Reliability Engineering and System Safety 275 (2026) 112773

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

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

Maintenance optimization for asset networks with unknown degradation parameters

Peter Verleijsdonk [a,c,*], Collin Drent [b,c], Stella Kapodistria [a,c], Willem van Jaarsveld [b,c]

[a] Department of Mathematics and Computer Science, Eindhoven University of Technology, P.O. Box 513, Eindhoven 5600 MB, The Netherlands
[b] Department of Industrial Engineering and Innovation Sciences, Eindhoven University of Technology, P.O. Box 513, Eindhoven 5600 MB, The Netherlands
[c] Eindhoven University of Technology, P.O. Box 513, Eindhoven 5600 MB, The Netherlands

ARTICLE INFO

Keywords:
Condition-based maintenance
Optimization
Bayesian inference
Deep reinforcement learning

ABSTRACT

We consider the key challenge of maintenance optimization for asset networks where degradation parameters are *heterogeneous* and *unknown*, and must be inferred from real-time degradation data. Combining stochastic modeling and Bayesian statistics, we formulate a partially observable Markov decision process that incorporates estimating shock rates and sizes. This formulation allows us to analytically establish that optimal replacement policies are threshold-type in both degradation levels and parameters. Moreover, we propose an open-loop feedback approach that allows policies trained via deep reinforcement learning (DRL) with access to true parameters to remain effective when deployed using real-time Bayesian point estimates. Complementarily, we develop a Bayesian Markov decision process (BMDP) framework in which the agent maintains and updates posterior distributions during deployment, capturing parameter uncertainty evolution and enabling *scalable* DRL policies that adapt as new data become available. We evaluate our approaches on synthetic data and a real-world case involving interventional X-ray filaments. The proposed DRL methods consistently outperform traditional heuristics. Policies trained for the BMDP remain robust when priors are estimated from historical data and perform well in highly heterogeneous networks. Access to true parameters provides only marginal cost improvements, highlighting the ability of the approach to make effective decisions under limited information.

1. Introduction

Optimizing the maintenance of an installed base of capital goods, such as wind turbine parks, medical imaging equipment, and wafer steppers, is critical for ensuring continuous availability and operational efficiency. The unavailability of such assets imposes significant costs, with an estimated annual economic impact of around $50 billion [1,2]. Nearly half of this unavailability results from component failures, which not only cause substantial financial losses but also pose safety hazards and disrupt essential supply chains. Minimizing these failures—particularly within sectors such as energy, healthcare, and semiconductors—is therefore essential to reduce downtime costs and mitigate safety risks.

To achieve maximum availability and minimize unnecessary maintenance costs, condition-based maintenance (CBM) has become a prominent proactive maintenance strategy. In CBM, sensors continuously collect operational data and transmit them in real time. The degradation signal is constructed from these measurements to represent the health of the asset over time, enabling maintenance actions to be triggered only when failure becomes imminent. While CBM may significantly reduce unnecessary interventions and costly downtime, effectively utilizing real-time degradation data to optimally schedule maintenance across an asset network remains a non-trivial challenge.

The challenge of optimizing CBM for real-world asset networks arises from two interrelated issues: *asset dependencies* and *component heterogeneity*. Asset dependencies, such as economic dependencies arising from joint maintenance setup costs, shared spare parts and limited maintenance resources, significantly influence maintenance scheduling across networks of interconnected assets [3]. Despite their practical significance, such dependencies are often ignored in traditional CBM optimization approaches due to the analytical complexity that results from simultaneously tracking the condition of multiple assets.

Component heterogeneity presents an additional challenge, as even nominally identical components can exhibit a significant variability in degradation behavior due to differences in manufacturing, environmental conditions, or operational usage. Traditional CBM approaches typically assume known degradation parameters and do not incorporate

* Corresponding author at: Eindhoven University of Technology, P.O. Box 513, Eindhoven 5600 MB, The Netherlands.
E-mail addresses: p.verleijsdonk@tue.nl (P. Verleijsdonk), c.drent@tue.nl (C. Drent), s.kapodistria@tue.nl (S. Kapodistria), w.l.v.jaarsveld@tue.nl (W. van Jaarsveld).

https://doi.org/10.1016/j.ress.2026.112773
Received 1 July 2025; Received in revised form 16 April 2026; Accepted 17 April 2026
Available online 20 April 2026

real-time parameter learning from observed degradation data [4,5]. Increasingly, scholars address the challenge of real-time learning by integrating Bayesian inference into maintenance optimization to dynamically update beliefs about component degradation parameters. While promising, such Bayesian models have primarily been studied in single-asset contexts [6–10], where they remain analytically tractable.

Recent work in the field of maintenance optimization has made progress either in (i) integrating uncertain prognostics into maintenance planning [11,12] and developing maintenance planning under partial observability or model uncertainty [13,14], or (ii) optimizing economically dependent multi-component CBM under a known degradation model [15]. What remains largely missing is a unified approach in which parameter uncertainty evolves over time and is updated as new degradation observations arrive, while policies still coordinate actions across assets due to shared setup costs—such as the fixed expenses incurred when dispatching maintenance engineers to service equipment.

In this paper, we contribute: (i) a CBM model for economically dependent assets with unknown and heterogeneous degradation parameters; (ii) a dual formulation as both a partially observable Markov decision process (POMDP) and, under conjugacy assumptions, a Bayesian Markov decision process (BMDP) with a compact belief-state representation; (iii) two solution approaches based on deep reinforcement learning (DRL) for the POMDP/BMDP maintenance optimization problem: (a) open-loop deployment of policies trained under full information and (b) direct training on posterior information; (iv) analytical monotonicity results for the full-information Markov decision process (MDP) with respect to both degradation levels and parameters; and (v) numerical and case-study evidence quantifying when belief-aware training is essential and when open-loop deployment degrades under high heterogeneity.

The assets are assumed to degrade independently, with the shared setup cost constituting the only dependence between assets in the network. In addition, costs are incurred for corrective or preventive component replacements. Other factors, such as safety hazards or operational disruptions, are often incorporated indirectly through the corrective-to-preventive cost ratio [4]. Component degradation is modeled using stochastic shock models [16,17]. We assign prior distributions to represent uncertainty about key degradation parameters, specifically, the shock occurrence rate and the distributional parameter for the damage incurred per shock. Conjugate priors facilitate analytically tractable Bayesian updating as degradation data accumulate, following the single-asset framework in [10].

To address the resulting high-dimensional control problem, we develop methodological innovations and insights that enable the training of DRL-based policies to optimize CBM for networks of interdependent assets that degrade according to a shock process with unknown parameters, which are progressively inferred from real-time degradation signals. Specifically, we develop two distinct Bayesian simulation environments for a multi-asset setting, each supporting a different DRL approach. The *first modeling approach* explicitly simulates the true degradation parameters for each component; these are drawn from a population distribution at the time of component replacement. Subsequent degradation then evolves according to a stochastic shock process governed by these sampled parameters. Since the true degradation parameters are unobservable in practice, this setup constitutes a POMDP. To enable deployment of the trained policies, we propose an *open-loop feedback approach*. Concretely, we collapse the posterior distributions into point estimates, which serve as input to the neural network in place of the true degradation parameters. To complement this first approach, we propose a *second modeling approach* that assumes the availability of a conjugate prior and, in that case, is equivalent in *objective* to the first approach—although the two differ in methodology. Instead of simulating the true degradation parameters, this approach explicitly maintains and updates uncertainty by tracking only the posterior distributions of the parameters. These posteriors are continuously updated through real-time degradation observations. This approach constitutes a BMDP, which forms the foundation for the development of DRL-based policies that operate *directly on the posterior distributional information* available to the asset manager. This marks a significant advancement beyond single-asset models by jointly addressing learning and maintenance optimization in heterogeneous asset networks.

As the training algorithm for our DRL-based policies, we employ specific forms of approximate policy iteration (API) [18], in which the derived monotonicity results are incorporated both through action space constraints during training and in the design of the initialization heuristics. In particular, two heuristic policies are adopted as benchmarks: (i) a two-threshold control-limit heuristic that groups maintenance actions and is used to initialize API, and (ii) an integrated Bayes heuristic known to yield near-optimal performance in the single-asset setting.

Through extensive numerical experiments and a practical case study utilizing real-world degradation data from interventional X-ray (IXR) filaments, we demonstrate the effectiveness and practical applicability of our DRL approaches. To better understand how policy performance varies with the availability of degradation information, we extend the concept of *information levels* [19] to our Bayesian setting: Information level $\mathbf{L}_2$ corresponds to full knowledge of the degradation parameters for each component. In information level $\mathbf{L}_1$, this parameter information is unknown, but the distribution of degradation parameters is available. Finally, information level $\mathbf{L}_0$—used only in the case study—represents a setting where even the distributional information must be estimated from historical data. Although the proposed framework is in this paper directly applied to medical equipment—specifically the filament component of the IXR system—it has broad applicability across other domains, including wind farms, wafer steppers, manufacturing processes, and beyond. The integration of Bayesian parameter learning, economic dependence, and DRL-based policy optimization applies to asset networks where degradation parameters are unknown and need to be learned over time.

Our numerical results show that DRL-based policies trained under both the POMDP and BMDP frameworks significantly outperform heuristic benchmarks when degradation parameters must be estimated (information level $\mathbf{L}_1$). While the POMDP-based DRL approach performs well in controlled settings with limited component heterogeneity, the BMDP-based DRL approach generally yields superior performance—particularly in scenarios with substantial heterogeneity in degradation parameters, as observed in the IXR case study. This advantage arises from the BMDP's ability to maintain and update the belief state in real time. By leveraging the full distributional information, policies trained under the BMDP framework enable more informed and effective maintenance decisions. Furthermore, our experiments reveal that having access to the true degradation parameters ($\mathbf{L}_2$) yields only modest cost reductions compared to when these parameters must be inferred ($\mathbf{L}_1$). Finally, our case study establishes the effectiveness of our methods when distributional information on degradation parameters must be estimated from historical data. In summary, our experiments demonstrate that integrating Bayesian inference and DRL yields effective policies for CBM of asset networks, even when degradation parameters are unknown and must be inferred from real-time degradation data.

The remainder of the paper is organized as follows. Section 2 presents an overview of the relevant literature. We provide a detailed model formulation in Section 3. In Section 4, we detail the heuristic solutions and provide a brief overview of the DRL algorithm. We demonstrate the effectiveness of the proposed solutions through a concise simulation study in Section 5. In Section 6, we present the results of the case study on degrading IXR filaments. We offer concluding remarks in Section 7. A visual roadmap of the paper's structure and the relationships between its main components is provided in Fig. 1.

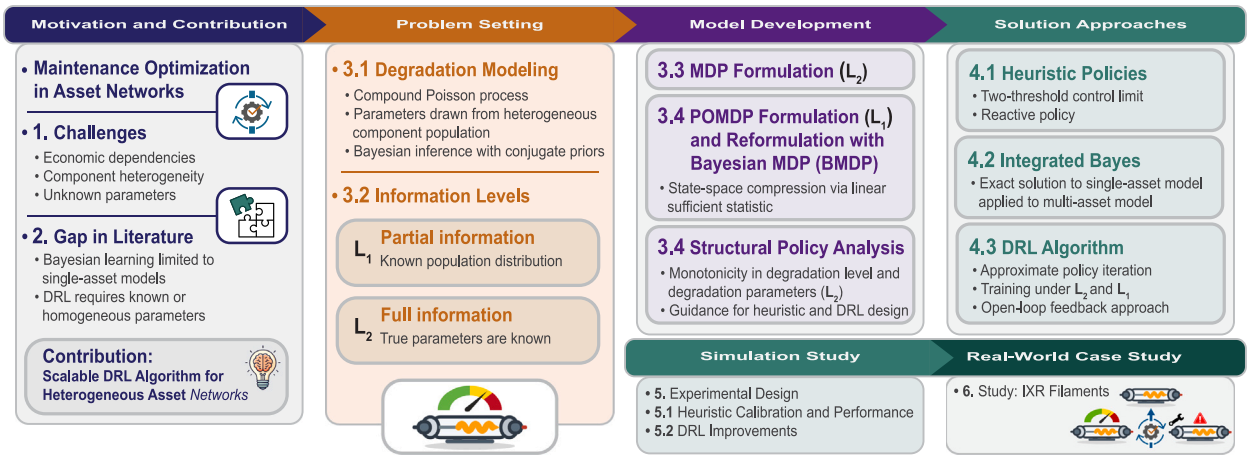


Fig. 1. Flowchart illustrating the conceptual structure of the paper and the relationships between the modeling framework, theoretical results, algorithmic design, and numerical results.

2. Literature review

Our work focuses on the optimization of CBM for asset networks with economic dependencies, the learning of maintenance strategies under component heterogeneity, and the application of DRL to CBM. We next review these three research streams.

Maintenance of asset networks amid economic dependencies

Maintenance models may be classified as single-asset or multi-asset [3,4]. Multi-asset models extend single-asset models by incorporating joint maintenance policies, accounting for various dependencies such as economic, structural, stochastic, or resource dependencies. Economic dependence refers to the cost relationship between joint maintenance of multiple assets and the maintenance of individual assets. If joint maintenance is more expensive, it indicates negative economic dependence; if it is less expensive, it indicates positive economic dependence.

Various models for CBM in systems with economic dependence have been developed in recent literature due to their relevance in industrial settings. Wijnmalen and Hontelez [20] propose a heuristic algorithm that minimizes long-run average costs in a multi-component system by determining simple, component-wise control limits and leveraging coordinated repairs to reduce shared setup costs. Bouvard et al. [21] propose a maintenance optimization method that dynamically groups tasks for multi-component systems using updated degradation data, demonstrated on commercial heavy vehicles. Tian et al. [22] propose a CBM strategy for wind farms that leverages neural networks to predict the degradation level and corresponding remaining useful lifetime (RUL) from real-time data. Tian and Liao [23] develop a CBM policy for multi-component systems using a proportional hazards model and a numerical algorithm to evaluate the cost of the proposed policy.

Zhu et al. [24] introduce a novel CBM policy for multi-component systems experiencing continuous stochastic degradation, utilizing a joint maintenance interval to reduce shared setup costs. Olde Keizer et al. [25] develop a dynamic programming model to determine the optimal CBM strategy for systems with both economic dependencies and redundancy, demonstrating that it significantly outperforms heuristic policies and offers insights into the optimal policy structure. Olde Keizer et al. [26] investigate the trade-off between prompt replacement of failed components and maintenance clustering opportunities in redundant systems with economic dependencies, revealing that heuristic threshold policies may be suboptimal and that misinterpreting load-sharing effects can lead to higher maintenance expenses. Do et al. [27] present a model for a CBM policy in a two-component system with stochastic dependencies, where the degradation of each component is influenced by the other's state, and economic dependencies, where combined maintenance activities are shown to be more cost-effective. Similarly, Oakley et al. [28] propose a CBM policy for multi-component systems that balances the trade-offs between the urgency of replacing failed components due to increased load and the economic benefits of clustering replacements to minimize downtime and maintenance costs. Zhang et al. [15] incorporates imperfect maintenance into CBM optimization for multi-component systems using genetic algorithms. We refer the reader to Zhu et al. [24, Table 1] for a more extensive summary of CBM models proposed for multi-component systems.

The geographical layout of asset networks often introduces a complex dependency. Abdul-Malak and Kharoufeh [29] investigate the challenge of optimally replacing multiple stochastically degrading systems within a shared environment using CBM. Soltani et al. [30] study the problem of optimally maintaining an offshore wind turbine farm under both economic and stochastic dependence due to shared maintenance setup costs and their common environment. They establish monotonicity of the cost function jointly in the degradation level and environmental state, characterize the structure of the optimal replacement policy, and show that sharing maintenance resources is cost-effective. Leppinen et al. [31] introduce directed graphs to represent the economic and structural dependencies of a multi-component system, including scenarios where maintenance on one component may require the disassembly or maintenance of others, and solve the resulting MDP using a modified policy iteration algorithm to determine the most cost-efficient policy.

Learning maintenance strategies under component heterogeneity

Component heterogeneity, characterized by varying degradation rates and failure patterns, necessitates the integration of learning the component-specific characteristics of the degradation process and optimizing maintenance strategies. Degradation processes can be represented using either a discrete or continuous set of degradation states, with Markov chains commonly used to model the degradation process in the discrete case. Information gathered from sensors can enhance the effectiveness of scheduled inspections.

Most existing studies on uncertainty in degradation processes typically assume a specific parametric form, with uncertainty modeled through its parameters and captured via a prior distribution. Bayesian inference is then used to update this uncertainty over time—but so far, these approaches have been limited to single-asset systems only. Van Oosterom et al. [9] develop a POMDP model to incorporate population heterogeneity for maintenance scheduling of a single-asset system that stochastically degrades; however, the population of spare parts consists of multiple indistinguishable types that degrade at varying

rates. Elwany et al. [6] and Si et al. [32] both study a Wiener degradation process with an unknown drift parameter. In both studies, periodic inspections are used to estimate the parameter, with Bayesian inference applied in the former and maximum likelihood estimation in the latter. These estimation methods, combined with observed degradation levels, guide the decision-making process for maintenance interventions. Flage et al. [33] consider a single asset subject to a degradation process with unknown parameters. Sequential inspections assess the degradation level, with the timing of the next inspection based on the current degradation level. Uncertainty in the parameters is modeled using a prior distribution and updated in a Bayesian manner. Drent et al. [10] investigate a single-asset model with one critical component that degrades according to a compound Poisson process with unknown parameters, leveraging real-time degradation data to infer the component's degradation behavior and adjust decision-making accordingly. They demonstrate that, accounting for component heterogeneity, the optimal policy depends on both the asset's age and the observed degradation signal, and that integrating learning with decision-making yields significant cost reductions.

In contrast to Drent et al. [10] and other single-asset models described above, our work considers a network of economically dependent assets, where managers must account for shared setup costs and coordinate maintenance decisions across multiple assets. Extending such single-asset models to this multi-asset setting substantially increases computational complexity, rendering exact solution approaches for the resulting BMDP intractable. We therefore develop scalable DRL-based policies that enable integrated learning and maintenance optimization in heterogeneous asset networks under parameter uncertainty.

Deep reinforcement learning for condition-based maintenance

DRL-based approaches to maintenance optimization have demonstrated promising results across various settings. Kuhnle et al. [34] train opportunistic DRL-based maintenance strategies using proximal policy optimization for parallel assets, achieving reductions in downtime and costs compared to traditional strategies such as reactive and time-based maintenance. Zhang and Si [35] propose a DRL algorithm based on deep Q-network (DQN) to train policies for CBM in multi-component systems with stochastic and economic dependencies, and show in a numerical study that the trained policies outperform benchmarks set by heuristic policies. Mohammadi and He [36] develop a DRL-based method using double DQN (DDQN) for the joint optimization of maintenance and renewal planning under practical constraints. Using historical inspection and maintenance data to simulate a rail infrastructure environment, they show that the approach generates cost-effective policies that enhance network reliability and safety. Hung et al. [37] employ DDQN to train DRL-based maintenance policies in a stochastic factory setting characterized by various degradation levels, uncertain repair times and fluctuating machine workloads.

Lee and Mitici [38] develop an integrated predictive maintenance framework that combines RUL estimation via convolutional neural networks with a DRL-based maintenance policy trained using a soft actor–critic algorithm. The policy triggers maintenance actions based on the estimated RUL distribution and yields significant cost savings and downtime reduction for aircraft turbofan engines. Tseremoglou and Santos [14] propose a two-stage CBM framework for aircraft fleets, where RUL predictions are first used to construct a maintenance policy via a POMDP and then integrated into a rolling-horizon DQN to schedule preventive and corrective tasks subject to resource constraints. Zhuang et al. [12] proposes a prognostics-driven framework that uses Bayesian deep learning to generate uncertainty-aware RUL distributions and dynamically updates maintenance and spare-part ordering decisions for turbofan engines.

Da Costa et al. [19] propose a DRL approach based on distributional DQN to minimize maintenance and downtime costs in asset networks serviced by a single maintenance engineer, where degradation levels are only partially observable. Building upon this work, Verleijsdonk et al. [39] develop DRL-based solutions trained via a form of API—shown to scale better than the aforementioned training algorithms—for industrial asset networks maintained by multiple engineers; however, their approach assumes full observability of all degradation levels.

Recent studies have made important progress (see, e.g., Andriotis and Papakonstantinou [40]) by integrating Bayesian inference with DRL in the context of maintenance optimization. These efforts focus on inferring the degradation state of components rather than the parameters of the underlying degradation process. Thus, while such studies represent key steps toward maintenance optimization under uncertainty, they typically assume that the parameters of the degradation processes are known and can be used to define the simulation environment in which the DRL-based policy is trained.

Our work represents a next step in the integration of learning and maintenance optimization by addressing degradation parameter uncertainty in multi-asset systems. We develop a modeling framework in which the degradation process itself is partially unknown and must be learned in real time from degradation signals. This enables DRL-based policies to adapt not just to the observed condition of assets, but also to the underlying heterogeneity in their degradation behavior. While such models have been considered in single-asset systems, we appear to be the first to consider this learning problem in the realistic context of (economically) interdependent asset networks, where coordination is essential due to, e.g., shared maintenance setup costs. We formalize the resulting optimization problem as both a POMDP and a BMDP, and propose scalable solutions that operate on evolving posterior distributions of degradation parameters. This allows us to make a significant next step in data-adaptive maintenance policies for realistic, industrial-scale asset networks.

3. Model formulation

In this section, we formulate the maintenance problem for asset networks, accounting for economic dependencies and component heterogeneity.

3.1. Compound Poisson degradation

We consider a set of assets (machines) $\mathcal{M} = \{1, \dots, M\}$, each with a single critical component that degrades independently as random shocks arrive. From this point forward, we will refer to the degradation of components as asset degradation. Shocks occur according to a Poisson process, and the damage that accumulates during each shock is a non-negative random variable, meaning that degradation follows a compound Poisson process. This type of shock model is appropriate for, e.g., certain metal and ceramic components in trains, aircraft, and medical equipment (including the IXR systems of our case study) that primarily deteriorate when subjected to discrete stress events [16]. In such settings, degradation accumulates through a sequence of shocks associated with random events, such as thermal cycles, mechanical loads, or electrical stress. Modeling these shock arrivals as a Poisson process is common in the reliability and maintenance literature, as it provides a tractable representation of random and approximately independent stress events occurring over time. Indeed, shock-based degradation models with Poisson arrivals have been widely used in maintenance optimization; see, e.g., [10,41–44]. Consistent with this literature, we adopt the same modeling approach. The Poisson intensity of shock arrivals at machine $m \in \mathcal{M}$ is denoted by $\lambda_m \in \mathbb{R}_+$. The random damage amount at machine m resulting from a shock adheres to the distribution of a member of the one-parameter (denoted as ϕ_m) exponential family. The probability density or mass function of such a random variable can be written as

$$f_m(x \mid \phi_m) = h_m(x) e^{\phi_m T_m(x) - A_m(\phi_m)},$$

where $T_m(x)$ represents the sufficient statistic, and $h_m(x)$ and $A_m(\phi_m)$ are known functions. We assume that shock sizes are non-negative and that the sufficient statistic is linear, i.e., $T_m(x) \equiv x$, which enables a state space reduction in our optimization problem. The function $T_m(\cdot)$ contains all the information necessary to compute any estimate of the parameter ϕ_m. In the literature, this group of distributions is commonly known as the linear exponential family or natural exponential family, named for its linear sufficient statistic, and was first introduced by Morris [45]. To illustrate our approach, we consider the geometric distribution (supported on $\mathbb{Z}_+$) as a representative example.

Example 1 (*Geometric Distribution*)**.** The probability mass function of a geometrically distributed shock size (with support $\mathbb{Z}_+$) with parameter $p_m \in (0,1)$ is given by

$$f(x \mid p_m) = (1-p_m)^x p_m = e^{\ln(1-p_m)x - \ln(1/p_m)}.$$

Note that $h_m(x) = 1$, $T_m(x) = x$, $\phi_m = \ln(1-p_m)$ and $A_m(\phi_m) = \ln(1/p_m)$.

Remark. The geometric distribution serves as an illustrative example, but the framework extends to any one-parameter exponential family with linear sufficient statistic $T_m(x) = x$. This class includes several commonly used discrete and continuous distributions for modeling degradation increments. In the discrete case, examples of the linear exponential family include the Poisson distribution, the binomial distribution (with fixed number of trials), and the negative binomial distribution (with fixed number of successes). In the continuous case, the exponential distribution, normal distribution (with fixed variance), inverse Gaussian (with fixed shape parameter), and the gamma distribution (with fixed shape parameter) are all members of the linear exponential family. In Appendix C, we demonstrate for two standard discrete distributions—the binomial and the Poisson—how they can be written in canonical exponential-family form with linear sufficient statistic, thereby fitting within our modeling framework.

We assume continuous, perfect, remote access to the degradation level for each machine through sensory equipment, while interaction with the system is limited to evenly spaced decision epochs (without loss of generality, we rescale time such that the time between two consecutive decision epochs equals one) corresponding to scheduled maintenance opportunities. Two remarks are in order to motivate this assumption. First, regular maintenance intervals are common in practice when maintenance planning and logistics must be arranged well in advance. For example, in geographically dispersed asset networks, maintenance visits are often scheduled at fixed intervals (e.g., semi-annually), as mobilizing crews and spare parts requires substantial preparation. Second, with the increasing deployment of internet-connected assets, embedded sensors, and digital monitoring infrastructures, near-continuous and accurate condition monitoring is becoming standard in many industrial applications. We therefore deliberately study this data-rich setting as a benchmark for integrated learning and decision-making. In the case study presented in Section 6, a time unit is defined to roughly correspond to the minimum time required to dispatch a maintenance team or deliver a spare part.

Let $N_m(t)$ denote the total number of shocks incurred by the mth asset since its installation up to its *operational age* (that is, the time since the last replacement) $t \in \mathbb{R}_+$, i.e., $N_m(t)$ is a Poisson process with rate λ_m. Here, it is important to note that real-time continuous monitoring of the machine, rather than relying solely on periodic inspections, enables an accurate count of the total number of shocks. The mth asset's degradation level is denoted by $X_m(t) \in [0, \xi_m]$ for some failure threshold $\xi_m > 0$. When the asset's degradation level reaches or exceeds ξ_m, it breaks down and requires corrective maintenance at the next decision epoch. Maintenance is instantaneous and restores the asset to an as-good-as-new condition.

We denote the number of shocks that arrive in the time interval $(t-1,t]$ by $K_m\big((t-1,t]\big) = N_m(t) - N_m(t-1)$. Furthermore, let $Y_m^{(i)}$ denote the size of the ith shock at machine m since the last replacement of the asset. The total incurred damage $X_m(t)$ is a compound Poisson process and satisfies

$$X_m(t) = \sum_{i=1}^{N_m(t)} Y_m^{(i)},$$

where $X_m(0) = 0$ and $N_m(0) = 0$ by definition. Furthermore, let $\boldsymbol{Y}_m\big((t-1,t]\big) = (Y_m^{(N_m(t-1)+1)}, \dots, Y_m^{(N_m(t))})$ be the sizes of the shocks that arrive between age $t-1$ and t, and let $Z_m\big((t-1,t]\big) = \sum_{i=N_m(t-1)+1}^{N_m(t)} Y_m^{(i)}$ be the corresponding total incurred damage. Let $k_m(t)$ denote the observed number of shocks of the mth asset, that is, $k_m(t)$ is the realization of $K_m\big((t-1,t]\big)$. Denote with $\boldsymbol{y}_m(t) = (y_m^{(1)}, \dots, y_m^{(k_m(t))})$ the corresponding observed shock sizes. That is, $\boldsymbol{y}_m(t)$ is the realization of $\boldsymbol{Y}_m\big((t-1,t]\big)$. The tuple $\boldsymbol{\theta}_m(t) = (k_m(t), \boldsymbol{y}_m(t))$ is the observed degradation signal of machine m between ages $t-1$ and t. Given the assumption of a linear sufficient statistic, we do not need to keep track of individual shock sizes. Instead, we can collapse the state space by summarizing the signal as $(k_m(t), z_m(t))$, where $z_m(t) = \sum_{i=1}^{k_m(t)} y_m^{(i)}(t)$ represents the accumulated damage between ages $t-1$ and t. That is, $z_m(t)$ is the realization of $Z_m\big((t-1,t]\big)$. This state space collapse is a key simplification that enables more tractable inference.

Recall that each component stems from a distinct heterogeneous population that consists of components with different degradation parameters λ_m and ϕ_m. In practice, these parameters are hidden; only the observed degradation signal $\boldsymbol{\theta}_m(t)$ is available at operational age t. In this case, we will employ a POMDP to model the integrated challenge of learning the degradation parameters while determining the optimal timing for replacement. In the *underlying* MDP, these parameters λ_m and ϕ_m are drawn from known distributions, Λ_m and $\boldsymbol{\Phi}_m$, respectively, and are observed by an oracle that is aware of the true population heterogeneity. The asset manager's knowledge about the degradation model and its parameters gives rise to various information levels, which are detailed in the next section.

3.2. Information levels

Inspired by the approach of Da Costa et al. [19], we formalize the asset manager's knowledge of the degradation model and its underlying parameters using three distinct *levels of information*:

- ($\mathbf{L}_0$) The asset manager has no information about the model's parameters λ_m, ϕ_m, nor any knowledge of their distribution (i.e., the distributions Λ_m and $\boldsymbol{\Phi}_m$ are unknown) for each $m \in \mathcal{M}$.
- ($\mathbf{L}_1$) The asset manager has no information about the model's parameters λ_m, ϕ_m, but has full knowledge of their distribution (i.e., the distributions Λ_m and $\boldsymbol{\Phi}_m$ are known) for each $m \in \mathcal{M}$.
- ($\mathbf{L}_2$) The asset manager has full information about the model's parameters λ_m, ϕ_m for each $m \in \mathcal{M}$.

Given an information level $\mathbf{L} \in \{\mathbf{L}_0, \mathbf{L}_1, \mathbf{L}_2\}$, the objective is to devise a policy $\pi^{\mathbf{L}}$ that minimizes the total expected discounted cost of managing a specific network of assets. The baseline information level, $\mathbf{L}_0$, reflects the typical conditions encountered by the asset manager in practice. To address the heterogeneity of the component population, hyperparameters of the distributions Λ_m and $\boldsymbol{\Phi}_m$ for each $m \in \mathcal{M}$ should be estimated from available historical degradation data. Subsequently, the asset manager formulates and solves an approximate problem under the enhanced information level $\mathbf{L}_1$, and implements the resulting solution in practice. In contrast, information level $\mathbf{L}_2$ provides the most accurate and informed basis for decision-making. Consequently, an optimal policy derived under this level of information achieves the lowest total expected discounted cost. This progression from $\mathbf{L}_0$ to $\mathbf{L}_2$ highlights the trade-off between the costs of acquiring additional information and the benefits of improved CBM planning. However, although relevant, these acquisition costs are not explicitly incorporated in our model and are left for future work.

The forthcoming sections formalize the model under information level $\mathbf{L}_2$ as an MDP and under information level $\mathbf{L}_1$ as a POMDP or a BMDP pending additional assumptions.

3.3. Partially observable Markov decision process formulation

Under information level $\mathbf{L}_2$, a state h of the asset network can be represented by a vector $h = (x_1, \lambda_1, \phi_1, \dots, x_M, \lambda_M, \phi_M)$, with a minor abuse of notation. Here, x_m denotes the degradation level of the mth asset, and λ_m and ϕ_m are its current degradation parameters. At every decision epoch, the asset manager must choose for each asset whether to maintain the asset or to postpone maintenance activities. Maintenance on failed assets is mandatory. Hence, the *state-dependent action set for the mth asset* is

$$\mathcal{U}_m(h) = \begin{cases} \{0,1\} & \text{if } x_m < \xi_m, \\ \{1\} & \text{if } x_m \geq \xi_m. \end{cases}$$

The *state-dependent action set* $\mathcal{U}(h) = \mathcal{U}_1(h) \times \cdots \times \mathcal{U}_M(h)$ is formed by taking the Cartesian product of the M individual state-dependent action sets. For tractability, we assume that maintenance capacity is sufficient to service all assets simultaneously.

The asset manager incurs costs related to either the corrective or preventive replacement of an asset. If the degradation level at a decision epoch is less than ξ_m, then we can either maintain the asset preventively at cost c_m^{PM} or proceed to the next decision epoch without incurring any cost. If the degradation level of machine m at a decision epoch is greater than or equal to the failure threshold ξ_m, then the failed asset is replaced correctively at cost c_m^{CM}. We assume that for all $m \in \mathcal{M}$ it holds that $0 < c_m^{\text{PM}} < c_m^{\text{CM}} < \infty$ to avoid unrealistic cases. This cost structure is commonly adopted in the maintenance literature [4]. Moreover, both types of replacements take negligible time, which is a reasonable assumption because in practice, replacement times are relatively small compared to the time between decision epochs. The assets are economically coupled through a shared setup cost, a widely adopted assumption for modeling economic dependence in multi-component maintenance systems [3]. In practice, such setup costs represent fixed expenses associated with initiating maintenance activities, such as dispatching a maintenance crew, preparing equipment, or temporarily shutting down part of the system. Specifically, if *at least* one asset is replaced during a decision epoch, a one-time setup cost $c^{\text{ST}} \geq 0$ is incurred. Although we focus on this particular economic dependence, the algorithmic developments presented in this paper readily extend to other forms of economic dependencies, such as more complex setup cost functions. After an asset m is replaced, new degradation parameters λ_m and ϕ_m are drawn from their respective distributions, Λ_m and Φ_m. Therefore, the costs incurred when taking action $a = (a_1, \dots, a_M) \in \mathcal{U}(h)$ in state h are:

$$C(h,a) = c^{\text{ST}} \mathbb{1}_{\{\sum_{m\in\mathcal{M}} a_m > 0\}} + \sum_{m\in\mathcal{M}} \left(c_m^{\text{CM}} \mathbb{1}_{\{a_m=1, x_m \geq \xi_m\}} + c_m^{\text{PM}} \mathbb{1}_{\{a_m=1, x_m < \xi_m\}} \right).$$

Under information level $\mathbf{L}_2$, the objective of the *underlying* MDP is as follows: We are interested in a policy, say $\pi^{\mathbf{L}_2}$, which minimizes the total expected discounted cost. A policy is defined as a series of decision rules, i.e., $\pi^{\mathbf{L}_2} = (\pi_1^{\mathbf{L}_2}, \pi_2^{\mathbf{L}_2}, \dots, \pi_t^{\mathbf{L}_2}, \dots)$, where the decision rule $\pi_t^{\mathbf{L}_2}$ at time t represents a probability distribution over the action set $\mathcal{U}(h(t))$ given the state $h(t)$. Let $J(\pi^{\mathbf{L}_2})$ denote the total expected discounted cost, given a discount factor $\gamma \in [0,1)$. The objective is to find an optimal policy $\pi_*^{\mathbf{L}_2}$ that satisfies

$$\pi_*^{\mathbf{L}_2} = \arg\min_{\pi^{\mathbf{L}_2}} J(\pi^{\mathbf{L}_2}) = \arg\min_{\pi^{\mathbf{L}_2}} \lim_{T\to\infty} \mathbb{E}_{\pi^{\mathbf{L}_2}} \left[\sum_{t=0}^{T} \gamma^t C\left(h(t), a(t)\right) \middle| h(0) = h \right], \tag{1}$$

where $(h(t), a(t))$ represents the tuple of the underlying MDP state and the corresponding action selected by the policy $\pi_t^{\mathbf{L}_2}$ at time t, $t \geq 0$, and $C(\cdot)$ indicates the associated costs (maintenance and setup).

Under information level $\mathbf{L}_1$, the objective becomes to find a policy $\pi_*^{\mathbf{L}_1}$ that satisfies

$$\pi_*^{\mathbf{L}_1} = \arg\min_{\pi^{\mathbf{L}_1}} J(\pi^{\mathbf{L}_1}) = \arg\min_{\pi^{\mathbf{L}_1}} \lim_{T\to\infty} \mathbb{E}_{\pi^{\mathbf{L}_1}} \left[\sum_{t=0}^{T} \gamma^t C\left(o(t), a(t)\right) \middle| o(0) = o \right], \tag{2}$$

where $(o(t), a(t))$ denotes the tuple of the POMDP state and the corresponding action selected by the policy $\pi_t^{\mathbf{L}_1}$ at time t. The state $o(t) = (t_1, \Theta_1((t-t_1, t]), \dots, t_M, \Theta_M((t-t_M, t]))$ contains the machine ages and observed degradation signal history for all assets, i.e., $\Theta_m((t-t_m, t]) = \{\theta_m(\tau) \mid \tau \in (t-t_m, t]\}$. A belief distribution over the unobserved parameters $(\lambda_1, \phi_1, \dots, \lambda_M, \phi_M)$ must be computed from the available history.

A Markovian belief state enables a POMDP to be formulated as an MDP, where each belief represents a state. Although computing belief updates is generally computationally intractable, conjugate pairs ensure that posterior belief states remain within the same distributional family as the prior. This consistency streamlines belief updates and enables us to reformulate the POMDP as a BMDP, improving the tractability of solving the objective in Eq. (2).

3.4. Bayesian Markov decision process formulation

BMDPs are used in decision-making when the true state of the environment is hidden. Instead of making decisions based on the actual state, which is hidden, managers base decisions on a belief state. A belief state is a probability distribution over all possible states that represents the knowledge about the actual state. This framework combines MDPs and Bayesian inference by updating belief states with new information retrieved from observations and actions.

The use of conjugate pairs refines this integration by ensuring that the updated belief distributions remain within the same family as the prior belief distributions. For example, the gamma distribution acts as a conjugate prior for the (unknown) rate parameter of a Poisson distribution. Thus, for each asset $m \in \mathcal{M}$, we assume that λ_m is drawn from a gamma distribution, $\Lambda_m \sim \text{Gamma}(\alpha_m, \beta_m)$, where $\alpha_m > 0$ is the shape parameter and $\beta_m > 0$ is the scale parameter. Additionally, for each $m \in \mathcal{M}$, we assume that Φ_m is distributed according to the general prior for a member of the exponential family characterized by hyperparameters $r_m > 0$ and $s_m > 0$. A member of the exponential family has a conjugate prior with a density that can be expressed as

$$f_{\Phi_m}\left(\phi_m \mid r_m(t), s_m(t)\right) = H_m\left(r_m(t), s_m(t)\right) e^{r_m(t)\phi_m - s_m(t) A_m(\phi_m)},$$

where $H_m\left(r_m(t), s_m(t)\right)$ is a normalizing constant [46]. The beta distribution serves as a conjugate prior for the parameter of several distributions in the exponential family, for instance the geometric distribution.

When asset m is installed, the parameters λ_m and ϕ_m of the compound Poisson degradation process are drawn from the *known* distributions Λ_m and Φ_m, and there is no available history. Thus, we interpret the parameters λ_m and ϕ_m as random variables, denoted by $\tilde{\Lambda}_m$ and $\tilde{\Phi}_m$.

Initial belief. At time $t = 0$, the manager has a prior belief about the degradation process parameters of each asset, which is expressed as a probability distribution over all possible combinations. The joint prior belief distribution of asset m satisfies

$$f^{(0)}_{\tilde{\Lambda}_m, \tilde{\Phi}_m}(\lambda_m, \phi_m) := f_{\Lambda_m}(\lambda_m \mid \alpha_m, \beta_m) \cdot f_{\Phi_m}(\phi_m \mid r_m, s_m).$$

Update with observations and actions. At time t, we use the observed degradation signal $\theta_m(t)$ to infer the joint distribution of $\tilde{\Lambda}_m$ and $\tilde{\Phi}_m$. As shown by Drent et al. [10, Proposition 1], this joint distribution can be factorized into two independent distributions of the same form, with parameters updated solely based on the information contained in $\theta_m(t)$ observed in the previous time period.

Proposition 1. *The joint posterior distribution at time t of $\tilde{\Lambda}_m$ and $\tilde{\Phi}_m$ is given by*

$$f^{(t)}_{\tilde{\Lambda}_m, \tilde{\Phi}_m}(\lambda_m, \phi_m) = f^{(t-1)}_{\tilde{\Lambda}_m}(\lambda_m \mid \alpha_m + k_m(t), \beta_m + t_m(t)) \cdot f^{(t-1)}_{\tilde{\Phi}_m}(\phi_m \mid r_m + x_m(t), s_m + k_m(t)). \tag{3}$$

This is an iterative updating process. After each action and observation, the belief distribution for each asset is updated, and these updated beliefs define the prior for the next step. Thus, at time t, the prior belief distribution of λ_m is a gamma distribution with parameters

$$\alpha_m(t) = \alpha_m + k_m(t) \tag{4}$$

and

$$\beta_m(t) = \beta_m + t_m(t). \tag{5}$$

Similarly, the prior belief distribution of ϕ_m at time t is a beta distribution with parameters

$$r_m(t) = r_m + z_m(t) \tag{6}$$

and

$$s_m(t) = s_m + k_m(t). \tag{7}$$

Thus, the state at time t of the associated BMDP can be represented by a vector $\tilde{h}(t) = (x_1(t), k_1(t), t_1(t), \dots, x_M(t), k_M(t), t_M(t))$. The objective of Eq. (2) is equivalent to

$$\pi_*^{\mathbf{L}_1} = \arg\min_{\pi^{\mathbf{L}_1}} J(\pi^{\mathbf{L}_1}) = \arg\min_{\pi^{\mathbf{L}_1}} \lim_{T\to\infty} \mathbb{E}_{\pi^{\mathbf{L}_1}} \left[\sum_{t=0}^{T} \gamma^t C\left(\tilde{h}(t), a(t)\right) \middle| \tilde{h}(0) = \tilde{h} \right], \tag{8}$$

where $(\tilde{h}(t), a(t))$ denotes the tuple of the BMDP state and the corresponding action selected by the policy $\pi_t^{\mathbf{L}_1}$ at time t.

3.5. Structural properties of the underlying Markov decision process

In this section, we establish several structural properties of optimal replacement policies $\pi_*^{\mathbf{L}_2}$ of the *underlying* MDP of the POMDP model presented in Section 3.3. We provide the proof of Theorem 1 and Theorem 2 explicitly for the case $M = 2$ in Appendix A. The argument extends analogously to the case $M > 2$.

Theorem 1 (*Monotonicity in Degradation Levels*)**.** *Let $x = (x_1, \lambda_1, \phi_1, \dots, x_M, \lambda_M, \phi_M)$ denote a state. Define $\mathcal{M}(x) \subseteq \mathcal{M}$ to be the set of machines for which it is optimal to do maintenance in state x. For any $y = (y_1, \lambda_1, \phi_1, \dots, y_M, \lambda_M, \phi_M)$ that represents a state with more severe degradation than x, i.e.,*

1. *$y_m \geq x_m$ for all $m \in \mathcal{M}(x)$, and*
2. *$y_m = x_m$ otherwise.*

Then, it holds that

$$a^*(x) \in \arg\min_{a \in \mathcal{U}(y)} Q(y, a).$$

Here, $a^*(x) \in \mathcal{U}(x)$ and $a^*(y) \in \mathcal{U}(y)$ denote the optimal actions in states x and y, respectively, and $Q(\cdot,\cdot)$ denotes the state–action value function of an optimal policy. In other words, the optimal action $a^*(x)$ for state x is also an optimal action for a state y with more severe degradation.

From Theorem 1, it follows that optimal policies are *state-dependent* threshold policies. Fig. 2 shows an example of an optimal policy for a 2-asset instance retrieved via policy iteration. Note that, due to the presence of shared setup costs, the optimal action for all states within the black boundary is to "maintain all", despite the fact that the optimal PM threshold for an asset considered in isolation is 15. The solid black boundary is inclusive, whereas the dashed boundary is exclusive. This example highlights the complexity involved in determining optimal policies.

A similar monotonicity result holds for the parameters of the compound Poisson process. We require the definition of the usual stochastic order that quantifies the concept of one random variable being "smaller" than another random variable.

Definition 1 (*The Usual Stochastic Order*)**.** Let X and Y be two random variables such that $\mathbb{P}(X > x) \leq \mathbb{P}(Y > x)$ for all $x \in \mathbb{R}$. Then X is said to be smaller than Y in the usual stochastic order, denoted by $X \leq_{\text{st}} Y$ [47, 1.A.1].

Applying the concept of the usual stochastic order, we establish the theorem on monotonicity in degradation parameters.

Theorem 2 (*Monotonicity in Degradation Parameters*)**.** *Let $x = (x_1, \lambda_1, \phi_1, \dots, x_M, \lambda_M, \phi_M)$ denote a state. Let $\mathcal{M}(x) \subseteq \mathcal{M}$ denote the set of machines for which it is optimal to do maintenance in state x. For any $x' = (x_1, \lambda'_1, \phi'_1, \dots, x_M, \lambda'_M, \phi'_M)$ that represents a state with worse degradation parameters than x, i.e.,*

1. *$\lambda'_m \geq \lambda_m$ and $\phi'_m \geq \phi_m$ for all $m \in \mathcal{M}(x)$, and*
2. *$\lambda'_m = \lambda_m$ and $\phi'_m = \phi_m$ otherwise.*

If $\phi'_m \geq \phi_m$ implies that the corresponding shock size distributions satisfy $Y'_m \geq_{st} Y_m$ (where Y'_m and Y_m are the random variables associated with the respective shock size distributions), then it holds that

$$a^*(x) \in \arg\min_{a \in \mathcal{U}(x')} Q(x', a).$$

Theorem 2 essentially states that if a state x' is "worse" than another state x in terms of degradation parameters, then the optimal maintenance action remains the same. Specifically, if the degradation parameters λ'_m in state x' are greater than or equal to λ_m in state x, and the shock size distributions Y'_m are larger than Y_m in the usual stochastic order, then the same maintenance actions should be applied in both states. Note that Theorem 2 applies to a broad range of MDPs with shock size distributions from the one-parameter exponential family, particularly the geometric distribution supported on $\mathbb{Z}_+$.

While solving for an optimal policy explicitly can be computationally infeasible for larger asset networks, the derived structural properties provide valuable insights that can guide policy design. Although trained DRL agents generally do not satisfy structural guarantees such as the monotonicity properties derived here, the results remain valuable in broader DRL contexts, as they provide theoretical guidance for designing effective heuristic policies and for reducing the complexity of the DRL algorithm by shrinking the action space. In DRL, such heuristic strategies play a crucial role: they serve not only as benchmarks for evaluating trained agents but also as initialization strategies that help kickstart learning in challenging environments. Verleijsdonk et al. [39], for example, use carefully designed heuristic policies to initialize the API algorithm and demonstrate that such initialization significantly reduces the number of iterations needed to reach a near-optimal solution.

4. Heuristic solution approaches

Heuristic methods offer practical and efficient strategies for solving complex decision-making problems, particularly when exact solutions are computationally infeasible. In this section, we discuss three heuristic approaches: the two-threshold control limit heuristic, the integrated Bayes heuristic and approximate policy iteration for BMDPs. We will now define each of these approaches.

4.1. Two-threshold control limit heuristic

The two-threshold control limit heuristic is an offline heuristic approach requiring at least information level $\mathbf{L}_1$. This heuristic involves two distinct thresholds: one for initiating preventive maintenance (PM) and another for opportunistic preventive maintenance (OPM). The PM threshold $\tau_m^{\text{PM}} \in (0, \xi_m]$ triggers scheduled maintenance activities when the mth asset's degradation signal is greater than or equal to τ_m^{PM}. Meanwhile, the OPM threshold $\tau_m^{\text{OPM}} \in (0, \tau_m^{\text{PM}}]$ is set at a lower degradation level, prompting maintenance activities when another asset is already

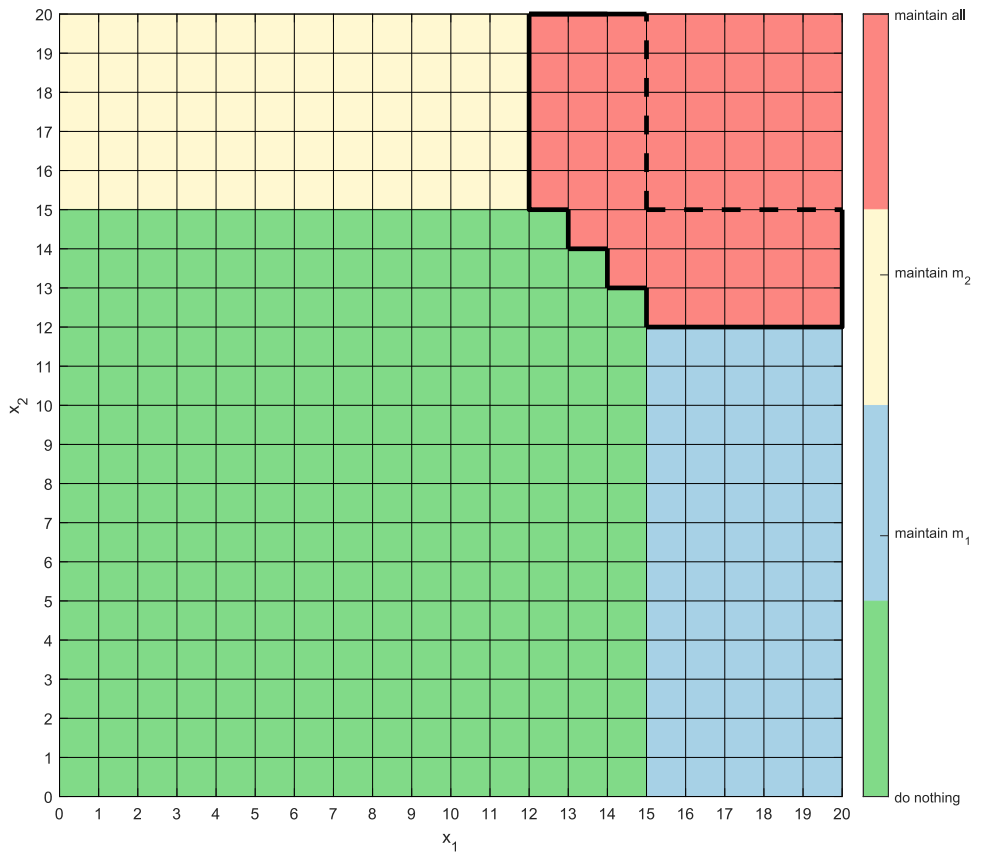


Fig. 2. An optimal policy $\pi_*^{\mathbf{L}_2}$ of the underlying MDP for the 2-asset instance with parameters $(c_m^{\mathrm{PM}}, c_m^{\mathrm{CM}}, c_m^{\mathrm{ST}}, \gamma) = (1, 10, 3, 0.99)$, $\Lambda_m \overset{\mathrm{d}}{=} 5$ and $Y_m^{(i)} \overset{\mathrm{d}}{=} 1$.

undergoing maintenance. This approach capitalizes on the opportunity to address potential issues while sharing maintenance setup costs, thereby reducing the overall costs. We refer to this approach as $\pi_{\mathcal{N}}^{\mathbf{L}_1}$ because it is the most naive heuristic approach; it does not explicitly consider component heterogeneity, yet it is widely used in practice. Note that setting $\tau_m^{\mathrm{OPM}} = \tau_m^{\mathrm{PM}} \geq \xi_m$ for all $m \in \mathcal{M}$ yields a reactive heuristic, which we shall denote by $\pi_{\mathcal{R}}^{\mathbf{L}_1}$.

4.2. Integrated Bayes heuristic

The integrated Bayes heuristic approach is adopted from Drent et al. [10, Section 5.3], and applies a state-space truncation to solve the objective of Eq. (8) in the single asset scenario using value iteration. We extend the policy to the multi-asset scenario by applying the obtained policy $\pi_{\mathcal{I}}^{\mathbf{L}_1}$ to each asset individually, thus ignoring the economic dependence. Due to the curse of dimensionality, solving the objective of Eq. (8) for larger asset networks is computationally intractable.

4.3. Approximate policy iteration for Bayesian Markov decision processes

To learn maintenance policies for economically coupled assets in the context of BMDPs, we employ a variation of approximate policy iteration. Specifically, we utilize deep controlled learning (DCL), which was first introduced by Temizöz et al. [18]. Verleijsdonk et al. [39] have shown that API/DCL can effectively learn maintenance policies (together with dispatching strategies) for an industrial-scale network of homogeneous assets, and that it outperforms several baseline heuristics, including decomposition-based methods and solutions derived from combinatorial optimization algorithms. Moreover, DCL has demonstrated superior performance in inventory problems compared to other DRL algorithms including *proximal policy optimization* and *asynchronous advantage actor-critic* [18].

To apply DCL to BMDPs, we introduce various novel techniques: (i) randomized action selection to avoid indexation bias, (ii) sampling from the prior distributions to directly train neural network policies for BMDPs, (iii) leveraging heuristic information to limit the action space in a significant number of states, and (iv) an open-loop feedback feature vector that enables the application of neural network policies trained in an $\mathbf{L}_2$ setting to be applied in an $\mathbf{L}_1$ setting.

4.3.1. Deep controlled learning for Bayesian Markov decision processes

Following the approach of Verleijsdonk et al. [39], we apply DCL to train a neural network for sequential action selection in asset management, thereby significantly reducing the complexity of the action space. To avoid indexation bias, we randomly select a new permutation $\sigma \in S(\mathcal{M})$, where $S(\mathcal{M})$ denotes the set of all permutations of $\mathcal{M}$, at each decision epoch, ensuring that the order in which actions are chosen for machines is randomized. Following each action selection, the input state is updated with the effects of the action before the algorithm decides on an action for the next asset. More specifically, the state transition $h \to h'$ is decomposed into two stages: The first stage is governed by the deterministic outcomes of the selected actions $a \in \mathcal{U}(h)$, while the second stage is influenced by the random progression of the degradation processes. In more detail, $h \to h'$ is decomposed into $h \overset{a_{\sigma(1)}}{\to} h^{a_{\sigma(1)}} \overset{a_{\sigma(2)}}{\to} \dots \overset{a_{\sigma(M)}}{\to} h^{a_{\sigma(M)}} =: h^a$ and to $h^a \overset{t\to t+1}{\longrightarrow} h'$. The sequence in which the actions $a_1, \dots, a_M$ are handled does not matter, so we will only describe the processing of the action for the mth asset.

In the case that $(a_m = 0)$, maintenance on the asset is postponed and the corresponding state variables remain unchanged. The action $(a_m = 1)$ represents initiating a maintenance action. The corresponding degradation level x_m is set to 0 and new degradation process parameters λ_m and ϕ_m are sampled from their respective sampling distributions. To determine h', we simply update the state variables of each asset based on the random evolution of the degradation processes.

We now provide a concise overview of the application of DCL to BMDPs (on the application of DCL to MDPs, we refer to Verleijsdonk et al. [39] and to Temizöz et al. [18]). Under the information level $\mathbf{L}$, the DCL algorithm comprises the following three steps:

1. Select an appropriate initial solution $\pi_0^{\mathbf{L}}$.
2. Using $\pi_0^{\mathbf{L}}$, construct a data set $\mathcal{D}$ containing state–action mappings.
3. Train a neural network classifier to learn the state–action mappings in $\mathcal{D}$.

In step three, the neural network can be regarded as a parameterized function mapping from $\mathbb{R}^r$ to $\mathbb{R}^s$ for some $r, s \in \mathbb{N}$. We denote this function as $N_\theta(\cdot)$, where θ signifies the function parameters. The input to the neural network is the feature representation $f^{\mathbf{L}}(h) \in \mathbb{R}^r$ of state h, and the output $N_\theta(\cdot) \in \mathbb{R}^s$, where $s = 2$ in our case, is converted into a probability distribution over the action space. The action $\tilde{a}$ with the highest probability, $N_\theta(\cdot)_{\tilde{a}}$, is selected, effectively defining the policy. Given the actions $a_{\sigma(1)}, \dots, a_{\sigma(m-1)}$ for the first $m-1$ assets (according to the permutation σ), the action $a_{\sigma(m)}$ for the asset $\sigma(m)$ in state h is selected using the following decision rule:

$$\pi_\theta^{\sigma(m)}(f^{\mathbf{L}}(h^{a_{\sigma(m-1)}})) = \underset{\tilde{a} \in \mathcal{U}_m(h^{a_{\sigma(m-1)}})}{\arg\max} \; [(N_\theta(f^{\mathbf{L}}(h^{a_{\sigma(m-1)}})))_{\tilde{a}}],$$

where by convention $h^{a_{\sigma(0)}} = h$. We refer to $\pi_\theta^{\mathbf{L}}$ as the neural network policy, trained in a setting with information level $\mathbf{L}$, that selects in each decision epoch, for each asset $\sigma(m)$, the action $\pi_\theta^{\sigma(m)}(f^{\mathbf{L}}(h^{\pi_\theta^{\sigma(m-1)}}))$.

To apply DCL to BMDPs, we construct a data set $\mathcal{D}$ containing state–action mappings for BMDP states $\tilde{h}$. To advance these states, we modify the two-stage decomposition as follows: The first stage, $\tilde{h} \to \tilde{h}^a$, remains largely unchanged, except that after a maintenance action ($a_m = 1$) is selected, the prior distributions $\tilde{\Lambda}_m$ and $\tilde{\Phi}_m$ are reset by setting the state variables $k_m(t)$ and $t_m(t)$ to 0. To determine $\tilde{h}^a \overset{t \to t+1}{\longrightarrow} \tilde{h}'$, we first sample the degradation process parameters from the prior distributions specified in Eq. (3). Using these parameters, we sample the shock arrivals and the resulting increase in degradation, and subsequently update the state variables and belief distributions through the prior-to-posterior update. The posterior distributions are then used as the prior distributions for the next time step. This enables us to directly train policies using DCL in the BMDP setting.

In the next section, we discuss suitable initial policies and feature representations for state information.

4.3.2. Initial policies and feature representations

Verleijsdonk et al. [39, Section 5.3] argue that initiating DCL with an appropriate policy $\pi_0^{\mathbf{L}}$ considerably reduces computation times. The optimized two-threshold control limit heuristic satisfies all relevant listed properties since it explores sufficiently many states and has low computational complexity (as opposed to the integrated Bayes heuristic from Section 4.2). Moreover, the optimized PM and OPM thresholds τ_m^{PM} and τ_m^{OPM} can be leveraged to effectively restrict the individual, state-dependent action sets $\mathcal{U}_m(h)$ for a significant number of states. This is particularly useful when the failure threshold ξ_m is relatively large. Specifically, we train and evaluate the policy improvements of the two-threshold control limit on a restricted action space defined as $\tilde{\mathcal{U}}(h) = \tilde{\mathcal{U}}_1(h) \times \cdots \times \tilde{\mathcal{U}}_M(h)$, where for each $m \in \mathcal{M}$, the individual action set $\tilde{\mathcal{U}}_m(h)$ is given by:

$$\tilde{\mathcal{U}}_m(h) = \begin{cases} \{0\} & \text{if } x_m \le \delta_m \cdot \tau_m^{\mathrm{OPM}}, \\ \{1\} & \text{if } x_m \ge \zeta_m \cdot \tau_m^{\mathrm{PM}}, \\ \{0,1\} & \text{otherwise.} \end{cases}$$

Here, for each $m \in \mathcal{M}$, $\delta_m \in [0,1]$ and $\zeta_m \ge 1$ are chosen conservatively to ensure that the action space is restricted only in clearly suboptimal regions of the state space. In all numerical experiments, we set $\delta_m \equiv 0.5$ and $\zeta_m \equiv 1.5$, so that PM is prohibited only when x_m is well below the OPM threshold, and doing nothing is prohibited only when x_m is substantially above the PM threshold.

The feature representation of a state depends on the information available to the manager. We propose a feature design that communicates the state information for each asset in a manner that is *customized to the specific asset for which we are currently determining an action*. Verleijsdonk et al. [39, Section 7.1] demonstrate that using such a feature design results in less training variability and consistently produces policies that perform significantly better.

In case of the underlying MDP under information level $\mathbf{L}_2$, the most compact representation of the state h is given by

$$f_1^{\mathbf{L}_2}(h) = \left(x_1, \lambda_1, p_1, \iota_1, \dots, x_M, \lambda_M, p_M, \iota_M, \eta\right),$$

that is, the feature vector $f_1^{\mathbf{L}_2}(h)$ includes an information block $(x_m, \lambda_m, p_m, \iota_m)$ for each $m \in \mathcal{M}$ that can be derived from h, along with one additional feature. Here, x_m is the observed degradation level of the mth asset, and the entries λ_m and p_m denote its degradation parameters (cf. Example 1). The last block entry ι_m indicates whether we are currently selecting an action for asset m. Lastly, the additional feature η indicates whether there is an opportunity for OPM. More specifically, η equals 1 if maintenance on at least one asset is mandatory or if the maintenance action has already been selected in the current decision epoch, and 0 otherwise. Note that the dimension of the feature vector is $r = 4M + 1$.

In the case that we are agnostic about the degradation parameters, these features need to be estimated from the available history. This can be done via maximum likelihood estimation or by collapsing the belief distribution into a point estimate. Note that the maximum likelihood estimators can only be computed when some data has been collected. Bayesian inference does not suffer from this drawback since the knowledge of the true hyperparameters already yields a good estimate. The belief distribution can be collapsed in a suitable feature representation using an open-loop feedback approach. Specifically, we collapse the belief distributions into the following point estimates, which represent the means of the distributions:

$$(\lambda_m^{\mathrm{Bayes}}, p_m^{\mathrm{Bayes}}) = \left(\frac{\alpha_m(t)}{\beta_m(t)}, \frac{r_m(t)}{r_m(t) + s_m(t)}\right), \tag{9}$$

where $\alpha_m(t), \beta_m(t), r_m(t)$ and $s_m(t)$ are defined in Eqs. (4)–(7).

To apply an $\mathbf{L}_2$ trained policy in the $\mathbf{L}_1$ setting, we modify the feature representation $f_1^{\mathbf{L}_2}(h)$ by substituting the estimates from Eq. (9), i.e.,

$$f_2^{\mathbf{L}_1}(\tilde{h}) = \left(x_1(t), \lambda_1^{\mathrm{Bayes}}, p_1^{\mathrm{Bayes}}, \iota_1, \dots, x_M(t), \lambda_M^{\mathrm{Bayes}}, p_M^{\mathrm{Bayes}}, \iota_M, \eta\right).$$

We denote the resulting policy as $\pi_\theta^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h}))$ to emphasize that the policy is trained in the $\mathbf{L}_2$ setting but applied in the $\mathbf{L}_1$ setting using the open-loop feedback approach.

Lastly, we extend this feature representation for BMDPs by including the relevant history captured by $k_m(t)$ (the number of shocks) and $t_m(t)$ (the machine age). Thus,

$$f_3^{\mathbf{L}_1}(\tilde{h}) = \Big(x_1(t), \lambda_1^{\mathrm{Bayes}}, p_1^{\mathrm{Bayes}}, k_1(t), t_1(t), \iota_1, \dots, x_M(t), \lambda_M^{\mathrm{Bayes}}, p_M^{\mathrm{Bayes}}, k_M(t), t_M(t), \iota_M, \eta\Big).$$

It is important to note that these feature designs can be effective even outside the Bayesian framework assumed so far and are not constrained by the modeling assumptions inherent to that framework. While their accuracy is generally highest when the data exhibits behavior similar to the underlying model structure, this similarity is not a strict requirement for their practical usefulness.

5. Simulation study

This section presents the findings of a compact simulation study where we optimize the decision process for two example instances. Unlike the case study that will be presented in Section 6, this simulation

Table 1
Hyperparameter settings and cost structures (adopted from Drent et al. [10, Section 5]) considered in the simulation study.

Instance	M	ξ_m	μ_Λ	CV_Λ	μ_Φ	CV_Φ	α	$1/\beta$	r	s	c_m^{PM}	c_m^{CM}	c^{ST}	γ
I.1	2	20	1	0.3	0.5	0.01	11.11	0.09	4999.5	4999.5	1	5	1	0.99
I.2	2	20	1	0.6	0.5	0.02	2.78	0.36	1249.5	1249.5	1	10	1	0.99

study starts from the premise that the manager has full distributional information about the underlying MDP model, i.e., information level $\mathbf{L}_1$ or above. This allows for a strictly controlled setting.

The aim of this simulation study is twofold:

1. To assess the benefits of integrating learning and decision-making, which explicitly takes into account the heterogeneity in asset degradation (value of integration).
2. To examine the value of acquiring additional information about the system to improve decision-making, particularly in understanding the uncertainty and variability of asset degradation (value of information).

We compare the performance of each proposed solution approach with that of a policy optimized under information level $\mathbf{L}_2$. We restrict our analysis to the case where $M = 2$ as it facilitates visualization of the trained policies. Furthermore, we assume that all component replacements come from the same pool of components, meaning that $\Lambda_m \overset{\text{d}}{\equiv} \Lambda$ and $\boldsymbol{\Phi}_m \overset{\text{d}}{\equiv} \boldsymbol{\Phi}$. For each instance of the simulation study, the true hyperparameters of the gamma distribution Λ and the beta distribution $\boldsymbol{\Phi}$ that model the population heterogeneity are denoted by α, β, r and s, and are listed in Table 1. The means of the distributions Λ and $\boldsymbol{\Phi}$ are denoted by μ_Λ and $\mu_{\boldsymbol{\Phi}}$, respectively. These instances are selected to reflect increasing volatility in component heterogeneity, represented by an increasing coefficient of variation (CV) of the distributions, as well as a range of representative cost parameters.

All performance results in this section are retrieved using 10^6 repetitions of length 10^3 time units. The reported half-widths represent asymptotic 95% confidence intervals and account solely for variability within the model.

5.1. Performance analysis of heuristic solution approaches

For the instances of the simulation study, the two-threshold control limit (see Section 4.1) is optimized using a two-step simulation-based process. Given that all component replacements draw from the same pool of components, we can exploit this symmetry by assuming identical PM and OPM thresholds for each asset, i.e., $\tau_m^{\text{PM}} \equiv \tau^{\text{PM}}$ and $\tau_m^{\text{OPM}} \equiv \tau^{\text{OPM}}$. Initially, the PM threshold is optimized without OPM. Once this optimal PM threshold τ_*^{PM} is established, we focus on determining the optimal OPM threshold τ_*^{OPM} under the PM threshold τ_*^{PM}. This sequential approach not only simplifies the optimization process but also significantly reduces computational complexity. The optimization procedure for both instances I.1 and I.2 is visualized in Fig. 3. As anticipated, increasing corrective maintenance costs combined with greater variability in degradation process parameters results in a lower PM threshold τ_*^{PM}. Optimizing the OPM threshold further reduces costs by 3.78% for instance I.1 and 1.80% for instance I.2. The performance results for the two-threshold control limit $\pi_{\mathcal{N}}^{\mathbf{L}_1}$, the reactive heuristic $\pi_{\mathcal{R}}^{\mathbf{L}_1}$ and the integrated Bayes heuristic $\pi_{\mathcal{I}}^{\mathbf{L}_1}$ are summarized in Table 2. Notably, the two-threshold control limit outperforms the integrated Bayes heuristic, which performs poorly in the multi-asset scenario due to its disregard of economic dependence.

5.2. Improving heuristic solutions through approximate policy iteration

For both instances I.1 and I.2, we improve the reactive heuristic $\pi_{\mathcal{R}}^{\mathbf{L}_1}$ and the two-threshold control limit heuristic $\pi_{\mathcal{N}}^{\mathbf{L}_1}$ by applying three policy improvement steps using DCL in both $\mathbf{L}_1$ and $\mathbf{L}_2$ settings.

Table 2
Summary of the performance results of the heuristic solution approaches for the instances I.1 and I.2.

Instance	τ_*^{PM}	τ_*^{OPM}	$J(\pi_{\mathcal{N}}^{\mathbf{L}_1})$	$J(\pi_{\mathcal{R}}^{\mathbf{L}_1})$	$J(\pi_{\mathcal{I}}^{\mathbf{L}_1})$
I.1	15	9	22.645 ± 0.007	46.177 ± 0.012	24.715 ± 0.007
I.2	13	9	19.741 ± 0.011	65.069 ± 0.031	20.380 ± 0.010

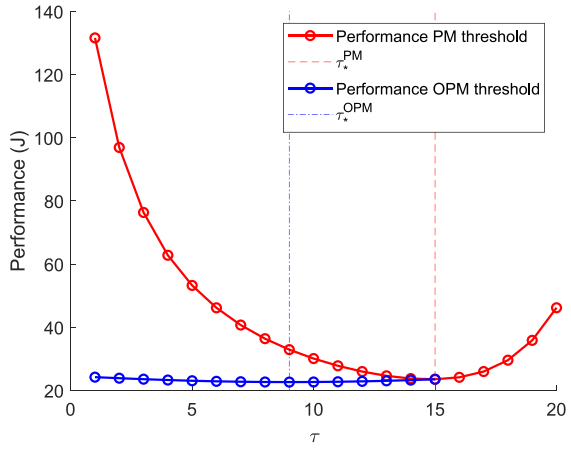
The training results for policies trained in the $\mathbf{L}_2$ setting (i.e., the underlying MDP) and their corresponding performance when applied in the $\mathbf{L}_1$ setting using the open-loop feedback approach are presented in Table 3. The training results for policies directly trained in the $\mathbf{L}_1$ setting (i.e., the BMDP) are presented in Table 4.

In conclusion, the results in Tables 3 and 4 demonstrate that initializing DCL with the two-threshold control limit $\pi_{\mathcal{N}}^{\mathbf{L}_1}$ significantly reduces the number of iterations needed to achieve near-optimal performance in both the information settings $\mathbf{L}_1$ and $\mathbf{L}_2$. The best-found neural network policies using the open-loop feedback approach $\pi_\theta^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h}))$ achieve similar performance to the best-found neural network policies $\pi_\theta^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h}))$ that are directly trained on the BMDP. The improvement of the best-found policy in setting $\mathbf{L}_1$ over the optimized two-threshold control limit is 4.14% (I.1) and 7.12% (I.2). This suggests that the value of integration is significant and increases with the volatility of component heterogeneity and the cost ratio $c_m^{\text{CM}}/c_m^{\text{PM}}$. However, the performance difference between the best-found policy in settings $\mathbf{L}_1$ and $\mathbf{L}_2$ is only 0.88% (I.1) and 1.32% (I.2), suggesting that the value of information is relatively small when the parameter uncertainty is managed effectively. Finally, all trained policies outperform the integrated Bayes heuristic $\pi_{\mathcal{I}}^{\mathbf{L}_1}$, which is the current state-of-the-art, with the best-found policy improving on it by 12.17% (I.1) and 10.03% (I.2).

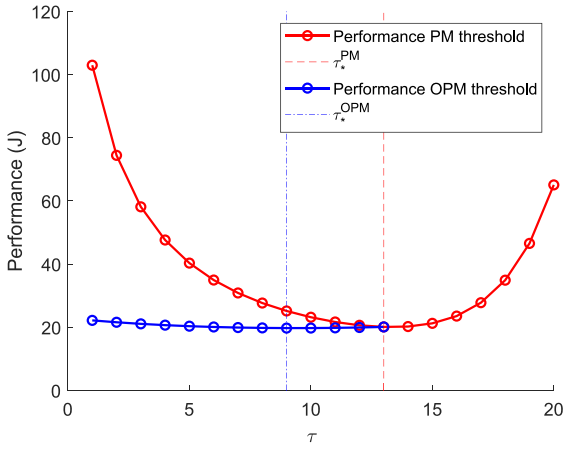
Lastly, to illustrate the effectiveness of the DRL approach in learning OPM strategies, we present two policy slices in Fig. 4 from the best-performing $\mathbf{L}_1$ neural network policy for instance I.1 (specifically, the 3rd-generation policy $\pi_{\theta_3}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h}))$ improving the two-threshold control limit policy $\pi_{\mathcal{N}}^{\mathbf{L}_1}$). These slices demonstrate that DCL learns, in just a few iterations, a complex transformation from PM to OPM decisions—one that cannot be captured by a simple set of rules. Notably, from a managerial perspective, the minimum degradation level required for maintenance intervention generally decreases when there is an opportunity to share maintenance setup costs.

6. Case study on degradation data of interventional X-ray system filaments

In this section, we assess the performance of the approaches outlined in Section 4 on real-world degradation data of a component crucial to the functioning of medical imaging systems. Medical imaging devices, such as IXR systems, cost about one million USD, with annual maintenance expenses around 10% of the initial cost [48]. Over a typical 10-year lifespan, maintenance accounts for nearly half of the total ownership cost. X-ray tubes, the most expensive components of IXR systems, are critical for image-guided procedures but prone to failure, predominantly due to filament wear. During each imaging procedure, a high electric current is applied to the tungsten filament to generate electrons for the X-ray beam. This repeated heating causes small amounts of tungsten to evaporate, thinning the filament and forming a hotspot that eventually leads to failure [49]. Importantly, degradation increments occur only during X-ray exposures and only when the electric current applied to the filament is sufficiently high



(a) The best-found PM and OPM thresholds for instance I.1 are $\tau_*^{PM} = 15$ and $\tau_*^{OPM} = 9$, respectively.

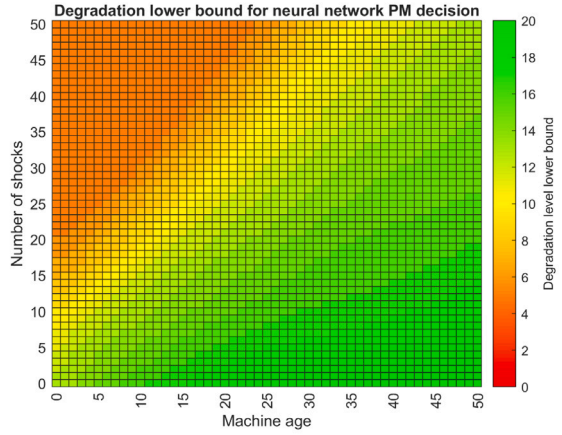


(b) The best-found PM and OPM thresholds for instance I.2 are $\tau_*^{PM} = 13$ and $\tau_*^{OPM} = 9$, respectively.

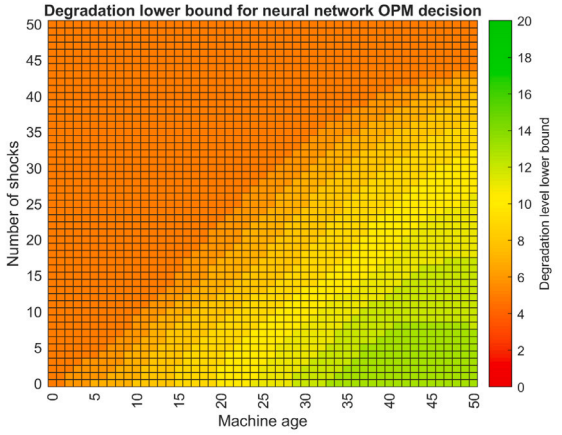
Fig. 3. Results from the two-step simulation-based optimization of the two-threshold control limit heuristic for the instances I.1 and I.2.

Table 3
One-step policy improvement results for instances I.1 and I.2. *Gray rows:* The performance of the neural network policy $\pi_\theta^{\mathbf{L}_2}$ in the $\mathbf{L}_2$ setting, trained on the underlying MDP. *White rows:* The performance of the neural network policy $\pi_\theta^{\mathbf{L}_2}$ applied in the $\mathbf{L}_1$ setting using the open-loop feedback approach. Bold: Indicates the lowest cost for each instance and information level across neural network generations.

		I.1		I.2	
	$\pi_0^{\mathbf{L}}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_R^{\mathbf{L}_1}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_R^{\mathbf{L}_1}$
Gen 0	$J(\pi_0^{\mathbf{L}})$	22.645 ± 0.007	46.177 ± 0.012	19.741 ± 0.011	65.069 ± 0.031
Gen 1	$J(\pi_{\theta_1}^{\mathbf{L}_2}(f_1^{\mathbf{L}_2}(h)))$	21.585 ± 0.007	34.225 ± 0.010	18.487 ± 0.010	28.208 ± 0.016
	$J(\pi_{\theta_1}^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h})))$	21.730 ± 0.007	36.522 ± 0.010	18.708 ± 0.010	38.188 ± 0.023
Gen 2	$J(\pi_{\theta_2}^{\mathbf{L}_2}(f_1^{\mathbf{L}_2}(h)))$	21.539 ± 0.006	25.138 ± 0.008	18.148 ± 0.010	20.523 ± 0.012
	$J(\pi_{\theta_2}^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h})))$	21.725 ± 0.007	25.659 ± 0.008	18.435 ± 0.010	22.892 ± 0.015
Gen 3	$J(\pi_{\theta_3}^{\mathbf{L}_2}(f_1^{\mathbf{L}_2}(h)))$	$\mathbf{21.518 \pm 0.006}$	24.258 ± 0.007	$\mathbf{18.097 \pm 0.010}$	19.674 ± 0.011
	$J(\pi_{\theta_3}^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h})))$	$\mathbf{21.709 \pm 0.007}$	24.761 ± 0.007	$\mathbf{18.376 \pm 0.010}$	21.514 ± 0.013

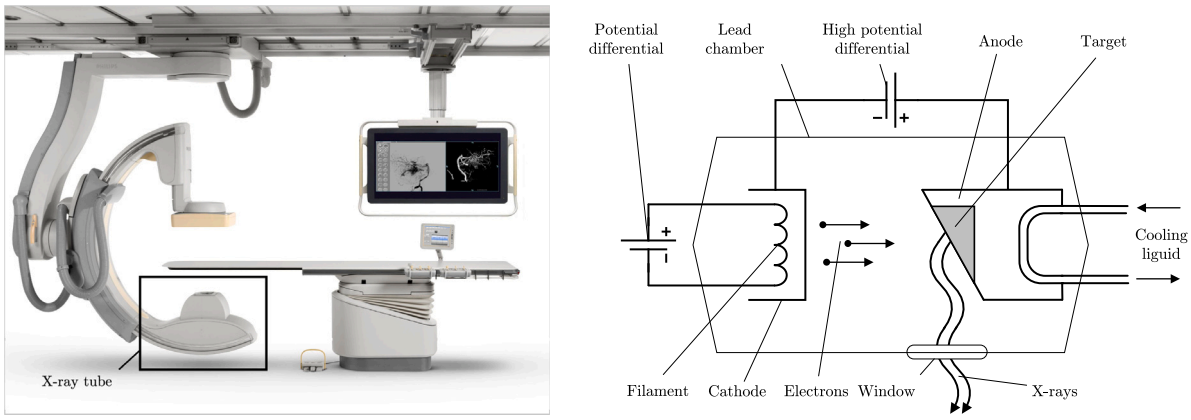


(a) Heatmap of the minimum degradation level x_1 at which maintenance is initiated, given $k_1(t)$ and $t_1(t)$, assuming the other machine is in the healthy state $\left(x_2(t), k_2(t), t_2(t)\right) = (0, 0, 0)$.



(b) Heatmap of the minimum degradation level x_1 at which maintenance is initiated, given $k_1(t)$ and $t_1(t)$, assuming the other machine is in the failed state $\left(x_2(t), k_2(t), t_2(t)\right) = \left(\xi_2, \frac{\xi_2 \cdot \mu_\Phi}{1-\mu_\Phi}, \frac{\xi_2 \cdot \mu_\Phi}{(1-\mu_\Phi)\cdot\mu_\Lambda}\right)$.

Fig. 4. Two policy slices from the best-performing neural network policy for instance I.1, illustrating the complex transformation from PM decisions to OPM decisions.



(a) An IXR system with the X-ray tube denoted by a rectangle.

(b) Simplified X-ray tube schematic.

Fig. 5. Diagram showing the positioning of the tungsten filament inside an IXR system.

Table 4
One-step policy improvement results for instances I.1 and I.2, where the neural network policies $\pi_{\theta}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h}))$ are trained directly on the BMDP. Bold: Indicates the lowest cost for each instance across neural network generations.

		I.1		I.2	
Gen 0	$\pi_0^{\mathbf{L}}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_{\mathcal{R}}^{\mathbf{L}_1}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_{\mathcal{R}}^{\mathbf{L}_1}$
	$J(\pi_0^{\mathbf{L}})$	22.645 ± 0.007	46.177 ± 0.012	19.741 ± 0.011	65.069 ± 0.031
Gen 1	$J(\pi_{\theta_1}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h})))$	21.834 ± 0.007	35.079 ± 0.010	18.611 ± 0.010	30.948 ± 0.018
Gen 2	$J(\pi_{\theta_2}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h})))$	21.729 ± 0.007	25.795 ± 0.008	$\mathbf{18.335 \pm 0.010}$	21.157 ± 0.012
Gen 3	$J(\pi_{\theta_3}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h})))$	$\mathbf{21.708 \pm 0.007}$	24.555 ± 0.007	18.340 ± 0.010	20.068 ± 0.011

to cause material degradation. Consequently, degradation accumulates through a sequence of operational events associated with system usage. In our modeling framework, these usage events are interpreted as the stochastic shocks that produce incremental degradation of the filament. We refer the reader to Fig. 5 for a schematic representation of an IXR system.

To prevent unnecessary unavailability of such scanning equipment, Philips, which is a manufacturer of the IXR system, has developed a health indicator to monitor filament wear, collecting real-time degradation data from each IXR system. This generates a large data set containing real-time degradation data, recorded as time-series of filament wear, from the start of each filament's lifespan until either failure or the present, for all IXR systems in operation. The data set used is obtained from Drent et al. [10, Section 7] and originates from Philips. In addition to manufacturing, Philips offers maintenance and service contracts to hospitals using their IXR systems. Dutch hospitals typically operate several IXR scanners across various departments.

This case study serves as a realistic industrial illustration of the proposed methodology. The modeling framework and DRL-based solution approach are not specific to medical imaging systems; rather, they apply to asset networks in which degradation can be represented by stochastic increment or shock processes with learnable parameters, such as wind turbines, semiconductor equipment, or other high-value capital goods.

6.1. Degradation data of X-ray tubes in IXR systems

The X-ray tube degradation data set consists of 52 time series, each representing the degradation level of a distinct X-ray tube. Let $\mathcal{I}$ denote the set of X-ray tubes for which data is available, i.e., $|\mathcal{I}| = 52$. The time series for each X-ray tube $i \in \mathcal{I}$ is denoted as $\mathcal{J}_i$. Each data point in the time series $\mathcal{J}_i$ is represented as a tuple $(t_j, x_j)_i$, where t_j is the age of the X-ray tube, and x_j is the degradation level at that age. Each tuple $(t_j, x_j)_i$ is generated when an IXR system is operated, and the time series contain between 20,000 and 300,000 data points, covering a time span of two to five years. These time series can be transformed into a series of BMDP trajectories, enabling us to directly evaluate the performance of our methods on the data.

For confidentiality reasons, the data was left-truncated and normalized. All time series start at $x_0 = 0$ for $t_0 = 0$ and end at $x_{|\mathcal{J}_i|} = 50$ (i.e., $\xi_m \equiv 50$ for all $i \in \mathcal{I}$). For each time series, the interarrival times $(t_j - t_{j-1})$ between consecutive data points and the degradation increments $(x_j - x_{j-1})$ were computed. To account for non-operational periods such as weekends, nights, or other extended downtimes, outliers in the interarrival times were removed. This allowed for the transformation of the original time series into those based on the operational age of each X-ray tube. Furthermore, several data points in the data where the image-guided procedure was deemed too short to cause significant wear on the X-ray tube were removed. Lastly, the time was normalized so that one unit of time corresponds roughly to the minimum operational time required for practical maintenance tasks, such as dispatching a service engineer to a hospital.

A statistical analysis of the resulting data set revealed that (i) there was no evidence to reject the assumption that shocks arrive according to a Poisson process, (ii) damage sizes are best modeled by a geometric distribution, and (iii) the data exhibits heterogeneity, meaning that the distribution parameters for interarrival times and shock sizes vary across components. Indeed, the estimated hyperparameters (see Table 5) indicate significant component heterogeneity. For the shock rate λ_m, the statistical analysis yields a mean of 1.414 and a CV of 0.157. Similarly, for the shock size parameter, the mean is 0.487 with a CV of 0.234.

6.2. Numerical experiments and model calibration

In this section, we outline the model calibration process, that is, estimating the hyperparameters of the degradation model. The data set introduced in the previous section is representative of the typical conditions encountered by the asset manager in practice, i.e., the baseline information level $\mathbf{L}_0$. To address the heterogeneity of the component population, we estimate the hyperparameters of the distributions Λ_m and Φ_m for each $m \in \mathcal{M}$ from the available X-ray tube degradation data using the maximum likelihood estimation procedure

Table 5
Hyperparameter settings and cost structures considered in the case study.

μ_Λ	CV_Λ	μ_Φ	CV_Φ	α	$1/\beta$	r	s
1.414	0.157	0.487	0.234	40.696	28.779	8.924	9.405

Instance	M	ξ_m	c_m^{PM}	c_m^{CM}	c^{ST}	γ
CS.1	1	50	1	5	0	0.99
CS.2	2	50	1	5	1	0.99
CS.3	5	50	1	5	1	0.99

Table 6
Summary of the heuristic solution calibration results for the case study instances CS.1–3.

Instance	τ_*^{PM}	τ_*^{OPM}	$J(\pi_{\mathcal{N}}^{\mathbf{L}_1})$	$J(\pi_R^{\mathbf{L}_1})$	$J(\pi_I^{\mathbf{L}_1})$
CS.1	40	–	3.146 ± 0.002	11.071 ± 0.006	2.974 ± 0.002
CS.2	41	28	11.381 ± 0.005	26.516 ± 0.010	11.558 ± 0.005
CS.3	41	29	26.405 ± 0.007	65.876 ± 0.016	28.270 ± 0.007

provided by Drent et al. [10, Online Appendix C]. As in Section 5, we assume that all component replacements stem from the same pool of components, i.e., $\Lambda_m \overset{\mathrm{d}}{\equiv} \Lambda$ and $\Phi_m \overset{\mathrm{d}}{\equiv} \Phi$. This assumption reflects asset-level homogeneity and is justified because, although the data may stem from distinct machines operated in different locations, the machines are of the same type, and the operating environment for each is controlled and standardized.

We divided the data set $\mathcal{I}$ into a training set $\mathcal{I}_{\mathrm{train}}$ with $|\mathcal{I}_{\mathrm{train}}| = 10$ and a test set $\mathcal{I}_{\mathrm{test}}$ with $|\mathcal{I}_{\mathrm{test}}| = 42$. The training set $\mathcal{I}_{\mathrm{train}}$ is a randomly selected subset of $\mathcal{I}$ used solely for parameter estimation, while the test set $\mathcal{I}_{\mathrm{test}}$ is reserved for evaluating our methods. For a detailed overview of the estimated parameters and cost settings used in this case study, refer to Table 5. The instance CS.1 features a single asset without consideration of economic dependence, which simplifies the problem to the maintenance problem studied in Drent et al. [10]. In contrast, instances CS.2 and CS.3, which involve two and five machines respectively, more accurately reflect the complexity of a real-world hospital setting.

For the calibration of the heuristic solution approaches, we optimize both the two-threshold control limit heuristic and the integrated Bayes heuristic for the fitted degradation model. Note that the integrated Bayes approach is specifically optimized for the cost ratio of CS.1, where $c_m^{\mathrm{CM}}/c_m^{\mathrm{PM}} = 5$, and the retrieved solution is used for the instances CS.2 and CS.3 as well. See Table 6 for a detailed summary of the heuristic solution calibration results. All performance results in this section are obtained from 10^6 repetitions, each lasting 10^3 time units. The reported half-widths again represent asymptotic 95% confidence intervals.

We improve the calibrated two-threshold control limit using 3 iterations of DCL, for which we present the results in Table 7 and Table 8.

The findings are mostly consistent with the results presented in Section 4. Notable is that the performance of the neural network policies $\pi_\theta^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h}))$ using the open-loop feedback approach is significantly worse compared to that of the neural network policies $\pi_\theta^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h}))$ directly trained on the BMDP. This decline in performance is likely attributable to a combination of several factors: (i) the substantial increase in component heterogeneity, as reflected by the higher CV of the distributions Λ and Φ; (ii) the limited number of observations available prior to component replacement; (iii) the reduction of the posterior distribution to a single point estimate, which entails a considerable loss of information; and (iv) the risk that unrepresentative initial data biases the inferred distribution parameters, potentially leading the neural network policy to trigger premature interventions.

Table 7
One-step policy improvement results for case study instances CS.1–3. *Gray rows:* The performance of the neural network policy $\pi_\theta^{\mathbf{L}_2}$ in the $\mathbf{L}_2$ setting, trained on the underlying MDP. *White rows:* The performance of the neural network policy $\pi_\theta^{\mathbf{L}_2}$ applied in the $\mathbf{L}_1$ setting using the open-loop feedback approach. Bold: Indicates the lowest cost for each instance under $\mathbf{L}_1$ across neural network generations.

		CS.1	CS.2	CS.3
Gen 0	$\pi_0^{\mathbf{L}}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$
	$J(\pi_0^{\mathbf{L}})$	3.146 ± 0.002	11.381 ± 0.005	26.405 ± 0.007
Gen 1	$J(\pi_{\theta_1}^{\mathbf{L}_2}(f_1^{\mathbf{L}_2}(h)))$	2.932 ± 0.002	10.564 ± 0.004	24.020 ± 0.006
	$J(\pi_{\theta_1}^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h})))$	3.936 ± 0.003	12.663 ± 0.006	28.498 ± 0.008
Gen 2	$J(\pi_{\theta_2}^{\mathbf{L}_2}(f_1^{\mathbf{L}_2}(h)))$	2.90 ± 0.002	10.471 ± 0.004	23.660 ± 0.006
	$J(\pi_{\theta_2}^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h})))$	$\mathbf{3.767 \pm 0.003}$	$\mathbf{12.527 \pm 0.006}$	$\mathbf{28.185 \pm 0.008}$
Gen 3	$J(\pi_{\theta_3}^{\mathbf{L}_2}(f_1^{\mathbf{L}_2}(h)))$	2.901 ± 0.002	10.516 ± 0.004	23.507 ± 0.006
	$J(\pi_{\theta_3}^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h})))$	3.804 ± 0.003	13.009 ± 0.007	29.336 ± 0.009

Table 8
One-step policy improvement results for case study instances CS.1–3, where the neural network policies $\pi_\theta^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h}))$ are trained directly on the BMDP. Bold: Indicates the lowest cost for each instance across neural network generations.

		CS.1	CS.2	CS.3
Gen 0	$\pi_0^{\mathbf{L}}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$	$\pi_{\mathcal{N}}^{\mathbf{L}_1}$
	$J(\pi_0^{\mathbf{L}})$	3.146 ± 0.002	11.381 ± 0.005	26.405 ± 0.007
Gen 1	$J(\pi_{\theta_1}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h})))$	2.975 ± 0.002	11.003 ± 0.005	24.765 ± 0.006
Gen 2	$J(\pi_{\theta_2}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h})))$	$\mathbf{2.957 \pm 0.002}$	10.703 ± 0.004	$\mathbf{24.197 \pm 0.006}$
Gen 3	$J(\pi_{\theta_3}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h})))$	2.959 ± 0.002	$\mathbf{10.601 \pm 0.004}$	24.605 ± 0.006

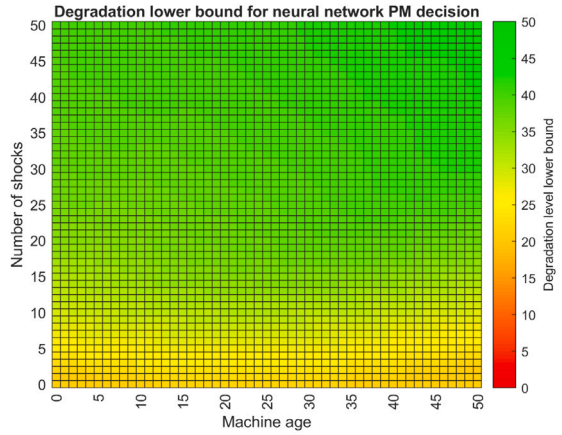
For instance CS.1, the best-found neural network policy in the $\mathbf{L}_1$ setting achieves comparable performance to the integrated Bayes heuristic. Both policies are depicted in Fig. 6 and exhibit a high degree of similarity.

An overlap analysis of Fig. 6 demonstrates a strong alignment between the two policies. Overall, 13.23% of the policy decisions were identical, and 65.36% differed by no more than 2% of the failure threshold ξ_m. These percentages increase significantly for the lower-left 20×20 subgrid, which contains more frequently visited states: 47.50% of values were identical, and 97.25% differed by at most 2% of ξ_m. These results indicate that DCL can produce policies that closely approximate the near-optimal integrated Bayes heuristic for a real-world component, especially in regions of the state space that are frequently encountered. Moreover, DCL produces policies with state-of-the-art performance in a multi-asset setting that more closely resembles real-world hospital conditions.

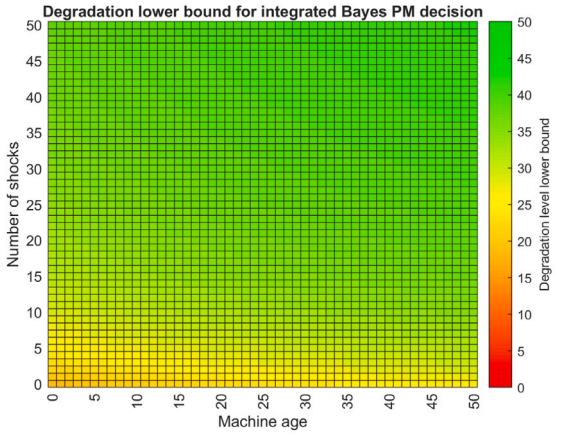
6.3. Numerical results and model validation

In this section, we assess the predictive accuracy and robustness of the best solution approaches identified in Tables 6–8 by evaluating their performance directly on the test data. Specifically, we analyze the ability of the model to generalize beyond the training data and evaluate how well the estimated hyperparameters of the distributions Λ and Φ capture the degradation behavior of other X-ray tubes across different case study instances. To this end, the time series in the test set $\mathcal{I}_{\mathrm{test}}$ are transformed into BMDP trajectories. All performance results are derived from 10^4 repetitions. The length of each repetition is set to 10^3, with trajectories sampled from $\mathcal{I}_{\mathrm{test}}$ with replacement. The reported half-widths correspond to asymptotic 95% confidence intervals.

In summary, the model validation results presented in Table 9 for case study instances CS.1-3 indicate that the fitted model generalizes



(a) Heatmap of the minimum degradation level x_1 at which maintenance is initiated according to the neural network policy, given $k_1(t)$ and $t_1(t)$.



(b) Heatmap of the minimum degradation level x_1 at which maintenance is initiated according to the integrated Bayes heuristic, given $k_1(t)$ and $t_1(t)$.

Fig. 6. Policy visualization of the best-performing neural network policy for instance CS.1 and the integrated Bayes heuristic, illustrating their similarity.

Table 9
Model validation results for the case study instances CS.1–3. Bold: Highlights the lowest cost achieved on the test set for each case study instance across the proposed solution approaches.

Instance	$J\left(\pi_{\theta_1}^{\mathbf{L}_2}(f_2^{\mathbf{L}_1}(\tilde{h}))\right)$	$J\left(\pi_{\theta_1}^{\mathbf{L}_1}(f_3^{\mathbf{L}_1}(\tilde{h}))\right)$	$J(\pi_{\mathcal{N}}^{\mathbf{L}_1})$	$J(\pi_{\mathcal{R}}^{\mathbf{L}_1})$	$J(\pi_I^{\mathbf{L}_1})$
CS.1	6.163 ± 0.058	4.278 ± 0.046	4.717 ± 0.050	11.270 ± 0.080	$\mathbf{4.175 \pm 0.045}$
CS.2	16.623 ± 0.099	$\mathbf{13.682 \pm 0.084}$	14.336 ± 0.087	26.956 ± 0.135	14.024 ± 0.087
CS.3	39.656 ± 0.143	$\mathbf{31.444 \pm 0.123}$	33.507 ± 0.128	66.875 ± 0.210	34.288 ± 0.135

well to unseen data. Integrating parameter estimation via the open-loop feedback approach results in a significant performance drop, as expected from the results in Table 7. In contrast, the $\mathbf{L}_1$ DCL-improved policies consistently outperform the benchmark solutions across all case study instances, except for the single asset scenario, where they achieve near-optimal performance (relative to the fitted model parameters in Table 5).

7. Conclusion

In this paper, we developed a novel maintenance model aimed at optimizing the management of a network of advanced industrial assets prone to costly and disruptive unplanned downtimes. By addressing the limitations of traditional condition-based maintenance models, which often assume homogeneous assets in isolation, our approach introduces a more practical framework that accounts for both asset heterogeneity and component heterogeneity, as well as economic dependencies between assets. For tractability purposes, the current formulation does not incorporate resource constraints. However, it can be extended to include shared resources—such as repair crews or spare parts—thereby allowing its application to asset networks with additional logistical complexities. Moreover, the degradation dynamics are modeled as a compound Poisson process with independent and identically distributed shock sizes. While this choice is standard in the reliability literature and supported by our case study data, it abstracts from certain practical complexities. In particular, we assume independent shock processes across assets and stationarity of degradation parameters within a component's lifetime, thereby excluding, for example, common-cause or environment-driven correlations and non-stationary or regime-switching degradation behavior. Extending the framework to correlated or non-stationary degradation processes constitutes a promising direction for future research.

Using a partially observable Markov decision process (POMDP) framework, our approach leverages real-time degradation data to learn cost-effective maintenance strategies for asset networks with economic dependencies. We further addressed the computational challenges of solving POMDPs by employing a deep reinforcement learning (DRL) approach, enabling the derivation of near-optimal maintenance policies under parameter uncertainty. The DRL-based approximate policy iteration algorithm demonstrated its effectiveness by learning complex opportunistic maintenance strategies, directly improving upon common heuristic methods.

Through theoretical contributions, including the establishment of the structural properties of optimal replacement policies of the underlying Markov decision process (MDP) model and the reformulation of the POMDP as a Bayesian MDP (BMDP), our approach facilitates scalable and efficient maintenance solutions for industrial-scale asset networks. However, the scope of the established structural properties is limited to the full-information setting and does not directly transfer to the partial-information case. Nevertheless, we expect that the monotonicity result in the degradation level also holds under partial information, and a similar monotonicity result could be established for another state variable, namely the operational age, which we leave for future work. Such structural properties could aid the development of heuristic methods tailored to the BMDP case.

The practical value of the proposed model was highlighted through a case study on degrading interventional X-ray system components, providing actionable managerial insights. The instances considered in this work are relatively small and assume asset-level homogeneity (i.e., all component replacements stem from the same pool), and therefore require relatively few samples to train our DRL algorithm; industrial-scale problems require many more, which can make single-node execution potentially prohibitive. By leveraging distributed computing and high-performance clusters, sample collection can be parallelized, keeping

computation times manageable even for large, complex industrial-scale instances. Our approach benefits significantly from smart initialization heuristics; developing a scalable initialization method for cases where heterogeneity is present at both the asset and component levels therefore appears promising.

However, when no computationally convenient parametric form exists (i.e., prior-to-posterior belief updating is not tractable) to model component heterogeneity, exact reformulation of the maintenance problem as a BMDP is no longer feasible. Alternative inference techniques, such as variational inference or Monte Carlo sampling methods, introduce approximations and significant computational challenges in the DRL algorithm. Consequently, it is not clear what the effect is on the trained DRL-based policies using such techniques. As a workaround, we proposed a solution where policies trained on the underlying MDP are applied in a POMDP setting using an open-loop feedback approach. However, our numerical experiments indicate that this solution is effective only in controlled settings and not when component heterogeneity is highly volatile, presenting an opportunity for future work.

Finally, the performance results of the trained policies under different information levels provide insight into the value of increasing the information level. In practice, additional information is obtained through investments in data availability, quality and processing infrastructure. The value of information can therefore be interpreted as a trade-off between such investment costs and the potential for improved CBM planning. Asset managers can evaluate this trade-off through scenario analysis to support investment decisions, which is a promising direction for future research.

CRediT authorship contribution statement

Peter Verleijsdonk: Writing – original draft, Visualization, Methodology, Investigation, Conceptualization. **Collin Drent:** Writing – review & editing, Methodology, Data curation, Conceptualization. **Stella Kapodistria:** Writing – review & editing, Supervision, Methodology, Funding acquisition, Conceptualization. **Willem van Jaarsveld:** Writing – review & editing, Supervision, Methodology, Funding acquisition, Conceptualization.

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.

Acknowledgments

This work used the Dutch national e-infrastructure with the support of the SURF Cooperative using grant no. EINF-5192. This work was supported by the Netherlands Organization for Scientific Research (NWO). Project: NWO Big data - Real Time ICT for Logistics. Number: 628.009.012. The work of Stella Kapodistria is supported by NWO through the Gravitation-grant NETWORKS-024.002.003. Collin Drent received support from NWO through Grant VI.Veni.241E.058.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.ress.2026.112773.

Data availability

The degradation data used for the case study is publicly available and the source is specified in the article.

References

[1] Coleman Chris, Damodaran Satish, Chandramouli Mahesh, Deuel Ed. Making maintenance smarter: Predictive maintenance and the digital supply network. Technical report, Deloitte; 2017, Retrieved September 9, 2022, from https://www2.deloitte.com/content/dam/insights/us/articles/3828_Making-maintenance-smarter/DUP_Making-maintenance-smarter.pdf.

[2] Wall Street Journal Custom Studios. How manufacturers achieve top quartile performance. 2017, Retrieved February 28, 2025, from https://partners.wsj.com/emerson/unlocking-performance/how-manufacturers-can-achieve-top-quartile-performance.

[3] Olde Keizer Minou CA, Flapper Simme Douwe P, Teunter Ruud H. Condition-based maintenance policies for systems with multiple dependent components: A review. European J Oper Res 2017;261(2):405–20.

[4] De Jonge Bram, Scarf Philip A. A review on maintenance optimization. European J Oper Res 2020;285(3):805–24.

[5] Arts Joachim, Boute Robert N, Loeys Stijn, Van Staden Heletjé E. Fifty years of maintenance optimization: Reflections and perspectives. European J Oper Res 2025;322(3):725–39.

[6] Elwany Alaa H, Gebraeel Nagi Z, Maillart Lisa M. Structured replacement policies for components with complex degradation processes and dedicated sensors. Oper Res 2011;59(3):684–95.

[7] Kim Michael Jong, Makis Viliam. Joint optimization of sampling and control of partially observable failing systems. Oper Res 2013;61(3):777–90.

[8] Chen Nan, Ye Zhi-Sheng, Xiang Yisha, Zhang Linmiao. Condition-based maintenance using the inverse Gaussian degradation model. European J Oper Res 2015;243(1):190–9.

[9] Van Oosterom Chiel, Peng Hao, Van Houtum Geert-Jan. Maintenance optimization for a Markovian deteriorating system with population heterogeneity. IISE Trans 2017;49(1):96–109.

[10] Drent Collin, Drent Melvin, Arts Joachim, Kapodistria Stella. Real-time integrated learning and decision making for cumulative shock degradation. Manuf Serv Oper Manag 2023;25(1):235–53.

[11] Mitici Mihaela, de Pater Ingeborg, Barros Anne, Zeng Zhiguo. Dynamic predictive maintenance for multiple components using data-driven probabilistic RUL prognostics: The case of turbofan engines. Reliab Eng Syst Saf 2023;234:109199.

[12] Zhuang Liangliang, Xu Ancha, Wang Xiao-Lin. A prognostic driven predictive maintenance framework based on Bayesian deep learning. Reliab Eng Syst Saf 2023;234:109181.

[13] Arcieri Giacomo, Hoelzl Cyprien, Schwery Oliver, Straub Daniel, Papakonstantinou Konstantinos G, Chatzi Eleni. Bridging POMDPs and Bayesian decision making for robust maintenance planning under model uncertainty: An application to railway systems. Reliab Eng Syst Saf 2023;239:109496.

[14] Tseremoglou Iordanis, Santos Bruno F. Condition-based maintenance scheduling of an aircraft fleet under partial observability: A deep reinforcement learning approach. Reliab Eng Syst Saf 2024;241:109582.

[15] Zhang Wenyu, Zhang Xiaohong, He Shuguang, Zhao Xing, He Zhen. Optimal condition-based maintenance policy for multi-component repairable systems with economic dependence in a finite-horizon. Reliab Eng Syst Saf 2024;241:109612.

[16] Esary JD, Marshall AW. Shock models and wear processes. Ann Probab 1973;1(4):627–49.

[17] Sobczyk K. Stochastic models for fatigue damage of materials. Adv in Appl Probab 1987;19(3):652–73.

[18] Temizöz Tarkan, Imdahl Christina, Dijkman Remco, Lamghari-Idrissi Douniel, Van Jaarsveld Willem. Deep controlled learning for inventory control. European J Oper Res 2025;324(1):104–17.

[19] Da Costa Paulo, Verleijsdonk Peter, Voorberg Simon, Akcay Alp, Kapodistria Stella, Van Jaarsveld Willem, Zhang Yingqian. Policies for the dynamic traveling maintainer problem with alerts. European J Oper Res 2023;305(3):1141–52.

[20] Wijnmalen Diederik JD, Hontelez Jan AM. Coordinated condition-based repair strategies for components of a multi-component maintenance system with discounts. European J Oper Res 1997;98(1):52–63.

[21] Bouvard K, Artus S, Bérenguer C, Cocquempot V. Condition-based dynamic maintenance operations planning & grouping. application to commercial heavy vehicles. Reliab Eng Syst Saf 2011;96(6):601–10.

[22] Tian Zhigang, Jin Tongdan, Wu Bairong, Ding Fangfang. Condition based maintenance optimization for wind power generation systems under continuous monitoring. Renew Energy 2011;36(5):1502–9.

[23] Tian Zhigang, Liao Haitao. Condition based maintenance optimization for multi-component systems using proportional hazards model. Reliab Eng Syst Saf 2011;96(5):581–9.

[24] Zhu Qiushi, Peng Hao, Van Houtum Geert-Jan. A condition-based maintenance policy for multi-component systems with a high maintenance setup cost. OR Spectr 2015;37:1007–35.

[25] Olde Keizer Minou CA, Teunter Ruud H, Veldman Jasper. Clustering condition-based maintenance for systems with redundancy and economic dependencies. European J Oper Res 2016;251(2):531–40.

[26] Olde Keizer Minou CA, Teunter Ruud H, Veldman Jasper, Babai M Zied. Condition-based maintenance for systems with economic dependence and load sharing. Int J Prod Econ 2018;195:319–27.
[27] Do Phuc, Assaf Roy, Scarf Phil, Iung Benoit. Modelling and application of condition-based maintenance for a two-component system with stochastic and economic dependencies. Reliab Eng Syst Saf 2019;182:86–97.
[28] Oakley Jordan L, Wilson Kevin J, Philipson Pete. A condition-based maintenance policy for continuously monitored multi-component systems with economic and stochastic dependence. Reliab Eng Syst Saf 2022;222:108321.
[29] Abdul-Malak David T, Kharoufeh Jeffrey P. Optimally replacing multiple systems in a shared environment. Probab Engrg Inform Sci 2018;32(2):179–206.
[30] Soltani Morteza, Kharoufeh Jeffrey P, Khademi Amin. Structured replacement policies for offshore wind turbines. Probab Engrg Inform Sci 2024;38(2):355–86.
[31] Leppinen Jussi, Punkka Antti, Ekholm Tommi, Salo Ahti. An optimization model for determining cost-efficient maintenance policies for multi-component systems with economic and structural dependencies. Omega 2025;130:103162.
[32] Si Xiaosheng, Li Tianmei, Zhang Qi, Hu Xiaoxiang. An optimal condition-based replacement method for systems with observed degradation signals. IEEE Trans Reliab 2018;67(3):1281–93.
[33] Flage Roger, Coit David W, Luxhøj James T, Aven Terje. Safety constraints applied to an adaptive Bayesian condition-based maintenance optimization model. Reliab Eng Syst Saf 2012;102:16–26.
[34] Kuhnle Andreas, Jakubik Johannes, Lanza Gisela. Reinforcement learning for opportunistic maintenance optimization. Prod Eng 2019;13(1):33–41.
[35] Zhang Nailong, Si Wujun. Deep reinforcement learning for condition-based maintenance planning of multi-component systems under dependent competing risks. Reliab Eng Syst Saf 2020;203:107094.
[36] Mohammadi Reza, He Qing. A deep reinforcement learning approach for rail renewal and maintenance planning. Reliab Eng Syst Saf 2022;225:108615.
[37] Hung Yu-Hsin, Shen Hong-Ying, Lee Chia-Yen. Deep reinforcement learning-based preventive maintenance for repairable machines with deterioration in a flow line system. Ann Oper Res 2024.
[38] Lee Juseong, Mitici Mihaela. Deep reinforcement learning for predictive aircraft maintenance using probabilistic remaining-useful-life prognostics. Reliab Eng Syst Saf 2023;230:108908.
[39] Verleijsdonk Peter, Van Jaarsveld Willem, Kapodistria Stella. Scalable policies for the dynamic traveling multi-maintainer problem with alerts. European J Oper Res 2024;319(1):121–34.
[40] Andriotis CP, Papakonstantinou KG. Deep reinforcement learning driven inspection and maintenance planning under incomplete information and constraints. Reliab Eng Syst Saf 2021;212:107551.
[41] Tian Feng, Sun Peng, Duenyas Izak. Optimal contract for machine repair and maintenance. Oper Res 2021;69(3):916–49.
[42] Drent Collin, Drent Melvin, van Houtum Geert-Jan. Optimal data pooling for shared learning in maintenance operations. Oper Res Lett 2024;52:107056.
[43] Drent Collin, Drent Melvin, Arts Joachim. Condition-based production for stochastically deteriorating systems: Optimal policies and learning. Manuf Serv Oper Manag 2024;26(3):1137–56.
[44] Wang Jia, Meng Bosen, Zhang Luyu, Yu Chengjiao. Degradation modeling and reliability estimation for mechanical transmission mechanism considering the clearance between kinematic pairs. Reliab Eng Syst Saf 2024;247:110093.
[45] Morris Carl N. Natural exponential families with quadratic variance functions. Ann Statist 1982;10(1):65–80.
[46] Ghosh JK, Delampady M, Samanta T. An introduction to Bayesian analysis: Theory and methods. Springer Science & Business Media; 2007.
[47] Shaked M, Shanthikumar JG. Stochastic orders. Springer series in statistics, Springer; 2007.
[48] ECRI. Healthcare product comparison system. Technical report, Plymouth Meeting, PA: ECRI Institute; 2013.
[49] Covington EJ. Hot spot burnout of tungsten filaments. J Illum Eng Soc 1973;2(4):372–80.