar to a_1. In the offered case study, we demonstrate how the effects of a_1 and a_2 can be learned on the basis of the efficacy of these repair actions.
- *Observations*, defined by the fractal values. The decision maker forms a belief over the state condition of the track, on the basis of the fractal values indicator, and makes a decision to follow one of the aforementioned actions. Fractal values comprise negative, continuous values which tend to decrease if no maintenance action is taken. The fractal values observed in our actual data, over the averaged observation lengths, reflect a clear negative trend, which motivate an attempt to model these observations as dependent on the previous value in order to ensure temporal coherence, introducing an autoregressive property among observations. Practical examples of the need for this property are given in the next section.
- *(Negative) rewards*, representing costs associated with actions and states. Typically, the costs of actions can be defined by the infrastructure operator. Quantifying the cost of different states is a far more difficult task. It should include costs and economic risks such as the deterioration of service due to imperfect track conditions, delays, environmental costs, working accidents or derailing risks. Hence, these costs are hard to quantify but crucial to justify maintenance expenses. We discussed both classes of costs with our SBB partners and report them in Table 1 in general cost units, although only cost ratios matter for the solution of the problem. The action do-nothing does not have any cost. Action a_1 costs 50 (units) regardless of the condition of the track. The cost of the renewal part of action a_2 varies from 2000 to 4000 units depending on the condition of the structure, plus 50 units due to the tamping action.

Table 1
Costs of the POMDP model.

State condition	s_0	s_1	s_2	s_3
Maintenance action				
a_0	0	0	0	0
a_1	−50	−50	−50	−50
a_2	−2,050	−2,710	−3,370	−4,050
Condition cost	−100	−200	−1,000	−8,000

The influence diagram reflecting the graphical representation of the described railway maintenance problem is shown in Fig. 5. Compared to Fig. 2, this graphical model presents arrows between observation variables displaying the autoregressive dependency. Autoregressive hidden Markov models (ARHMMs) have been deeply studied in the literature [39] with application that range from the modeling of wind time-series [40], to fault detection and prognostics tasks [41]. Similarly to our case, both works exploited ARHMMs to capture the switching between different internal states, while ensuring temporal coherence on observations stemming from sensor measurements. This is crucial in our case given the continuous nature of the derived fractal value measurements. This extension emphasizes the high flexibility of probabilistic graphical models, when incorporated into the POMDP schema. Moreover, the continuous dimension of the observations (fractal values) and the dependency among observations render this problem

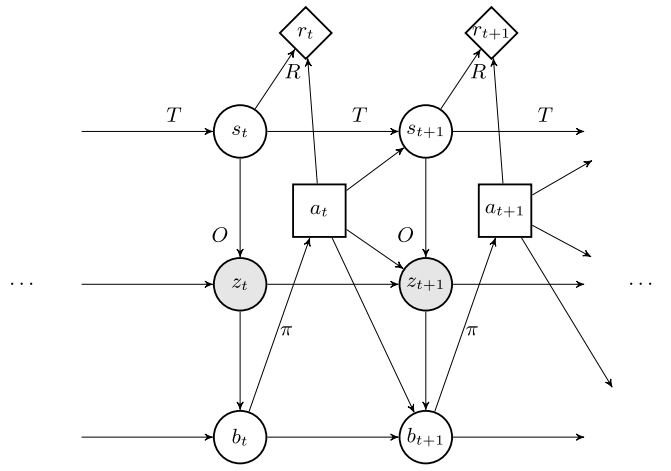


Fig. 5. Probabilistic graphical model of the considered POMDP. The dependency among observations is displayed with an additional arrow among these variables.

non-trivial to solve by means of common POMDP solution algorithms, which commonly assume Z to be discrete.

For the POMDP problem to be fully specified, (i) the transition dynamics T describing the probability $p(s_{t+1}|s_t, a_t)$ and (ii) the observation model O describing the likelihood $p(z_t|s_t, a_{t-1}, z_{t-1})$ must still be defined. We learn both models on the basis of the collected data of the fractal values indicator. The next section presents the employed methods and the inference results.

5. Model inference

The inference of the parameters governing the transition dynamics and the observation generating process serves to simulate and eventually optimize the railway maintenance planning problem. The recovered posterior distributions $p(\theta|D)$ are conditioned on data presented in Section 3, which comprise information on the data-derived fractal values time-series and the recorded maintenance actions over the tracks.

The model used for the inference is a HMM conditioned on actions. The transition model is defined as follows:

$$\begin{aligned} T_0 &\sim \text{Dirichlet}(\alpha_0) \\ s_0 &\sim \text{Categorical}(T_0) \\ T &\sim \text{Dirichlet}(\alpha_T) \\ s_t|s_{t-1}, a_{t-1} &\sim \text{Categorical}(T) \end{aligned} \tag{10}$$

where T_0 represents the initial probability state distributions, while α_0 and α_T are the prior concentration parameters. T_0 is assigned a uniform flat prior, whereas T is given a strongly informative prior to regularize the deterioration or the repairing process. In a transition matrix, the diagonal represents the probability to remain in the same state, while upper-right and lower-left triangles are associated with the probabilities of the system to deteriorate and improve its condition, respectively. As such, the transition matrix related to the action do-nothing, which describes the deterioration process of the system, is regularized with higher prior probabilities on the diagonal, lower on the upper-right triangle and near-zero on the lower-left triangle. In contrast, the transition matrices associated with maintenance actions present higher prior probabilities on the left triangle and near-zero on the right triangle – in order to inform the model that improvements of the system should follow a repair action – but no assumption on the magnitude of improvement.

The observation generating process differs on the basis of the assumed previous action, which can be either a_0 (deterioration process) or one of two possible maintenance actions a_1, a_2. The deterioration process is reflected in the observation model as a *Truncated Student's t* process as follows:

$$z_t - z_{t-1} \sim \text{TruncatedStudentT}(\mu_{d|s_t}, \sigma_{d|s_t}, \nu_{d|s_t}, \text{ub} = -z_{t-1}) \tag{11}$$

The Student's t distribution assigns higher probabilities to tail events than, e.g., the Normal distribution. With a Gaussian likelihood, outliers would induce large shifts in the learned model, in an attempt to render tail events more likely. The Student's t distribution is thus here adopted to enhance robustness of the HMM inference to outliers, which are expected in real-world measurements. Nevertheless, the inference is still free to estimate a high value of degrees of freedom $\nu_{d|s_t}$ if the "fat tail" hypothesis is not correct. The difference among subsequent observations depends on the parameters $\mu_{d|s_t}$, $\sigma_{d|s_t}$, and $\nu_{d|s_t}$ which are state-dependent. An inferred negative value of $\mu_{d|s_t}$ will reflect the negative trend observed in the actual data, but the deterioration process is not forced to monotonically decrease, such that measurement errors are permissible. This implies that an observation can assume a value that is higher to a previous one even when no maintenance actions are taken. The distributions are truncated in $-z_{t-1}$, imposing the negative property of fractal values. The process in Eq. (11) can be seen as a random walk with Truncated Student's t steps or as a particular case of an autoregressive process, where the autoregressive parameter is not learned [42]. In existing literature, the deterioration process is also often modeled as a Gamma process [9]. This alternative approach has been tested herein, but led to common inference issues, such as divergence and non-identifiability. Consequently, we adopted a truncated Student's t process, which yielded improved inference results.

The repair process is correspondingly modeled as an autoregressive process with a truncated Student's t likelihood, so that, once again, only negative values are permissible:

$$z_t \sim \text{TruncatedStudentT}(k_{r|a_{t-1}} * z_{t-1} + \mu_{r|s_t}, \sigma_{r|s_t}, \nu_{r|s_t}, \text{ub} = 0) \tag{12}$$

Specifically, the average improvement in fractal values of the repair process is controlled by an autoregressive action-dependent parameter $k_{r|a_t}$ and a state-dependent parameter $\mu_{r|s_t}$, with standard deviation $\sigma_{r|s_t}$. It is worth clarifying that if the repair process presents no autoregressive property, the model inference will simply assign values close to 0 to the parameter $k_{r|a_t}$.

Since we cannot know whether the first observation stems from a deterioration or a repair process, similarly to the inference of the first hidden state, we model it separately as follows:

$$z_0 \sim \text{TruncatedStudentT}(\mu_{s_{t_0}}, \sigma_{s_{t_0}}, \nu_{s_{t_0}}, \text{ub} = 0) \tag{13}$$

Finally, the aforementioned parameters that influence the observation generating process are defined as follows:

$$\begin{aligned} \mu_{d|s_t} &\sim \text{Normal}(\bar{\mu}_{\mu_{d|s_t}}, \bar{\sigma}_{\mu_{d|s_t}}) \\ \sigma_{d|s_t} &\sim \text{TruncatedNormal}(\bar{\mu}_{\sigma_{d|s_t}}, \bar{\sigma}_{\sigma_{d|s_t}}, \text{lb} = 0) \\ \nu_{d|s_t} &\sim \text{Gamma}(\bar{\alpha}_{\nu_{d|s_t}}, \bar{\beta}_{\nu_{d|s_t}}) \\ \mu_{r|s_t} &\sim \text{TruncatedNormal}(\bar{\mu}_{\mu_{r|s_t}}, \bar{\sigma}_{\mu_{r|s_t}}, \text{ub} = 0) \\ \sigma_{r|s_t} &\sim \text{TruncatedNormal}(\bar{\mu}_{\sigma_{r|s_t}}, \bar{\sigma}_{\sigma_{r|s_t}}, \text{lb} = 0) \\ \nu_{r|s_t} &\sim \text{Gamma}(\bar{\alpha}_{\nu_{r|s_t}}, \bar{\beta}_{\nu_{r|s_t}}) \\ \mu_{s_{t_0}} &\sim \text{TruncatedNormal}(\bar{\mu}_{\mu_{s_{t_0}}}, \bar{\sigma}_{\mu_{s_{t_0}}}, \text{ub} = 0) \\ \sigma_{s_{t_0}} &\sim \text{TruncatedNormal}(\bar{\mu}_{\sigma_{s_{t_0}}}, \bar{\sigma}_{\sigma_{s_{t_0}}}, \text{lb} = 0) \\ \nu_{s_{t_0}} &\sim \text{Gamma}(\bar{\alpha}_{\nu_{s_{t_0}}}, \bar{\beta}_{\nu_{s_{t_0}}}) \\ k_{r|a_t} &\sim \text{Beta}(\bar{\alpha}_{k_{r|a_t}}, \bar{\beta}_{k_{r|a_t}}) \end{aligned} \tag{14}$$

The entire HMM graphical model is displayed in Fig. 6, where shaded nodes indicate the observed variables provided from inspection data. Hidden variables are inferred by means of MCMC sampling exploiting a Hamiltonian Monte Carlo algorithm; namely the No-U-Turn sampler (NUTS) [43]. First, a sample of initial probability distribution

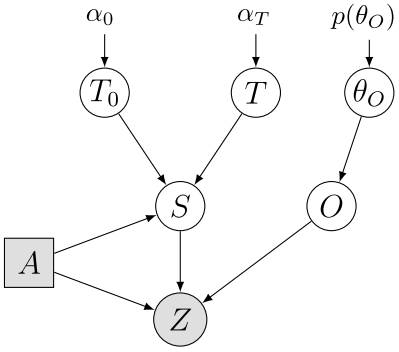


Fig. 6. Graphical model of the inferred HMM. For simplicity, we defined θ_O and $p(\theta_O)$ as the parameters of the observation model in Eq. (14) and their priors, respectively. Arrows indicate dependencies, while shaded nodes indicate observed variables.

and transition matrices are generated. Based on these and the observed trajectory of actions, the complete trajectory of hidden states is jointly sampled. Conditioned on the inferred hidden states, the observation model finally computes the likelihood of the observations for the sampled model, which is used to generate the next sample through the NUTS algorithm. The latter is considered as a robust MCMC sampling algorithm and it is the default method in many packages for Bayesian inference, e.g., PyMC [44].

The model here presented is fed with the aforementioned fractal values and the performed actions. As stated in Section 3, we have access to 10 years of recordings of fractal values and maintenance actions over several tracks. Fractal values are sampled twice per year every 2.5 m, but maintenance actions produce effects over a much broader portion of the track. In addition, fractal values of such a small section are noisy due to the effect of potential measurement errors that generally affect real-world measurements. In order to mitigate these effects, we average fractal values every 150 m. We finally build a dataset of 62 time-series, each one composed by 20 fractal values and 20 maintenance actions (action do-nothing included). As a result, one time-step of the POMDP problem is equal to 6 months. The inference is run with 4 chains and 3,000 samples collected after 4,000 burn-in samples per chain. The recovered posterior distributions present good post-inference diagnostic statistics, with no divergences and high homogeneity between and within chains.

5.1. Inference results

The inferred transition matrix related to the action do-nothing a_0 is reported in Fig. 7. Differently from the transition matrices shown in [32], for example, each entry is not a single parameter but a distribution of plausible values as a consequence of the robust formulation here and the MCMC inference. As seen in Fig. 7, consistent with what is naturally expected in deterioration processes, the highest probability is assigned to remaining in the same state after one time-step (diagonal entries). A deterioration to the subsequent condition level is the second most likely transition, while improvements have near zero probability. Once the structure has reached the worst possible state, i.e., s_3, it stays in this condition with a probability that almost equals 1.

Fig. 8 displays the transition matrix associated with the tamping action a_1, which is a low cost maintenance action with limited effect. If this action is assumed at state s_0, the environment stays in this condition with a probability almost equal to one (high certainty). For deteriorated states, it appears most probable to remain in the same condition or improve by a maximum of one state, although some smaller probabilities are assigned for larger improvements from state s_2 and s_3, which reflects the reduced influence of this action. Deterioration from any given state, upon assumption of such an action, reflects an almost zero probability.

Fig. 9 displays the transition matrix associated with action a_2. Differently from the previous action, transition to the best possible state s_0 is consistently assigned the highest probability, regardless of the starting state. While we provided informative priors to regularize the deterioration or the repairing process, we stress that the MCMC inference learned this higher repairing effect of this maintenance action purely from data. It is worth mentioning that a lower probability of remaining in the same deteriorated state does exist, albeit substantially smaller than for action a_1, reflecting a "failure" of maintenance actions, which was also observed in the training data.

Finally, the observation model parameters are reported in Appendix B in Figs. B.15–B.17. It is worth noting the inferred results for the autoregressive parameter $k_{r|a_i}$. The distribution related to action a_1 comprises significantly higher values than the distribution associated with action a_2, highlighting that the fractal values are allowed to improve more when the latter is applied. While the two parameters were given the same prior, the MCMC inference still learned the substantial different effect of the two maintenance actions. Interestingly, the posterior distributions of the degrees of freedom suggest that the observations are especially "far" from being normally distributed during deterioration.

In order to further validate the goodness of the results, Fig. 10 compares an indicative time-series from real data with one sampled from inferred parameters, where starting values are close and no maintenance action was taken (pure deterioration process). Despite the stochasticity of the observations, the two time-series look extremely similar. Furthermore, it is possible to observe the slow variation of the underlying hidden states, as a result of the inferred transition matrix in Fig. 7, which assigns the highest probability along the diagonal. Conditioning every observation on the previous value in the HMM allowed to correctly model the negative trend of the observations even in absence of changes in the hidden states. As a result, time-series of fractal values simulated from inferred parameters highly resemble the real data. A simpler purely non-autoregressive HMM would not have been able to capture this behavior and would have produced observations that would oscillate around some mean values. The inferred hidden state of the penultimate observation might be questionable and it is probably worth explaining. First, it should be noted that the trajectory of hidden states plotted is computed from the average across MCMC samples and jointly sampled during inference, i.e., each hidden state in the trajectory affects each other's inference. After a high number of deterioration time-steps, the likelihood of the state remaining invariant becomes significantly lower. When this likelihood becomes too low, but the state transition is not reflected in the observation, the inference might still assign the observation to the new state and explain the given value as an outlier/measurement error, which are indeed permissible thanks to the Student's t likelihood. This is exactly what occurred at the penultimate observation, for which the average MCMC samples revealed high uncertainty on whether the hidden state was s_0 or s_1. The inferred state transition is then further strengthened by the clearly visible fractal value jump, revealing the change to condition state s_1 in the following and final observation. Likewise, Fig. 11 shows real and simulated data, where a maintenance action was taken in similar conditions, in order to examine the goodness of the learned repair effect.

While we cannot directly provide the code of the inference presented in this section to protect railway data provided by SBB, we provide a tutorial[1] on the inference run on simulated data that resembles our recordings. The tutorial shows how to recover transition and observation model for MDP and POMDP cases. We hope that such a practice will favor the modeling of MDP and POMDP settings based on data and will further support their utilization in solutions for real-world applications.

[1] Code available on GitHub.

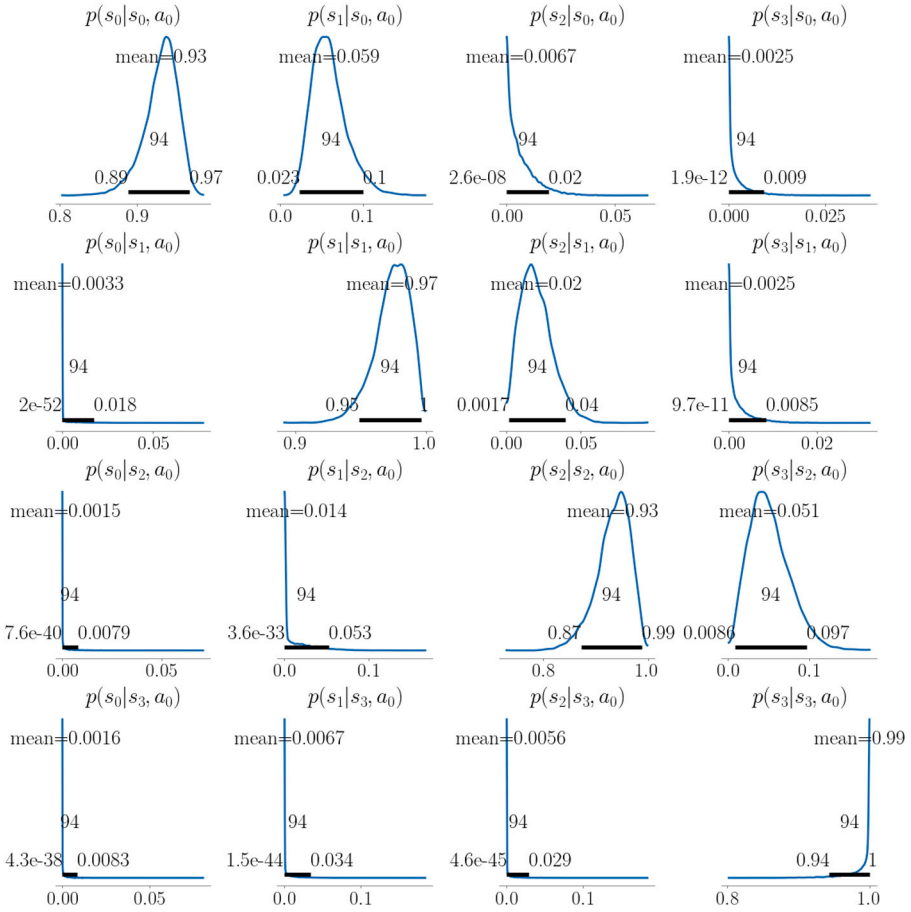


Fig. 7. Transition matrix related to action do-nothing a_0. The distribution at row i and column j is associated with the probability to transition from state i to j when action a_0 is taken. Consistent with what is expected in deterioration processes, the highest estimated probabilities are associated with the state remaining invariant (diagonal entries), lower probabilities exist for deterioration transitions (upper right triangle), and almost zero probability is estimated for improvements of the system (lower left triangle).

It is worth noting that the modeling choices for the POMDP inference, reported in detail in Section 5, are suited to the specific characteristics of the observations and deterioration processes considered in this paper. However, the overall framework presented is a general one admitting different modeling choices, tailored to the data and the characteristics of the problem at hand. In this specific case, the railway condition is indirectly measured via fractal values, i.e., a continuous and negative-valued indicator, justifying the use of a Truncated Student's t process for the likelihood model. While we thus show how to deal with the rather complex case of continuous observations, in the simpler discrete observation case the observation model would be constituted by a probability matrix $|S|\times|Z|$, which can be modeled via a Dirichlet distribution, similarly to the modeling of the transition dynamics in Eq. (10). Furthermore, several extensions to the HMM used are possible. For instance, a Bayesian hierarchical model [45] could be applied to allow dependencies between components of the system [19,24,46,47]. In our case, we may model dependencies between closer tracks that may be affected by similar substructure deterioration. Moreover, one may extend the HMM to time-dependent transition matrices, if there is evidence that the parameters governing the dynamics change over time. Transition matrices would be then enlarged by a further dimension representing time, which would be encoded in the solution. Significant amount of additional data would, however, be required in such a case in order to adequately estimate the model parameters.

6. Solving the POMDP

After having inferred all model parameters, it is now possible to solve the optimization problem, namely to find the optimal policy to be executed given states or observations.

6.1. Full observability

First, full observability of the problem is assumed, i.e., the optimal policy is computed for the case when states are directly and accurately observed. This allows to draw an upper bound of the performance that the POMDP solution can achieve. We consider an infinite horizon problem, with $\gamma = 0.995$, and apply the Q-value iteration algorithm (Eq. (3)) over the entire model distributions, represented by 12,000 samples, that in the MDP case here considered coincides with $p(T|D)$. By implementing the algorithm with JAX [48,49], a JIT compiler for generating high-performance accelerator code, solving the problem for the entire inferred distributions takes only a handful of seconds. As a result, Q-value distributions are computed. Applying Eq. (9), it is possible to take an expectation and compute the optimal action for each state. The resulting actions are thus optimal for the entire range of parameter distributions, i.e., they are robust over epistemic uncertainty. The optimal actions are reported in Table 2. Interestingly, if one applies Q-value iteration algorithm to only the mean values of the inferred transition matrices, i.e., discarding all information contained in the posterior distributions, a different optimal policy is obtained. The policy optimized with the mean parameters estimates is also reported in Table 2.

Moreover, by considering the full transition dynamics distribution, it is possible to compute the percentage of samples for which a specific action is optimal. The results are displayed in Fig. 12. This allows to consider how confident one can be about action optimality, and highlights that it is very likely to obtain a different optimal policy if one optimizes for only a single sample of the transition model distribution. For instance, action a_1 is still optimal in 47% and 43% of samples

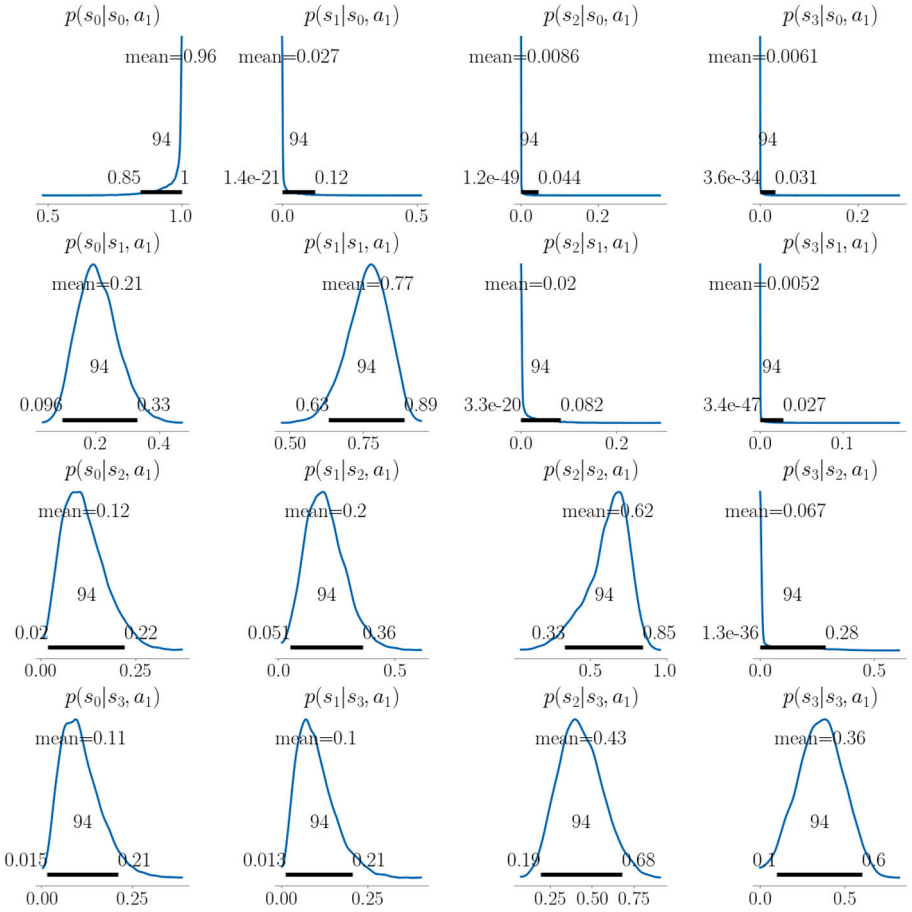


Fig. 8. Transition matrix related to action a_1 (tamping). The distribution at row i and column j is associated with the probability to transition from state i to j when action a_1 is taken. Deterioration of the system (upper right triangle) reflects an almost zero probability, while it appears most probable to remain in the same condition or improve by a maximum of one state, which reflects the reduced influence of this action.

Table 2
Optimal action for every state, optimized for all posterior distributions (top) and only for the mean values (bottom).

State condition	s_0	s_1	s_2	s_3
Robust optimal action	a_0	a_1	a_1	a_2
Optimal action with posterior mean	a_0	a_1	a_2	a_2

Table 3
Expected total life-cycle costs of the robust optimal policy, the policy optimal for the mean values of transition matrices distributions, and a policy which chooses always action a_1 over 240 millions simulations.

	Average	SE	HDI 2.5%	HDI 97.5%
Robust optimal policy	−13,377	0.67	−33,700	−5,000
Optimal action with posterior mean	−13,493	0.60	−31,600	−5,000
Policy always a_1	−16,072	0.94	−46,300	−7,500

when the system is in state s_0 and s_3, respectively. As a result, a policy where a tamping action is taken at every decision step is optimal for a significant number of samples of the inferred distributions. Exploiting the whole distribution parameter space turns out to be crucial for enhancing robustness of the computed policy.

Finally, we simulate the problem 20,000 times with 50 time-steps for all transition model samples, for a total of 240 millions possible trials. The robust optimal policy shown in Table 2 is then applied over all simulations. As a comparison, we also show results for the policy optimized only over the mean values and for the policy that always chooses action a_1. Results are reported in Table 3 in terms of average costs, Standard Error (SE), and 95% Highest Density Interval (HDI). The robust optimal policy delivers the best expected result, although we clarified in Section 2.3 that it does not necessarily have to be the best one for any specific value of the model parameters.

6.1.1. Finite horizon

Concluding the MDP solution study, we compute and showcase the optimal policy considering a finite horizon problem of $H = 50$ time-steps, with terminal value of 0. The Q-value iteration algorithm applied over all inferred distribution parameters now computes $S \times A \times H$ distributions. Similarly to the infinite case, the optimal action at each time-step t is computed as follows:

$$a_t^* = \arg\max_{a \in A} \mathbb{E}_{\theta \sim p(\theta|D)} \left[Q_\theta^{\pi^*}(s, a, t) \right] \tag{15}$$

The resulting policy is reported in Fig. 13. Consistently with the infinite horizon case, solving the Bellman equation for the mean values of the inferred distributions leads to different results, especially for state s_2, further highlighting the importance of incorporating epistemic uncertainty into the solution.

6.2. Partial observability

This section presents now the solution to the POMDP problem. The states are hidden variables and the agent forms a belief over states given the observations received. As pointed out in Section 2.2, planning an optimal policy through beliefs is a far more challenging task and point-based value iteration algorithms offer approximate solutions. However, solving POMDP problems with continuous observations

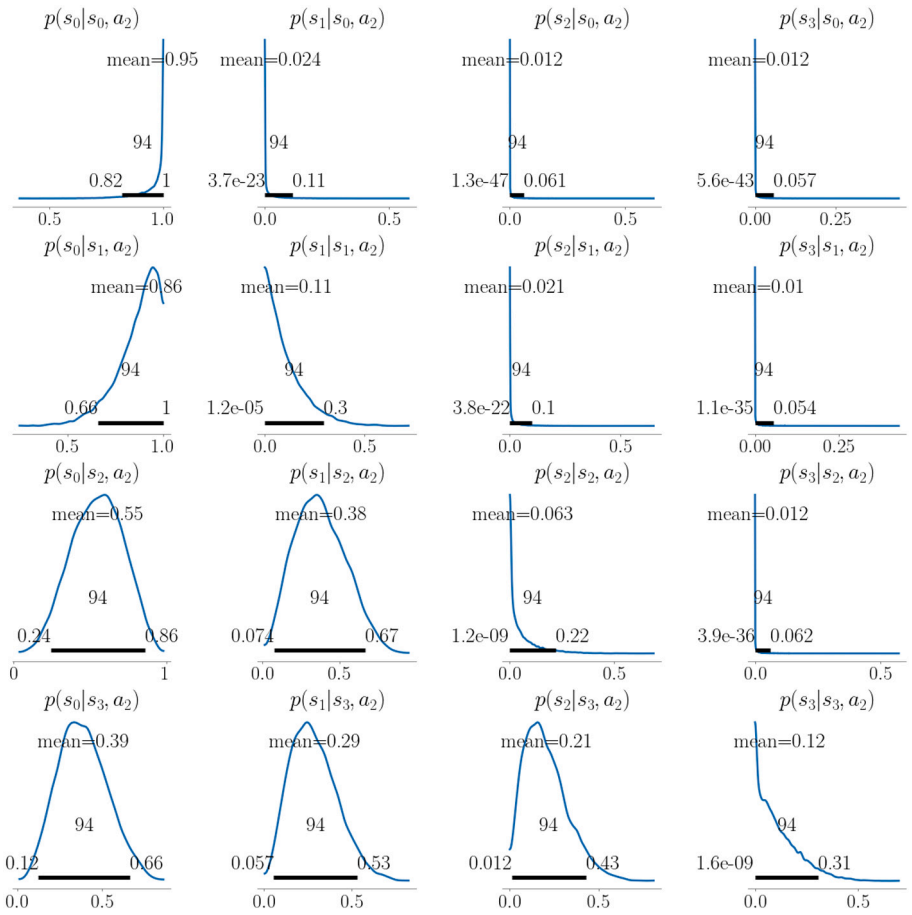


Fig. 9. Transition matrix related to action a_2 (renewal plus tamping). The distribution at row i and column j is associated with the probability to transition from state i to j when action a_2 is taken. Transition to the best possible state s_0 is consistently assigned the highest probability, regardless of the starting state, reflecting the higher repairing effect of this maintenance action.

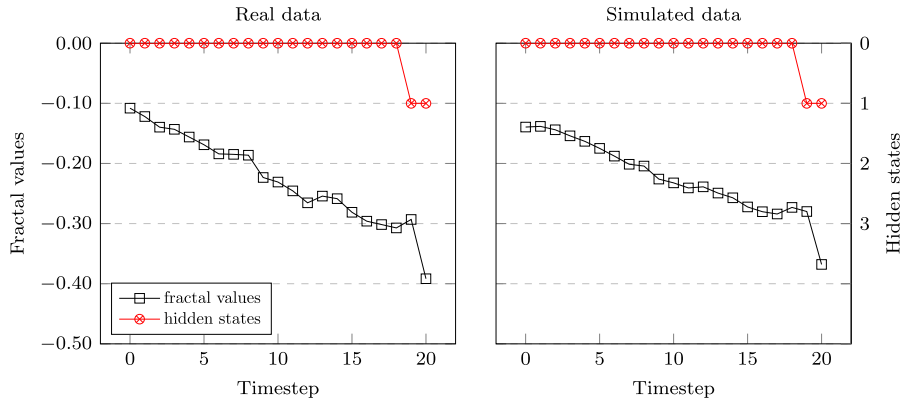


Fig. 10. One indicative time series of fractal values sampled from real data (left) and simulated parameters (right). No maintenance action was taken in the two samples. The associated hidden states are reported in red circles.

remains an even more challenging task, even for these methods, which rely on discretization of the observation space. While some recent advances have been achieved to extend POMDP solvers to continuous observations, e.g. in [50], in this work we rely on a simpler applicable method called Q_{MDP} [51]:

$$\pi_{Q_{MDP}} = \arg\max_{a \in A} \sum_{s \in S} b(s) Q^{\pi^*}(s,a) \tag{16}$$

Namely, the Q_{MDP} method ignores the observation model and computes the Q-values of the underlying MDP given the transition model. It then finds the optimal action at each step by only updating the belief $b(s)$ with Eq. (5). This results in extremely low computational load when compared to point-based methods, at the expense of reduced accuracy, in general problems. The belief update is not affected by the continuous nature of the observation data as it is not computed as a sum over the observations but only as a sum over the hidden states. This implies that Eq. (5) operates for both continuous and discrete observations. The computation of the belief updates is reported in detail in Algorithm 1, clarifying in practice how they are not affected by the continuous nature of the observations.

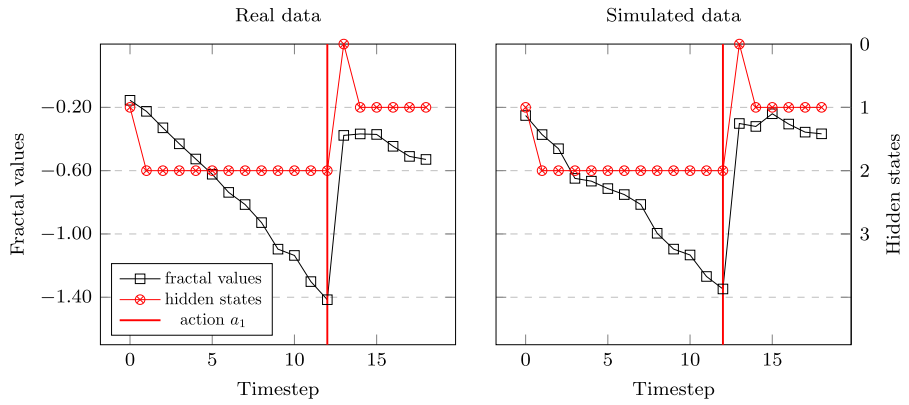


Fig. 11. One indicative time series of fractal values sampled from real data (left) and simulated parameters (right). A maintenance action a_1 was taken at timestep 12 in both cases. The associated hidden states are reported in red circles.

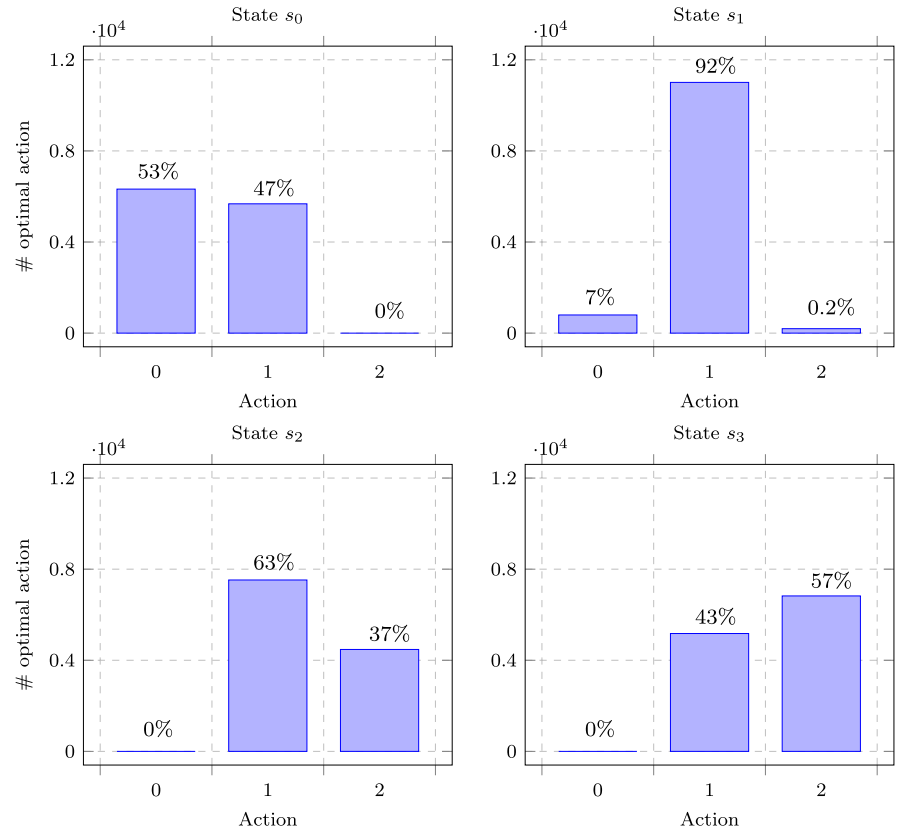


Fig. 12. Number of model samples for which each action is optimal at a given state. It allows to consider how optimal each action is with respect to the model distributions.

Extending this algorithm to all inferred distributions to account for epistemic uncertainty is then straightforward:

$$a^* = \arg\max_{a \in A} \mathbb{E}_{\theta \sim p(\theta|D)} \left[\sum_{s \in S} b_\theta(s) Q_\theta^{\pi^*}(s, a) \right] \tag{17}$$

where both the Q-values and the beliefs depend on θ, which is a sample of the entire POMDP model from transition and observation parameter distributions. All computations among different samples are independent and thus easily parallelizable, without substantially increasing the computational load.

The Q_{MDP} method assumes that the agent's observation uncertainty is removed after one step, in which case the method would provide the optimal solution. Thus, the agent always chooses the action associated with the highest long-term reward, for the current level of uncertainty. Based on this assumption, the main drawback of the method is that it does not choose information gathering actions. In cases where the POMDP problem comprises these actions, the transition dynamics are fast, and/or the observation uncertainty is significant, the method may result in poor performance [10], otherwise it might be remarkably effective in some settings [51]. The reasons why the Q_{MDP} method is especially suited to the studied problem in this work and further insights on the quality of the solution are drawn in Section 6.3.

6.2.1. Numerical results

The entire POMDP simulation algorithm is reported in Algorithm 2. As in Section 6.1, we simulate several trials from the POMDP

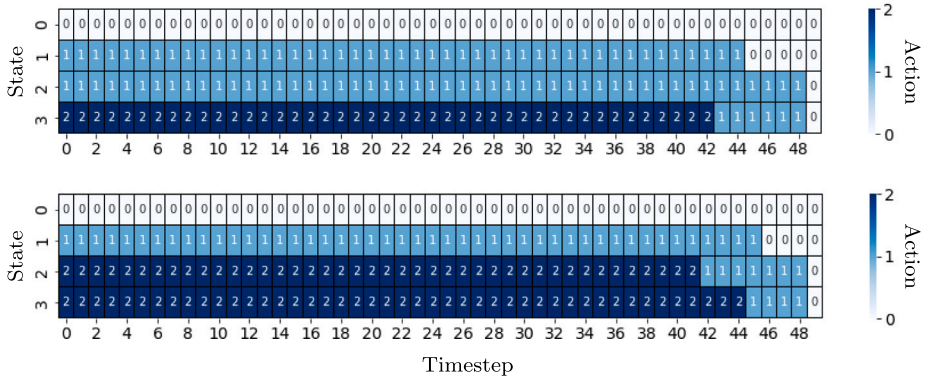


Fig. 13. Optimal policy considering all model distribution (above) and only the mean parameters (bottom) for any given state in time for a finite horizon of 50 time-steps.

Algorithm 1: Computation of the belief updates.

Data: New observation z_{t+1}, previous observation z_t, action a_t, belief $b(s_t)$, transition model T, and observation model O
Result: Updated belief $b(s_{t+1})\ \forall\ s_{t+1} \in S$
TotProb = 0
forall $\bar{s}_{t+1} \in S$ **do**
  Compute new observation likelihood
  $p(z_{t+1}|\bar{s}_{t+1}, a_t, z_t) = O(z_{t+1}, \bar{s}_{t+1}, a_t, z_t)$
  TransitionProb = 0
  forall $s_t \in S$ **do**
    Compute transition probability
    $p(\bar{s}_{t+1}|s_t, a_t) = T(\bar{s}_{t+1}, s_t, a_t)$
    TransitionProb += $p(\bar{s}_{t+1}|s_t, a_t)$
  end
  $b(s_{t+1} = \bar{s}_{t+1}) = p(z_{t+1}|\bar{s}_{t+1}, a_t, z_t) *$ TransitionProb
  TotProb += $b(s_{t+1} = \bar{s}_{t+1})$
end
$b(s_{t+1})$ /= TotProb

parameter distributions to obtain the average performance under nearly all possible scenarios over a finite horizon of 50 time-steps. At the beginning of every simulation, a different POMDP configuration is sampled from the parameter distributions and kept fixed over the 50 time-step horizon. However, the agent does not access the sampled transition and observation model parameters to compute the optimal policy, but it exploits all inferred parameter distributions, approximated through samples, as shown in the computation of a_t in Algorithm 2. The agent thus solves 12,000 POMDP problems in parallel, i.e., it computes distributions of solutions, and selects actions that maximize the expected value with respect to the entire model parameter distribution, according to Eq. (17). The resulting policy is thus robust to parameter uncertainty and it does not need to access the actual POMDP environment parameters. This scheme aims to resemble a real-world scenario, where the agent would never access the real-world true parameters, and it needs to tackle this additional uncertainty. We show that, instead of assuming a particular sample of the POMDP parameter distributions as ground truth, our robust policy represents a natural and safer choice against model uncertainty. We compare the robust policy with the policy based on the means of the posterior parameter distributions, as well as five other different agents that assume knowledge of specific POMDP samples. Specifically, we order our samples based on their (unnormalized) posterior probability and select the ones that correspond to the $[0, 25, 50, 75, 100]$ percentiles. Each of the five agents computes the optimal policy with respect to the sample of the associated percentile. The resulting policies are then evaluated with the scheme previously described. Table 4 summarizes the results in terms of mean performance, SE, and 95% HDI over 100k simulations. The table also reports the results of the policy that always chooses actions a_1, already shown in Table 3. It is worth noting that the standard error of the mean performance of this latter policy is only lower due to the larger number of evaluated simulations, as reported in Section 6.1. The robust policy achieves better mean performance than the other five benchmarking solutions based on specific samples. In particular, among the latter, three benchmarking policies prove not substantially better than the policy that always chooses action a_1. Even though the results from the policies associated with the percentiles 0 and 50 are not very distant from the results of the robust policy, this cannot be known a priori. Likewise, it should not be surprising that the policy based on the posterior mean parameters even shows slightly better empirical results than the robust policy. Indeed, it can be possible to find specific samples, among all possible values, that perform similarly or even slightly better than the robust policy. However, this is strongly dependent on the shape of the parameter distributions and the assumed cost matrix, and one cannot know the performance of such samples until they are actually evaluated. As such, in the context of model uncertainty, the safest choice is represented by the robust policy, which is optimized over all POMDP parameter samples, namely it is "robust" from a model uncertainty perspective.

Despite the partial observability, the robust policy in this case also delivers only slightly worse performance than the full observability case. We note that the agent's belief converges to the actual hidden states within a handful of observations in this problem, after which the actions taken are nearly always optimal. At initiation of the horizon the agent exhibits a conservative behavior by mostly choosing action a_1, which is indeed the most likely to be optimal when the uncertainty over the state is high. As a result, the observed disparity in the MDP versus the POMDP performance is primarily due to early decisions, when the agent's belief is not yet accurate.

In all time-steps, the agent persistently selects actions that are optimal considering all parameter distributions, i.e., robust to epistemic uncertainty. Such a policy is agnostic with respect to real environment transition dynamics and observation generating processes. As such, computed solutions do not overfit a specific POMDP model configuration and are more likely to perform well when deployed to real-world applications, where the environment remains uncertain.

6.3. On the quality of the Q_{MDP} solver

While our focus is not shed on the type of solver to adopt for solution of the POMDP problem, different solution techniques can have an important impact on the accuracy of the POMDP results. We thus further explain here the reasons why the simple Q_{MDP} solver is particularly suited to this specific case. This is tied to the following problem traits: (i) the permanent monitoring nature of the problem, namely the agent does not have to take information-gathering actions, but informative observations are provided at every time-step, (ii) the

Algorithm 2: Evaluation algorithm

Data: Posterior POMDP model parameter distributions $p(\theta|D)$, Reward/cost matrix R
Result: List of total costs through several simulations ListEpisodeRewards
Compute optimal Q-values $Q_\theta^{\pi^*}(s,a,t)$ for all samples θ via Eq. (3)
Set initial belief to all posterior initial probability samples $b_\theta(s_0) \leftarrow p(T_0|D)$
Initialize ListEpisodeRewards $\leftarrow \emptyset$
forall *simulations* $i \in [0 \dots N]$ **do** // N=100,000
  Sample POMDP parameters $\hat{\theta} \sim p(\theta|D)$ from posterior distributions
  Sample initial hidden state $s_0 \sim T_{0_{\hat{\theta}}}$
  Set EpisodeRewards $\leftarrow 0$
  forall *timesteps* $t \in [0 \dots H]$ **do** // H=50
    Plan action $a_t = \text{argmax}_{a \in A} \mathbb{E}_{\theta \sim p(\theta|D)} \left[\sum_{s \in S} b_\theta(s_t) Q_\theta^{\pi^*}(s,a,t)\right]$
    Execute action and sample new hidden state and observation $s_{t+1} \sim T_{\hat{\theta}}(s_t, a_t), z_{t+1} \sim O_{\hat{\theta}}(s_{t+1}, a_t, z_t)$
    Compute reward $r_t = R(s_t, a_t)$ and collect EpisodeRewards += r_t
    Update all belief samples $b_\theta(s_{t+1})$ with Algorithm 1
  end
  ListEpisodeRewards $\leftarrow$ ListEpisodeRewards $\cup$ {(EpisodeRewards)}
end

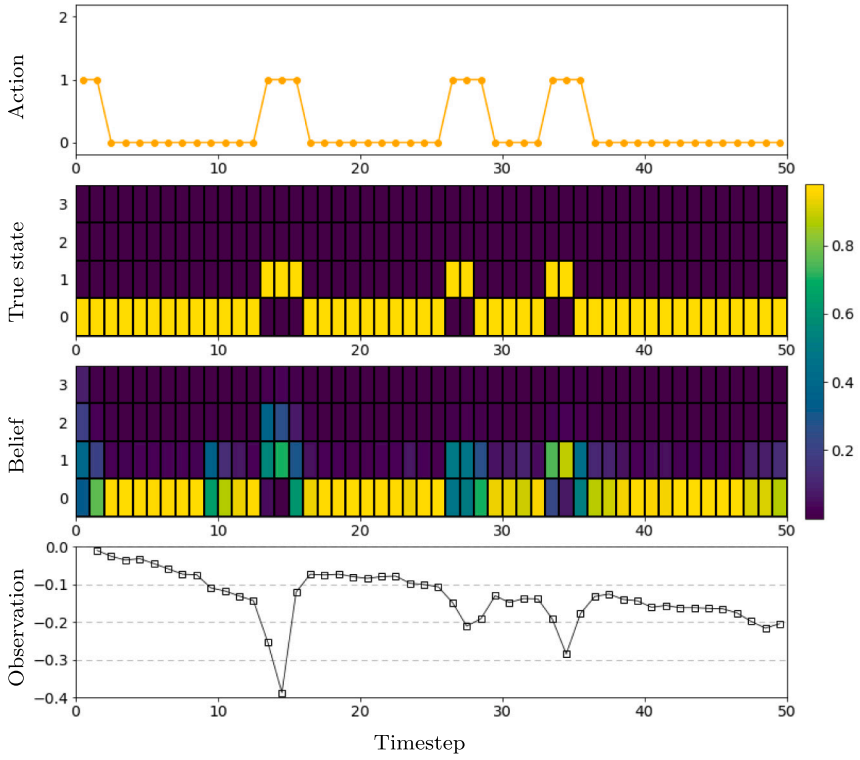


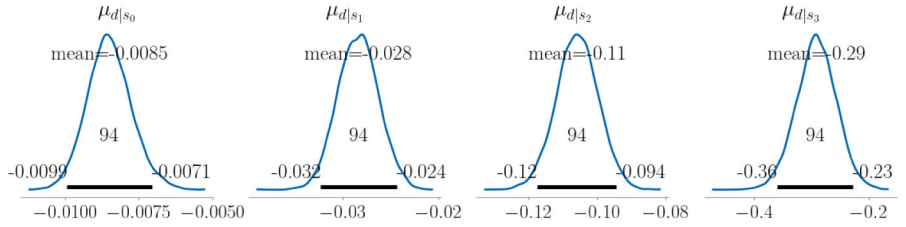
Fig. 14. A sample from the Q_{MDP} planner. From bottom to top: the observations (fractal values) that the agent receives; the agent's beliefs based on the observations; the true hidden states (not accessed by the agent); the planned actions based on the agent's beliefs. After only one observation, the agent's belief already detects the true hidden state and perfectly converges with the second observation. Despite the partial observability, the agent almost always plans the optimal action.

Table 4
Expected total life-cycle costs associated with the Q_{MDP} robust policy, the policy based on posterior mean parameters, and other five benchmarking policies computed from specific samples of the POMDP parameter distributions, evaluated over a finite horizon of 50 time-steps and 100k simulations of the POMDP environment.

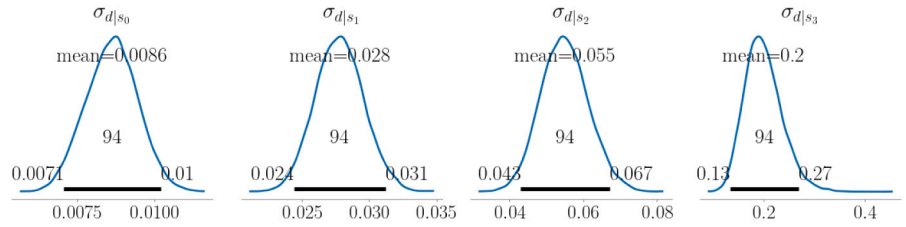
	Average	SE	HDI 2.5%	HDI 97.5%
Robust policy	−14,526	39.47	−39,750	−5,050
Policy with posterior means	−14,478	37.68	−38,250	−5,050
Policy with percentile 0	−14,590	39.66	−40,000	−5,050
Policy with percentile 25	−15,933	45.15	−45,420	−7,400
Policy with percentile 50	−14,912	41.68	−40,350	−5100
Policy with percentile 75	−15,822	48.4	−44,950	−7,200
Policy with percentile 100	−16,497	50.51	−47,010	−7,100
Policy always a_1	−16,072	0.94	−46,300	−7,500

continuous dimension of the observations, that would negatively impact the performance of more sophisticated algorithms, which would rely on discretization of the observation space, while this observations attribute is compatible with and does not affect the performance of the Q_{MDP} solver, and (iii) the high probabilities on the diagonal of the transition matrix associated with action a_0 do-nothing (Fig. 7), which result in slow variation of the underlying hidden states, allowing the Q_{MDP} solver to accurately detect the true hidden states with a few measurements (often only one). In absence of any of those characteristics, the Q_{MDP} solver would not have been equally effective.

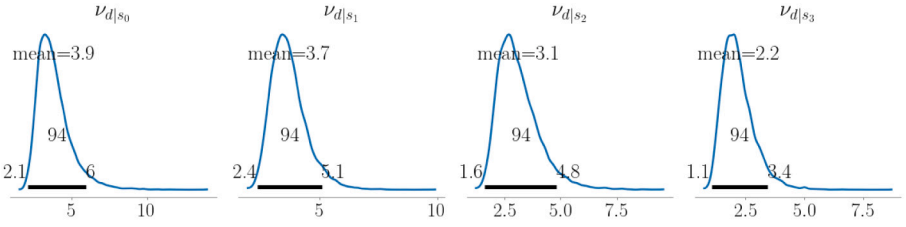
In this section, we illustrate some practical findings to prove the quality of the solver for this POMDP problem, with a finite horizon setting. A key evidence is represented by the comparison of the theoretical



(a) Posterior distributions of state-dependent parameters $\mu_{d|s_t}$.



(b) Posterior distributions of state-dependent parameters $\sigma_{d|s_t}$.



(c) Posterior distributions of state-dependent parameters $\nu_{d|s_t}$.

Fig. B.15. Posterior distributions of observation model parameters (deterioration process).

value for the initial belief $V_{Q_{MDP}}(b_0)$ with the mean simulated reward achieved, where:

$$V_{Q_{MDP}}(b_0) = \max_{a \in A} \sum_{s \in S} b_0(s) Q^{\pi^*}(s, a, t = 0) \tag{18}$$

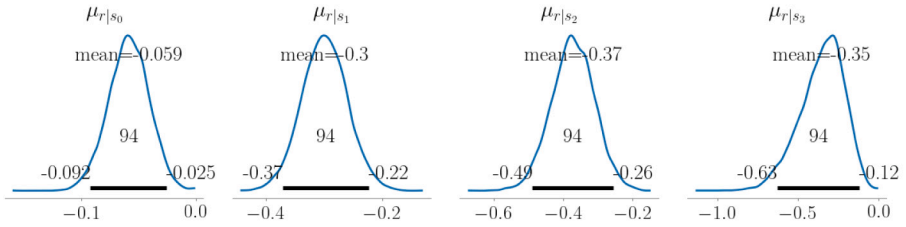
The theoretical value is an optimistic upper bound, while the simulated value represents a lower bound (it is not the optimal policy). If the two are sufficiently close, that can be a good indication that the Q_{MDP} policy is close to the optimal one. In order to only evaluate the quality of the Q_{MDP} planner, we fix the POMDP model parameters to their means. The theoretical value $V_{Q_{MDP}}(b_0)$ expects costs equal to −13,405 (upper bound), while 100k simulations of the Q_{MDP} finite horizon policy over 50 time-steps, and optimized over the means of the model parameters, achieve an average cost of −14,374 (lower bound). Considering the high variability of the costs depending on the realized states (the simulations achieve −5,050 and −123,800 in the best and worst case scenario, respectively), the difference between the two bounds is quite tight.

As a further example, Fig. 14 displays a sample trial of the Q_{MDP} planner. The bottom figure shows the observations, i.e., the fractal values that the agent receives over the trial. Based on the observations, the agent forms beliefs over the states (third subplot), which are compared against the true hidden states, shown in the second subplot. Based on the beliefs, the agent plans the optimal actions, reported in the top subplot. The belief is initialized according to the initial probability state distribution T_0 and is hence not accurate at the beginning. After only one observation, the agent's belief already largely detects the true hidden state and perfectly converges with the second observation. Afterwards, it remains extremely accurate. The agent is able to correctly detect the change to state s_1 at time-step 14 and plan the optimal action a_1 until the state returns to s_0. For other two consecutive times, the agent accurately and timely detects the deterioration to state s_1. In both cases, the state returns to perfect condition s_0 after 2 time-steps. However, the agent is uncertain about the correct state (whether it is s_0 or still s_1) and prefers to precautionary take a further third maintenance action a_1 to be sure of the improvement of the condition. These two decisions represent the only two instances, where after the initial warm up time post-initialization, the agent does not plan the optimal action under actual observation of the hidden state; this deficiency is owed to the uncertainty in the observations.

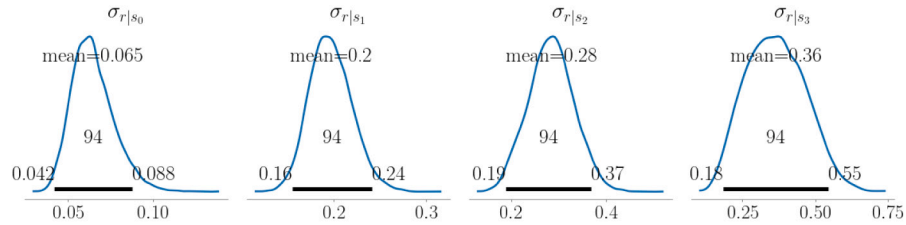
7. Conclusions

In this work, a maintenance planning problem is modeled and solved by means of a POMDP framework. A main contribution of this work is the demonstration of the end-to-end inference of the POMDP model purely from available data. We showcase our method on a real-world maintenance planning problem for railway track infrastructure. We exploit real-world observations (monitoring data) in the form of computed fractal values and actual maintenance actions recorded across Switzerland's railway network. We apply a hidden Markov model conditioned on actions, relying on a truncated Student's t process which describes the deteriorating system, to infer the transition dynamics and the observation generating process of the POMDP problem. Parameter distributions that represent all plausible values under the available data are inferred through MCMC sampling of the model, exploiting the NUTS algorithm. The results present high evidence of convergence, with the simulations highly resembling the real data.

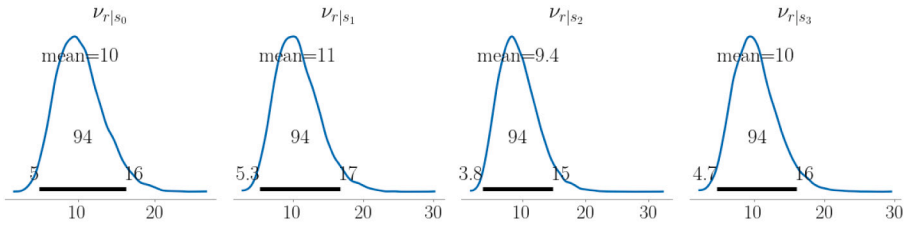
A further contribution of this work lies in application of the inferred model parameter distributions for solving the maintenance planning problem, i.e., computing the optimal sequence of maintenance actions that minimize costs and economic risks over the structure life-cycle. By exploiting all model parameter distributions, the computed policy is not optimal only for specific parameters but accounts for all plausible



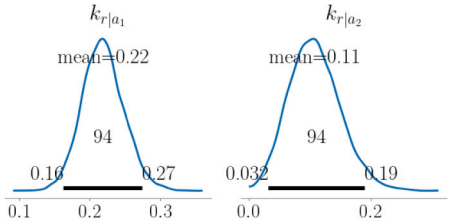
(a) Posterior distributions of state-dependent parameters $\mu_{r|s_t}$



(b) Posterior distributions of state-dependent parameters $\sigma_{r|s_t}$.



(c) Posterior distributions of state-dependent parameters $\nu_{r|s_t}$.



(d) Posterior distributions of the autoregressive parameters $k_{r|a_t}$ for a_1 (left) and a_2 (right).

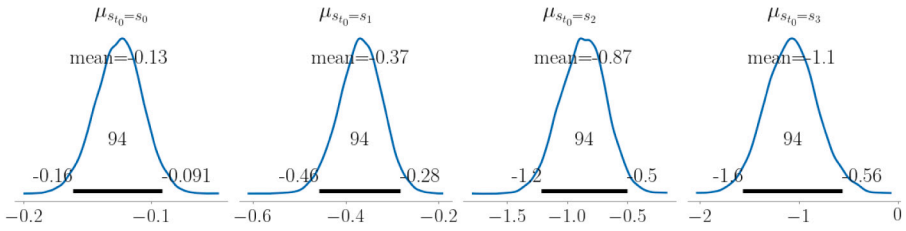
Fig. B.16. Posterior distributions of observation model parameters (repair process).

values that the POMDP environment may assume. The resulting solution is thus robust to epistemic uncertainty over the model parameters. Only a few prior works have managed to combine Bayesian decision making and Dynamic Programming to obtain POMDP solutions that are robust to model uncertainty. In addition to the novel character of the formulation presented in this work, to the best of our knowledge this is also the first time that the two fields are involved with real-world application data.

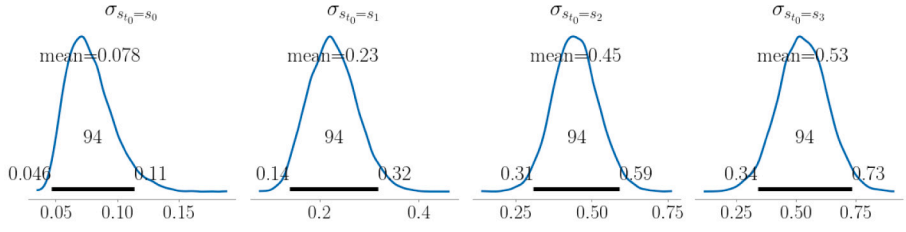
To help the reader in understanding of the current implementation and to facilitate further applications of the POMDP robust planning, we also provide a tutorial that demonstrates the step-by-step solution procedure on a simple example, available on GitHub at https://github.com/giarcieri/Tutorial-on-POMDP-inference-and-robust-planning.

Although our presented framework of POMDP inference and robust solution is showcased on the specific problem of maintenance planning for railway assets, its applicability is general. It should be stressed that the framework is not dependent on the specific modeling configuration choices, as for instance the likelihood model or the selection of the Q_{MDP} method as the particular solution algorithm, which are here tailored to the available observations and problem settings. These configurations can be varied in order to allow the POMDP inference model to be adapted to different applications and to the data at hand. The inferred parameter distributions can then be used to derive robust solutions by employing different solution methods merged with Bayesian decision making principles.

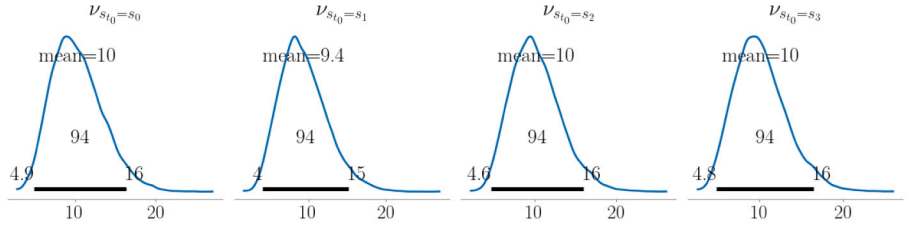
Therefore, this work opens up paths on both new applications and the development of methods for decision making under uncertainty. Possible extensions pertain to the hidden Markov model characteristics, used to infer the parameters of the POMDP environment, with time-dependent transition dynamics or hierarchical system dependencies comprising two possible further paths to explore. In the future, we further wish to investigate the use of Reinforcement Learning (RL) techniques for the development of solutions for maintenance planning that are robust to epistemic uncertainty, without any required prior knowledge of the problem. Several options can be explored along this path, such as the use of model-free [52] or model-based [53] algorithms, while multi-agent RL techniques can be merged with a hierarchical inferred model [46] and possible maintenance constraints



(a) Posterior distributions of parameters $\mu_{s_{t_0}}$.



(b) Posterior distributions of parameters $\sigma_{s_{t_0}}$.



(c) Posterior distributions of parameters $\nu_{s_{t_0}}$.

Fig. B.17. Posterior distributions of observation model parameters (initial observation).

(e.g., budget) can be tackled with solution methods as developed in Andriotis et al. [54].

CRediT authorship contribution statement

Giacomo Arcieri: Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization, Writing – review & editing. **Cyprien Hoelzl:** Data curation, Writing – original draft. **Oliver Schwery:** Resources, Validation. **Daniel Straub:** Supervision, Validation, Methodology, Writing – review & editing. **Konstantinos G. Papakonstantinou:** Supervision, Validation, Methodology, Writing – review & editing. **Eleni Chatzi:** Funding acquisition, Project administration, Supervision, Validation, Methodology, Conceptualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The link to the code is shared in the paper. However, the authors do not have permission to share the data that has been used.

Acknowledgments

The authors acknowledge the support of the Swiss Federal Railways (SBB) as part of the ETH Mobility Initiative project REASSESS.

Algorithm 3: Computation of short, mid and long wave fractal values

Data: Longitudinal level band pass filtered to the range 1 m to 70 m
Result: Fractal values in short, mid and long wave range
Definition of dividers $i \in [5 \dots 580]$;
forall *dividers* i **do**
  Select y, 150 m measurement window from longitudinal level signal;
  Compute the polynomial length L as:
  $L(\lambda = \frac{150}{i}) = \sum_{j=1}^{i} \sqrt{(x_j - x_{j-1})^2 + (y(x_j) - y(x_{j-1}))^2}$ where x and y are the spacial window coordinates subdivided into i segments;
  Divide the Richardson plot into three sections (short, mid, long wave range) with delimiters:
    - Delimiter section 1-2: $log(20'000\,\text{mm}/4) \simeq 3.7$
    - Delimiter section 2-3: $log(3000\,\text{mm}/4) \simeq 2.9$

  foreach *section* $s_i \in i = 1, 2, 3$ **do**
    Run a linear regression for s_i on: $log(L(\lambda))\ \forall\ log(\lambda) \in s_i$;
    Return slope (corresponding to fractal value in wavelength bands i);
  end
  Repeat the fractal analysis taking a 150 m signal window with a 1m shift;
end

Appendix A. Computation of fractal values

See Algorithm 3.

Appendix B. Observation model parameters

See Figs. B.15–B.17.

References

[1] Farrar CR, Worden K. Structural health monitoring: a machine learning perspective. John Wiley & Sons; 2012.
[2] Hughes A, Bull LA, Gardner P, Barthorpe RJ, Dervilis N, Worden K. On risk-based active learning for structural health monitoring. Mech Syst Signal Process 2022;167:108569.
[3] Andriotis CP, Papakonstantinou KG, Chatzi EN. Value of structural health information in partially observable stochastic environments. Struct Saf 2021;93:102072.
[4] Kamariotis A, Chatzi EN, Straub D. Value of information from vibration-based structural health monitoring extracted via Bayesian model updating. Mech Syst Signal Process 2022;166:108465.
[5] Giordano PF, Limongelli MP. The value of structural health monitoring in seismic emergency management of bridges. Struct Infrast Eng 2022;18(4):537–53.
[6] Straub D, Chatzi E, Bismut E, Courage W, Döhler M, Faber MH, Köhler J, Lombaert G, Omenzetter P, Pozzi M, et al. Value of information: A roadmap to quantifying the benefit of structural health monitoring. In: ICOSSAR-12th International conference on structural safety & reliability. 2017.
[7] Jochen H, Krzysztof W. SwissTAMP–big data in proactive track asset management. Eur Railw Rev 2016;(6):41–4.
[8] Ellis H, Jiang M, Corotis RB. Inspection, maintenance, and repair with partial observability. J Infrast Syst 1995;1(2):92–9.
[9] Papakonstantinou KG, Shinozuka M. Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part I: Theory. Reliab Eng Syst Saf 2014;130:202–13.
[10] Papakonstantinou KG, Shinozuka M. Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part II: POMDP implementation. Reliab Eng Syst Saf 2014;130:214–24.
[11] Faddoul R, Raphael W, Soubra A-H, Chateauneuf A. Incorporating Bayesian networks in Markov decision processes. J Infrast Syst 2013;19(4):415–24.
[12] Memarzadeh M, Pozzi M, Zico Kolter J. Optimal planning and learning in uncertain environments for the management of wind farms. J Comput Civ Eng 2015;29(5):04014076.
[13] Schöbi R, Chatzi EN. Maintenance planning using continuous-state partially observable Markov decision processes and non-linear action models. Struct Infrast Eng 2016;12(8):977–94.
[14] Kıvanç İ, Özgür-Ünlüakın D, Bilgiç T. Maintenance policy analysis of the regenerative air heater system using factored POMDPs. Reliab Eng Syst Saf 2022;219:108195.
[15] Wari E, Zhu W, Lim G. A discrete partially observable Markov decision process model for the maintenance optimization of oil and gas pipelines. Algorithms 2023;16(1):54.
[16] Papakonstantinou KG, Shinozuka M. Probabilistic model for steel corrosion in reinforced concrete structures of large dimensions considering crack effects. Eng Struct 2013;57:306–26.
[17] Song C, Zhang C, Shafieezadeh A, Xiao R. Value of information analysis in non-stationary stochastic decision environments: A reliability-assisted POMDP approach. Reliab Eng Syst Saf 2022;217:108034.
[18] Guo C, Liang Z. A predictive Markov decision process for optimizing inspection and maintenance strategies of partially observable multi-state systems. Reliab Eng Syst Saf 2022;226:108683.
[19] Memarzadeh M, Pozzi M, Kolter JZ. Hierarchical modeling of systems with similar components: A framework for adaptive monitoring and control. Reliab Eng Syst Saf 2016;153:159–69.
[20] Ross S, Pineau J, Chaib-draa B, Kreitmann P. A Bayesian approach for learning and planning in partially observable Markov decision processes.. J Mach Learn Res 2011;12(5).
[21] Durango PL, Madanat SM. Optimal maintenance and repair policies in infrastructure management under uncertain facility deterioration rates: an adaptive control approach. Transp Res A 2002;36(9):763–78.
[22] Pozzi M, Memarzadeh M, Klima K. Hidden-model processes for adaptive management under uncertain climate change. J Infrast Syst 2017;23(4):04017022.
[23] Morato PG, Papakonstantinou KG, Andriotis CP, Nielsen JS, Rigo P. Optimal inspection and maintenance planning for deteriorating structural components through dynamic Bayesian networks and Markov decision processes. Struct Saf 2022;94:102140.
[24] Luque J, Straub D. Risk-based optimal inspection strategies for structural systems using dynamic Bayesian networks. Struct Saf 2019;76:68–80.
[25] Koller D, Friedman N. Probabilistic graphical models: Principles and techniques. MIT Press; 2009.
[26] Puterman ML. Markov decision processes: Discete stochastic dynamic programming. John Wiley & Sons; 2014.
[27] Bertsekas D. Dynamic programming and optimal control: Volume I. vol. 1, Athena scientific; 2012.
[28] Bellman R. Dynamic programming. Science 1966;153(3731):34–7.
[29] Sutton RS. Learning to predict by the methods of temporal differences. Mach Learn 1988;3(1):9–44.
[30] Sutton RS, Barto AG. Reinforcement learning: An introduction. MIT Press; 2018.
[31] Spaan MT, Vlassis N. Perseus: Randomized point-based value iteration for POMDPs. J Artif Intell Res 2005;24:195–220.
[32] Papakonstantinou KG, Andriotis CP, Shinozuka M. POMDP and MOMDP solutions for structural life-cycle cost minimization under partial and mixed observability. Struct Infrast Eng 2018;14(7):869–82.
[33] Berger JO. Statistical decision theory and bayesian analysis. Springer Science & Business Media; 2013.
[34] Hoelzl C, Dertimanis V, Landgraf M, Ancu L, Zurkirchen M, Chatzi EN. On-board monitoring for smart assessment of railway infrastructure: A systematic review. In: The Rise of Smart Cities: Advanced Structural Sensing and Monitoring Systems, Chapter 9. 2022.
[35] Wang H, Berkers J, van den Hurk N, Layegh NF. Study of loaded versus unloaded measurements in railway track inspection. Measurement 2021;169:108556.
[36] Landgraf M, Hansmann F. Fractal analysis as an innovative approach for evaluating the condition of railway tracks. Proc Inst Mech Eng, F: J Rail Rapid Transit 2019;233.
[37] Hoelzl C, Dertimanis V, Chatzi EN, Winklehner D, Züger S, Oprandi A. Data driven condition assessment of railway infrastructure. In: Bridge maintenance, safety, management, life-cycle sustainability and innovations. CRC Press; 2021, p. 3251–9.
[38] Audley M, Andrews JD. The effects of tamping on railway track geometry degradation. Proc Inst Mech Eng, F: J Rail Rapid Transit 2013;227.
[39] Mor B, Garhwal S, Kumar A. A systematic review of hidden Markov models and their applications. Arch Comput Methods Eng 2021;28(3):1429–48.
[40] Ailliot P, Monbet V. Markov-switching autoregressive models for wind time series. Environ Model Softw 2012;30:92–101.
[41] Juesas P, Ramasso E, Drujont S, Placet V. Autoregressive hidden Markov models with partial knowledge on latent space applied to aero-engines prognostics. 2021, arXiv preprint arXiv:2105.00211.
[42] Knight K. Limit theory for autoregressive-parameter estimates in an infinite-variance random walk. Can J Stat/Revue Can Stat 1989;261–78.
[43] Hoffman MD, Gelman A, et al. The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo.. J Mach Learn Res 2014;15(1):1593–623.
[44] Salvatier J, Wiecki TV, Fonnesbeck C. Probabilistic programming in python using pyMC3. PeerJ Comput Sci 2016;2:e55.
[45] Gelman A, Carlin JB, Stern HS, Rubin DB. Bayesian data analysis. Chapman and Hall/CRC; 1995.
[46] Andriotis CP, Papakonstantinou KG. Managing engineering systems with large state and action spaces through deep reinforcement learning. Reliab Eng Syst Saf 2019;191:106483.
[47] Morato PG, Andriotis CP, Papakonstantinou KG, Rigo P. Inference and dynamic decision-making for deteriorating systems with probabilistic dependencies through Bayesian networks and deep reinforcement learning. Reliab Eng Syst Saf 2023;109144.
[48] Frostig R, Johnson MJ, Leary C. Compiling machine learning programs via high-level tracing. Syst Mach Learn 2018;4(9).
[49] Bradbury J, Frostig R, Hawkins P, Johnson MJ, Leary C, Maclaurin D, Necula G, Paszke A, VanderPlas J, Wanderman-Milne S, Zhang Q. JAX: composable transformations of python+numpy programs. 2018, URL http://github.com/google/jax.
[50] Hoerger M, Kurniawati H. An on-line POMDP solver for continuous observation spaces. In: 2021 IEEE international conference on robotics and automation (ICRA). IEEE; 2021, p. 7643–9.
[51] Littman ML, Cassandra AR, Kaelbling LP. Learning policies for partially observable environments: Scaling up. In: Machine learning proceedings. Elsevier; 1995, p. 362–70.
[52] Zhu P, Li X, Poupart P, Miao G. On improving deep reinforcement learning for POMDPs. 2017, arXiv preprint arXiv:1704.07978.
[53] Arcieri G, Wölfle D, Chatzi E. Which model to trust: Assessing the influence of models on the performance of reinforcement learning algorithms for continuous control tasks. 2021, arXiv preprint arXiv:2110.13079.
[54] Andriotis CP, Papakonstantinou KG. Deep reinforcement learning driven inspection and maintenance planning under incomplete information and constraints. Reliab Eng Syst Saf 2021;212:107551.