Reliability Engineering and System Safety 244 (2024) 109914

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

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

An efficient procedure for optimal maintenance intervention in partially observable multi-component systems

Oktay Karabağ [a,b,*], Önder Bulut [c], Ayhan Özgür Toy [c], Mehmet Murat Fadıloğlu [c]

[a] Erasmus School of Economics, Erasmus University Rotterdam, PO Box 1738, 3000, DR Rotterdam, the Netherlands
[b] Department of Industrial Engineering, İzmir University of Economics, Sakarya Caddesi No:156, Balçova, 35330, İzmir, Turkey
[c] Department of Industrial Engineering, Yaşar University, Üniversite Caddesi Ağaçlı, Yol No:37-39, Bornova, 35100, İzmir, Turkey

ARTICLE INFO

Keywords:
Condition-based maintenance
Spare part quantity
Markov decision process
Linear programming
Stochastic degradation
Partially observable systems

ABSTRACT

With rapid advances in technology, many systems are becoming more complex, including ever-increasing numbers of components that are prone to failure. In most cases, it may not be feasible from a technical or economic standpoint to dedicate a sensor for each individual component to gauge its wear and tear. To make sure that these systems that may require large capitals are economically maintained, one should provide maintenance in a way that responds to captured sensor observations. This gives rise to condition-based maintenance in partially observable multi-component systems. In this study, we propose a novel methodology to manage maintenance interventions as well as spare part quantity decisions for such systems. Our methodology is based on reducing the state space of the multi-component system and optimizing the resulting reduced-state Markov decision process via a linear programming approach. This methodology is highly scalable and capable of solving large problems that cannot be approached with the previously existing solution procedures.

1. Introduction

In today's world, many systems are becoming more complex and the number of components in these systems keeps on growing very fast. Examples include water pump systems for irrigation or mining purposes, off-shore power generation systems – which were the primary inspiration for work –, military and medical equipment, and so on. Such systems are mission-critical for their users. Hence, it is of utmost importance to keep them in operating condition via well-planned maintenance activities. Obviously, it makes more sense to embark on maintenance activities relying on some information about the current system condition, which is known as condition-based maintenance (CBM) planning in the literature.

CBM policies typically depend on condition monitoring information such as temperature, sound, vibration, or power consumption. Industry 4.0 involves a significant advance in real-time data collection and analysis of the obtained data with the goal of optimizing the performance of the underlying processes. This makes CBM the only viable solution for maintenance activities within the Industry 4.0 paradigm [1]. As a result, the use of CBM policies is becoming more popular in practice as well as in the scientific literature [2,3]. It is reported that type of CBM policies can achieve savings exceeding 50% on the maintenance costs [4]. For more detailed discussions on CBM policies and their applications, we refer to the studies of Keizer et al. [5], De Jonge and Scarf [3], and Quatrini et al. [6].

Due to technical, economic, and confidentiality restrictions, each component in a system may not have a dedicated sensor, implying that the exact condition of each component cannot be monitored. However, a summarized system-level signal can be still available, especially in remote settings such as off-shore wind farms or radar systems on vessels [7,8]. Specifically, wind farms constitute an excellent example that motivates what we present in the manuscript. Wind farms consist of a number of identical wind turbines operating individually and they are commonly installed in rural or remote areas, such as farms and ranches or coastal and island communities, where high-quality wind resources are often found. These farms typically belong to organizations that specialize and operate in the energy sector; therefore, most of the time, their maintenance and operations management are provided by a third-party entity, namely a maintenance provider. The electricity generated in wind farms is usually sold on a day-ahead market via auctions. In these auctions, having any piece of information – such as how many turbines are down, what is the condition of turbines, and etc. – that allows the bidders (wind farm owners) to estimate how much electricity will be potentially available to their competitors is

* Corresponding author at: Erasmus School of Economics, Erasmus University Rotterdam, PO Box 1738, 3000, DR Rotterdam, the Netherlands.
E-mail address: karabag@ese.eur.nl (O. Karabağ).

https://doi.org/10.1016/j.ress.2023.109914
Received 29 September 2023; Received in revised form 12 December 2023; Accepted 26 December 2023
Available online 4 January 2024

of vital importance to improve their profitability and increase their market shares.

Each turbine in wind farms is equipped with a number of high-tech sensors that collect data on wind speed, wind direction, generated power, turbine noise, etc. in a periodic fashion. In practice, this collected data set is known as *SCADA* (Supervisory Control and Data Acquisition) data. Even though the producers possess this rich and comprehensive data set, they are not willing to share it with any other third-party entity, even with their maintenance providers, due to the risk of potential abuse in the day-ahead electricity auction. Correspondingly, the service providers are usually permitted to collect only some partial information on the general condition of the farm. Hence, they decide when to make a site visit for the wind farms they are responsible for and how many spare parts/equipment to bring to the site based on this partial information.

Due to the incomplete nature of available information, there is always a risk of bringing an incorrect number of spare parts/equipment to the maintenance site. Specifically, if an unnecessary spare part/equipment is brought to the site, it will have to be returned at the end of the maintenance intervention. On the other hand, if the number of spare parts transferred to the site is not sufficient for all spare parts/equipment that have to be replaced, the maintenance provider would find itself in a position to undertake an emergency shipment that usually carries a high cost due to the remoteness of the site. This dynamic renders the spare part quantity decision – that has to be addressed simultaneously with the maintenance decision – nontrivial and crucial. Similar problems are also observed in water purifying systems [9], water pumps in series used in mining processes for dewatering [10], and industrial printing machines [7,11].

As can be seen from the given example, the presence of partial information complicates the implementation of CBM policies for multi-component remote systems. Such systems are discussed in detail in the studies of Karabağ et al. [7], Yeter et al. [12], Zhou et al. [13], and Karabağ et al. [14]. All these studies focus on the analysis of these kinds of systems with the purpose of devising effective CBM policies using well-known techniques such as POMDP and simulation optimization. However, these standard techniques typically necessitate impractical solution times and excessive memory requirements, hindering their application to any real-life setting of moderate size.

Among the studies in the relevant literature, the works developed by Karabağ et al. [7] and Karabağ et al. [14] are the closest to our research with the problem framework they propose. Our work differs from [7] and Karabağ et al. [14] in two crucial ways. Karabağ et al. [7] study the exact same problem setting as ours via a partially observable Markov decision process (POMDP) approach and employ a grid-based approximate solution methodology. This approach to numerically obtain an optimal policy for a POMDP was first proposed by Lovejoy [15]. However, as Karabağ et al. [7] note in their paper, the approach profoundly suffers from the curse of dimensionality, involves a discretization that is approximate, and cannot even solve problem instances with rather limited three components in less than 150 h (almost a week) with the help of a supercomputer. Their numerical results indicate that even though the authors have proposed a model that is theoretically capable of dealing with multi-component systems, their approach is impractical to handle systems of the size observed in most realistic settings. So, here, our main goal is to develop an alternative methodology capable of analyzing larger systems that may be encountered in real-life practices.

In order to achieve this goal, we reformulate the same problem as a Markovian decision process (MDP) that is very similar to the one in Karabağ et al. [7] and propose a novel and computationally efficient numerical solution procedure based on a linear programming approach to MDPs. With this new solution procedure, it is possible to get optimal solutions for problem instances with more than 10 components in less than an hour. A very similar solution procedure has been used in the work of Karabağ et al. [14] to study maintenance planning decisions for a single component system. However, our work is the first study to harness the advantage of this type of solution procedure for multi-component systems. Furthermore, our proposed procedure not only addresses when to invoke a maintenance intervention but also the decision on the number of spare parts to be taken along to the maintenance site. So, our attempt extends their work both methodologically and contextually as well as rendering the approach applicable to realistic settings.

In this work, we study CBM strategies addressing integrated maintenance and spare part quantity decisions for partially observable multi-component systems. The components are considered to be identical and they are subject to deterioration throughout the time. The system has a single sensor that provides only a three-level signal about the general condition of the system, i.e., the full information on the exact deterioration levels of components can only be observed through on-site inspection not through the sensor. This makes the system partially observable, resulting in a challenging optimization problem of when to make an intervention and how many spare parts to take along to the maintenance site. Considering such an optimization problem, we focus on three main research questions: (*i*) *Is it possible to obtain, in reasonable computation times, optimal maintenance and spare part quantity decisions for partially observable multi-component systems of practically meaningful sizes*? (*ii*) *How do system characteristics affect the system performance under the optimal policy*? (*iii*) *Under what circumstances does the optimal policy significantly outperform simpler heuristic policies*? (*iv*) *How does our approach scale when the number of components and the number of degradation states increase*?

We address the first research question we posed by proposing an extremely efficient approach compared to previously. Via this approach it is possible to optimally solve the larger problem instances that were introduced in Karabağ et al. [7] under an hour. Karabağ et al. [7] could not solve the same instances within a week using the standard POMDP approach. This demonstrates that our methodology delivers a viable tool for practically meaningful system sizes that should be instrumental for practitioners in conducting extensive numerical experiments. We believe that this is the major contribution of our work to the relevant research stream. To address the second and third research questions, we conducted an extensive numerical analysis. This analysis revealed that using simpler heuristic policies, such as no maintenance intervention until the system cannot operate, a.k.a. "*corrective maintenance policies*", can be very costly when the components in the system are significantly unreliable and/or there exists a small number of components in the system. This implies that using a CBM type of policy is more valuable for systems consisting of a small number of less reliable components than for those of a large number of more reliable components. It is also noteworthy that all heuristic policies result in significant optimality gaps, i.e. greater than 20% in most settings, underlining the need for this work. Such insights should help researchers for a better understanding of how system parameters affect the system performances. They are also invaluable for reliability engineers especially in the design phase of new systems because the types of sensors to be installed and the maintenance plan to be employed can be arranged accordingly. Lastly, the numerical experiments also reveal that our approach is capable of solving problems with 6 components and 7 deterioration levels under 20 min. Note that similar problem sizes could not be solved even with supercomputers in the literature [7].

The organization of the remaining part of this work is as follows. In Section 2, we review the relevant studies in the literature and position our work. In Section 3, we introduce the details of our problem description, mathematical model, and numerical solution procedure. In Section 4, we present an extensive numerical study and discuss the results of our findings. In Section 5, we provide a discussion of our key findings, and finish with proposing future research directions.

2. Literature review

Condition-based maintenance of multi-component and complex systems has attracted significant attention in recent years with the rapid advances in information technologies. The current state-of-art for this sizeable research stream is comprehensively reviewed in Alaswad and Xiang [2], Quatrini et al. [6], and De Jonge and Scarf [3]. In this section, we restrict our focus to previous research that our work directly improves upon. Specifically, we review only the pertinent studies that employ Markov decision models to study condition-based maintenance strategies for partially observable multi-component systems. Furthermore, for the sake of brevity and clarity, in this review, we cover solely the studies published after 2020. As alluded to earlier, the work of Karabağ et al. [7] is parallel to our work. In order to avoid unnecessary repetition and to clearly and concisely emphasize our positioning within the relevant research stream, for all relevant literature before 2020, we refer to their work.

It is noteworthy that in the short span of the last three years (2020–2023) research kept on being published at an astonishing rate. As an early work within this span, Shi et al. [16] study a problem of condition-based maintenance planning for a system consisting of serially connected, multi-component subsystems subject to a reliability requirement. The authors formulate the problem as a continuous-time Markov chain process and resort to a numerical method based on a simulation approach to obtain the optimal solutions. Zhang and Si [17] emphasize that finding optimal maintenance points is often suitable for low-dimensional systems but becomes challenging for high-dimensional CBM and highlight this phenomenon as a major limitation of most existing CBM research in the literature. In order to overcome such a challenge that may appear in a maintenance planning problem of systems consisting of economic- and stochastic-dependent components, they propose a data-driven method based on a deep Q-learning approach. Xu et al. [18] and Cheng and Zhao [19] also focus on how such economic- and stochastic-dependency among system components can affect optimal CBM strategies; but, different than previous works, they address new and divergent types of system structures and consider imperfect maintenance activities. Yousefi et al. [20] employ a very similar approach to that of Zhang and Si [17] to study CBM strategies for a multi-component system with individually repairable components, namely, stochastically and economically independent components. Liu et al. [21] address a system having two heterogeneous components whose degradation follows a bi-variate gamma process by using a Markovian modeling approach and reveal that for the case where the components' exact deterioration levels are fully observable, the optimal condition-based maintenance decision is a type of two-dimensional control-limit policy.

Liu et al. [22] consider the problem of obtaining the optimal maintenance policy for partially observed systems in various industrial maintenance practices, where online monitoring combined with more than one type of inspection with different diagnosis capabilities. Arismendi et al. [23] propose a framework to address cases where delays between decision-making and maintenance action execution may lead to further system deterioration, necessitating adjustments in the original maintenance plan. Andersen et al. [24] present a unified view of condition-based and time-based maintenance (TBM) strategies for a system whose components wear according to a multivariate gamma process with Lévy copula dependence. The authors also investigate the computational limitations of their TBM and CBM models and state that their procedure based on a dynamic programming approach is getting significantly slower as the number of components in the system becomes more than six or seven. Analogously, Zhang et al. [25] use a Lévy process to model components' deterioration; however, different than Andersen et al. [24], the authors address a k-out-of-n system. Hao et al. [26] employ a simulation-based approach to study a condition-based maintenance problem regarding k-out-of-n systems, where components' deterioration processes may differ due to different workloads, usage rates, or environmental stresses that are associated with their locations. Zhang et al. [27] adopt a Markovian decision process to develop a CBM model for a system having multiple components whose degradation processes not only depend on their intrinsic characteristics but also on their common operating environment. Under the assumption that the components' deterioration levels are fully observable, they derive the structural properties of the maintenance policy and the optimal maintenance cost. Hu et al. [28] also examine a very similar problem to that of Zhang et al. [27], incorporating imperfect maintenance operations.

Li et al. [29] develop a CBM strategy model that integrates maintenance, product quality, and working schedule for manufacturing systems. Sanoubar et al. [30] investigate the problem of implementing condition-based maintenance for a set of geographically distributed assets considering a single maintenance resource that travels between the locations of these assets. Guo and Liang [31] adapt a Markovian modeling approach for the joint optimization of successive inspection times and maintenance decisions for multi-component systems. Furthermore, this literature stream has been expanding with studies that integrate condition-based maintenance planning with other types of decisions related to inventory control [32–34], production control [35–40], spare part selection [7], components' deterioration levels balancing [41,42], and maintenance team scheduling/allocation [43,44].

Our work is distinguished from the rest of the literature since it provides a scalable optimization approach that does not suffer from the dreaded curse of dimensionality. Hence, it is applicable to systems with many components in a way that other MDP models are not.

3. The model

The system we consider comprises C identical components. Each component deteriorates over time by making transitions among a finite number of levels from 0 to K. The components and their deterioration levels are represented by $i = 1, 2, \dots, C$ and $d_i \in D$ where $D = \{0, 1, 2, \dots, K\}$, respectively. For each component i, the deterioration levels are ordered in a way that 0 represents the level at which the component works perfectly and K represents the level at which the component has failed and requires replacement. The levels in between, i.e., $d_i \in \{1, 2, \dots, K-1\}$, represent increasing levels of degradation that do not prevent the component's operation. The degradation of a component is always to the next level. Under the setting we consider herein, an intervention is required whenever at least one of the components is at degradation level K or it is through a decision at other degradation levels. An intervention always restores all components to perfect working conditions, i.e., $d_i = 0, \forall i$. Hence, we will use the terms *intervention* and *restoration* interchangeably, hereafter. Independent of their degradation levels, all components are set to halt (system stoppage) before intervention. The intervention duration is the time period from the system stoppage till the resumption of the system upon restoration of all components. We assume that the duration of an intervention is constant and known.

In our setting, the full information on the exact deterioration levels of components can only be observed through on-site inspection. However, the system has a single sensor that provides partial information via a system-wide, three-level signal denoted by $\sigma \in \{0, 1, 2\}$: *(i)* $\sigma = 0$ (a green signal), when all components are at the deterioration level 0 ($\forall i, d_i = 0$); *(ii)* $\sigma = 2$ (a red signal), when at least one component is at level K ($\exists i : d_i = K$); and *(iii)* $\sigma = 1$ (a yellow signal), otherwise ($\forall i, d_i < K, \exists i : d_i > 0$). Intervention decisions are based on the partial information by these signals. We assume that the system operates with *jidoka (autonomation) principle* where all components automatically stop working upon an intervention decision [45]. Alluded to above, the system-wide stoppage creates an opportunity for all components for restoration (see, Berk and Toy [46]).

The sensor transmits the signal periodically. We assume that the transmission interval (period) of the signals is sufficiently short such that at most one level of degradation may occur for each component. That is, in each period, the deterioration level of a component either increases by one with probability α or remains the same. An intervention is required when $\sigma = 2$ (upon receiving a red signal) and optional when $\sigma = 1$ (upon receiving a yellow signal). No intervention is needed when $\sigma = 0$ (a green signal) since the exact deterioration levels of all components are known to be zero; hence, it would not result in any benefit. However, in our modeling approach, we allow such sub-optimal interventions for completeness.

Every period, after observing the signal, the decision maker decides on whether or not to perform an intervention based on the signal history denoted by $\boldsymbol{\sigma}$. In our setting, an intervention restores all components to deterioration level 0 by replacement, hence the relevant signal history for the decision making is the signals since the last restoration. Upon an intervention decision, a site visit is required, before which the decision maker also needs to determine the number of components to be transferred for replacement. We denote the aggregate decision of visiting and the number of components to be transferred, henceforth will be called *action*, with a single variable $a \in A$ where $A = \{0, 1, 2, \ldots, C\}$. When $a = 0$, the decision is not to perform a site visit; when $a > 0$, the decision is to perform a site visit with a components.

For illustration of the relationship between signal history $\boldsymbol{\sigma}$ and action a, consider the following realizations:

(i) Suppose the signal history since the last regeneration is $\boldsymbol{\sigma} = \{0, 1, \ldots, 1, 2\}$, then a site visit is required with a particular number of components, i.e., $a > 0$; in which case the next signal after the restoration is $\sigma = 0$ with the signal history $\boldsymbol{\sigma} = \{0\}$.

(ii) Suppose now the signal history since the last regeneration is $\boldsymbol{\sigma} = \{0, 1, \ldots, 1\}$, then a site visit is optional ($a > 0$ in case of a site visit, and $a = 0$ otherwise) with a particular number of components, i.e., $a \geq 0$; in which case the next signal is $\sigma = 0$ with the signal history $\boldsymbol{\sigma} = \{0\}$ when $a > 0$, and $\sigma = 1$ or 2 with the signal history $\boldsymbol{\sigma} = \{0, 1, \ldots, 1\}$ or $\boldsymbol{\sigma} = \{0, 1, \ldots, 1, 2\}$, respectively, when $a = 0$.

(iii) Lastly, suppose the signal history since the last regeneration is $\boldsymbol{\sigma} = \{0\}$, then a site visit is optional ($a > 0$ in case of a site visit, and $a = 0$ otherwise) with a particular number of components, i.e., $a \geq 0$; in which case the next signal is $\sigma = 0$ with the signal history $\boldsymbol{\sigma} = \{0\}$ when $a > 0$, and $\sigma = 0$ or 1 with the signal history $\boldsymbol{\sigma} = \{0\}$ or $\boldsymbol{\sigma} = \{0, 1\}$, respectively, when $a = 0$.

The *jidoka principle* and our assumption about intervention duration allow this duration to be of arbitrary length, i.e., longer or shorter than the length of a period. As it will be clear soon, the derivation below relies on modeling the long-run behavior of the system in terms of stochastically identical cycles which start and end with system restoration, namely, $d_i = 0, \forall i$; hence, without loss of generality, we assume that intervention lasts a single period.

Next, we discuss our cost structure. We assume that there are fixed and variable cost components of the intervention. The fixed cost component depends on the type of the most recent signal of the signal history, denoted by c_σ. The fixed cost is due to transportation to and from the system site, the labor of the necessary crew, and consumable parts. Note that in the literature, c_2 is referred to as corrective maintenance cost and c_1 as preventive maintenance cost [7]. Typically, the corrective maintenance cost c_2 is assumed to be greater than the preventive maintenance cost c_1 (for a detailed discussion, see, e.g., Alaswad and Xiang [2], Keizer et al. [5]), as is the case in our study. Additionally, without any loss of generality, we assume that c_0 is less than c_1 and is equal to zero.

As per the variable cost components of intervention, during an intervention, all deteriorated components, i.e., all i such that $d_i \geq 1$, are replaced with brand new components at a unit replacement cost of c_r, restoring the levels of all components to zero. Replacement of all deteriorated components necessitates the transfer of a sufficient number of components to the system site. In our setting, the number of components to be replaced, $Y \in \{1, 2, \ldots, C\}$, is a random variable at a decision instance, and its realization, $y(\mathbf{d})$, is observed only after the on-site inspection which reveals the exact deterioration level of all components $\mathbf{d} = (d_1, \ldots, d_C)$, i.e., $y(\mathbf{d}) = \sum_{i=1}^{C} \mathbb{1}_{\{d_i \neq 0\}}$. Based on the decision a, a transfer cost of c_t is incurred per unit of components transferred to the site. The decision a may result in an excess or a shortage of transferred components. In case of a shortage, $a < y(\mathbf{d})$, a second transfer of a sufficient number of components, $(y(\mathbf{d}) - a)$ with an expected value of $E[(Y - a)^+]$, is scheduled to replace the remaining deteriorated components at a unit transfer cost of c_s. In case of an excess, $a > y(\mathbf{d})$, all unused components $(a - y(\mathbf{d}))$ with an expected value of $E[(a - Y)^+]$, are transferred back at a unit cost of c_e. Note that in the corresponding expectation expressions, $(x)^+$ where x is a real number represents $\max\{0, x\}$. Under this cost structure, the expected cost of a single period (SPC) is:

$$SPC(a|\boldsymbol{\sigma}) = \begin{cases} 0, & \text{if} \quad a = 0 \\ c_\sigma + c_t a + c_r E[Y] + c_s E[(Y - a)^+] + c_e E[(a - Y)^+], & \text{if} \quad a > 0 \end{cases}. \tag{1}$$

Our objective is to identify the optimal intervention policy that minimizes the long-run expected cost rate. Note that an intervention policy maps an *action a* to each *signal history* $\boldsymbol{\sigma}$. That is, given the signal history, the intervention policy dictates if the site visit will take place and jointly how many components will be taken along in case of a site visit.

Our optimization procedure relies on modeling the system as a Markov Decision Process (MDP). First, we develop the probability distributions and system equations required to calculate the long-run expected cost rate for a given policy. Next, using the aforementioned probability distributions and system equations, we construct a Linear Programming (LP) model to obtain the optimal policy.

The evolution of exact deterioration levels, $\mathbf{d}$ (*the complete system information*), and signal history, $\boldsymbol{\sigma}$ (*the partial system information*), generates two separate stochastic processes. Henceforth, we will refer to the stochastic process of complete system information as the Core Process (CP) and the stochastic process of partial system information as the Observed Process (OP). The probability distributions and system equations will specifically include (*i*) the one-step transition probability distribution of CP, (*ii*) the unconditional probability distribution of the system state of CP, (*iii*) the probability distribution of the system state in CP given OP, (*iv*) the one-step transition probability distribution of OP, and (*v*) the balance equations and long-run expected cost rate function.

3.1. Core and observed processes

In this section, we first introduce the Core Process and then the Observed Processes. In addition, we provide the optimization model that we use to numerically determine the optimal maintenance and spare part quantity decisions.

3.1.1. Core process

The evolution of complete system information, $\mathbf{d}$, along with the *actions* gives rise to the Core Process (CP). Let $\{\Delta_n, n \geq 0\}$ denote the stochastic process CP with the state space $S_\Delta = \{\mathbf{d} = (d_1, d_2, \ldots, d_C) : \forall i, 0 \leq d_i \leq K \text{ and } d_i \in \mathbb{Z}\}$. Depending on the actions, the state space of CP can be reduced down to a sub-space of S_Δ.

Note that upon intervention at any state $\mathbf{d}$ or self-transition at state $\mathbf{d} = \mathbf{0}$ the system is restored to the state $\mathbf{d}' = \mathbf{0}$, therefore, the CP is a regenerative process. Since every regeneration point trigger stochastically identical cycles, the characteristics of a cycle reflect long-run characteristics of the system due to the renewal reward theorem [47, 48]. Consequently, without loss of generality, $n = 0$ when $\mathbf{d} = \mathbf{0}$. The time index now corresponds to the number of periods since the last regeneration point. For the completeness of system characterization, we will need to differentiate the boundary condition, i.e., $(\mathbf{d} = \mathbf{0}, n = 0)$,

Table 1
A list of notation.

C:	Total number of components.	i:	Component index.
K:	Last deterioration level.	d_i:	Deterioration level of component i, i.e., $d_i \in D$ where $D = \{0, 1, 2, \dots, K\}$.
D:	Set of deterioration levels.	$\mathbf{d}$:	C-dimensional vector encapsulating the deterioration levels of all components.
σ:	Signal type, i.e., $\sigma \in \{0, 1, 2\}$.	$\boldsymbol{\sigma}$:	Signal history.
A:	Set of admissible actions.	a:	Action that integrates the maintenance and spare part quantity decisions, i.e., $a \in A$ where $A = \{0, 1, 2, \dots, C\}$.
α:	Probability that the deterioration level increases by a single unit.	Y:	Total number of components required to be replaced in case of a site visit. It is a random variable and its realization is denoted by $y(\mathbf{d})$.
c_1:	Preventive maintenance cost.	c_2:	Corrective maintenance cost.
c_t:	Cost of transferring a single component to the site.	c_r:	Cost of replacing a single component in the site visit.
c_s:	Cost of bringing a single component to the site with an extra transfer option.	c_e:	Cost of returning a single component back.
$SPC(a\|\boldsymbol{\sigma})$:	Expected cost of a single period.	c_e:	Cost of returning a single component back.
$(.)^+$:	Ramp-up function, i.e., $\max(0, .)$.	$\mathbb{1}_{(.)}$:	Indicator function, i.e., it returns 1 if (.) holds; otherwise, it gives 0.
Δ_n:	Stochastic process of CP.	S_Δ:	State space for the stochastic process of CP.
n:	Number of periods since the last regeneration point.	$\mathbb{Z}$:	Set of integer numbers.
δ_i:	Increment in the deterioration level of component i, i.e., $\delta_i \in \{0, 1\}$.	Ω_n:	Stochastic process of OP.
S^n_Ω:	State space for the stochastic process of OP.	E^σ_Δ:	Partitioning of the state space of process CP.
$\Pi_{\sigma,n,a}$:	Limiting probability of being in state σ and taking decision a at period n.	$\mathbb{Z}^{\text{nonneg}}$:	Set of non-negative integer numbers, i.e., $\mathbb{Z}^{\text{nonneg}} = \{0, 1, 2, \dots, U\}$.
$\mathbb{Z}^+$:	Set of positive integer numbers, i.e., $\mathbb{Z}^+ = \{1, 2, \dots, U\}$.	U:	Truncation level.
ϵ:	Tolerance level.	$Z(\Pi_{\sigma,n,a})$:	Long-run expected cost rate.

and transition which result in $\mathbf{d}' = \mathbf{0}$, i.e., a self-transition or restoration. Hence, we will denote the next time index right before regeneration as n^+. In elaboration, time instance n^+ corresponds to the end of a cycle.

Although the system state of the last period of a cycle can be any state $\mathbf{d}$, the system at time n^+ is always in state $\mathbf{d}' = \mathbf{0}$ with intervention when $\mathbf{d} \neq \mathbf{0}$ and without intervention (a self-transition) when $\mathbf{d} = \mathbf{0}$. The one-step transition probabilities of CP for all admissible state–action pairs, $(\mathbf{d}, a)$, are:

Case 1: For $\{(\mathbf{d}, a) : \forall i, d_i < K \text{ and } a = 0\}$,

For a self-transition in state $\mathbf{d} = \mathbf{0}$,

$$P(\Delta_{0^+} = \mathbf{0} | \Delta_0 = \mathbf{0}) = (1 - \alpha)^C, \tag{2}$$

and, for other transitions,

$$P(\Delta_{n+1} = \mathbf{d}' | \Delta_0 = \mathbf{d}) = \prod_{i=1}^{C} \alpha^{\delta_i} (1 - \alpha)^{1 - \delta_i}, \text{ when } \mathbf{d} = \mathbf{0} \text{ and } \mathbf{d}' \neq \mathbf{0}. \tag{3}$$

$$P(\Delta_{n+1} = \mathbf{d}' | \Delta_n = \mathbf{d}) = \prod_{i=1}^{C} \alpha^{\delta_i} (1 - \alpha)^{1 - \delta_i}, \text{ when } n \geq 1, \mathbf{d} \neq \mathbf{0}, \text{ and } \mathbf{d}' \neq \mathbf{0}, \tag{4}$$

where $\mathbf{d}' = (d_1 + \delta_1, d_2 + \delta_2, \dots, d_C + \delta_C)$ with $\delta_i \in \{0, 1\}$; and 0 for all other $\mathbf{d}'$ with $\delta_i > 1$. Note that the transition probability is positive only for $\mathbf{d}'$ where the increase in the deterioration level of any component, δ_i, is at most 1.

Case 2: For $\{(\mathbf{d}, a) : \mathbf{d} \in S_\Delta \text{ and } a > 0\}$,

$$P(\Delta_{n^+} = \mathbf{d}' | \Delta_n = \mathbf{d}) = 1, \text{ when } n \geq 0 \text{ and } \mathbf{d}' = \mathbf{0}, \tag{5}$$

and 0 for all other $\mathbf{d}'$. Note that the time index n^+ is the instance just before the beginning of the next cycle. Hence, no event can occur between time indices n^+ and 0, i.e., $P(\Delta_{n^+} = \mathbf{d}' | \Delta_n = \mathbf{d}) = P(\Delta_0 = \mathbf{d}' | \Delta_n = \mathbf{d})$.

Next, we can generate the probability of CP being in a state $\mathbf{d}$ at any given period n, i.e., the unconditional state distribution for intermediate periods of a cycle,

$$P(\Delta_n = \mathbf{d}) = \sum_{\mathbf{h} \in S_\Delta} P(\Delta_n = \mathbf{d} | \Delta_{n-1} = \mathbf{h}) P(\Delta_{n-1} = \mathbf{h}), \text{ when } n \geq 1, \tag{6}$$

and for the last period of the cycle with intervention,

$$P(\Delta_{n^+} = \mathbf{0}) = \sum_{\mathbf{h} \in S_\Delta} P(\Delta_{n^+} = \mathbf{0} | \Delta_n = \mathbf{h}) P(\Delta_n = \mathbf{h}), \tag{7}$$

with the boundary condition

$$P(\Delta_0 = \mathbf{0}) = 1. \tag{8}$$

3.1.2. Observed process

The above CP is valid under no information. However, due to our construction, a signal from the site, which contains partial information about the system is available. This partial information is a system-wide signal (σ) for the current and all previous periods. This set of information along with the *actions* is sufficient to construct another regenerative stochastic process, denoted as Observed Process (OP). For OP, the system state representation with minimal dimensions is obtained by keeping track of the current signal, σ, and the number of periods since the last regeneration point, n. We define the regeneration point of OP, in line with the definition for CP as the restoration of all components into deterioration level zero, as $\sigma = 0$ and $n = 0$. Now, let $\{\Omega_n, n \geq 0\}$ denote the stochastic process OP with the state space $S^n_\Omega = \{\sigma\}$ such that

$$\begin{cases} S^n_\Omega = \{0\}, & n = 0, \quad \text{(a)} \\ \{1\}, & n = 1, 2, \dots, K - 1, \quad \text{(b)} \\ \{1, 2\}, & n \geq K. \quad \text{(c)} \end{cases} \tag{9}$$

Note that (9)(a) holds due to our definition of the regeneration point; (9)(b) holds due to (9)(a) and transition into $\sigma = 2$ requires at least K transitions; (9)(c) holds due to (9)(b) and self transitions in CP generating $\sigma = 1$.

Formally, we can relate the CP $\{\Delta_n, n \geq 0\}$ and OP $\{\Omega_n, n \geq 0\}$ as follows. We first introduce the partition of the state space of CP and OP, with respect to σ. Let E^σ_Δ be a partition of the state space of process CP with respect to σ:

$$E^0_\Delta = \{\mathbf{d} : \forall i, d_i = 0\}, \; E^1_\Delta = \{\mathbf{d} : \forall i, d_i < K \text{ and } \exists i, d_i > 0\}, \; E^2_\Delta = \{\mathbf{d} : \exists i, d_i = K\}, \tag{10}$$

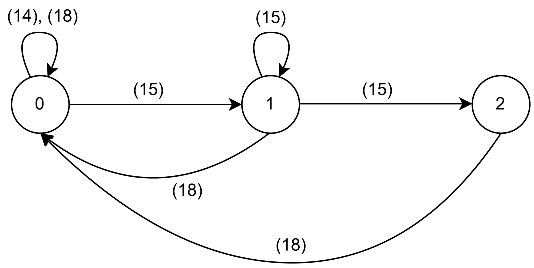

Fig. 1. The diagram for signal transition dynamics.

The relation between the two processes can be defined in terms of the relationship between the partition sets as follows:

$$(\sigma = 0) \iff E_\Delta^0, (\sigma = 1) \iff E_\Delta^1, (\sigma = 2) \iff E_\Delta^2. \tag{11}$$

Next, we show how the state distribution of CP can be updated upon receiving additional information from OP.

$$\begin{aligned} P(\Delta_n = \mathbf{d} | \Omega_n = \sigma) &= \frac{P(\Delta_n = \mathbf{d}, \Omega_n = \sigma)}{P(\Omega_n = \sigma)} \\ &= \frac{P(\Omega_n = \sigma | \Delta_n = \mathbf{d}) \times P(\Delta_n = \mathbf{d})}{\sum_{\mathbf{h} \in S_\Delta} P(\Omega_n = \sigma | \Delta_n = \mathbf{h}) \times P(\Delta_n = \mathbf{h})} \end{aligned} \tag{12}$$

From (11), $P(\Omega_n = \sigma | \Delta_n = \mathbf{h}) = 1$ when $\mathbf{h} \in E_\Delta^\sigma$; otherwise, it is zero. Hence, we have

$$P(\Delta_n = \mathbf{d} | \Omega_n = \sigma) = \begin{cases} 1, & \text{if } \sigma = 0 \text{ and } \mathbf{d} \in E_\Delta^0, \\ \dfrac{P(\Delta_n = \mathbf{d})}{\sum_{\mathbf{h} \in E_\Delta^1} P(\Delta_n = \mathbf{h})}, & \text{if } \sigma = 1 \text{ and } \mathbf{d} \in E_\Delta^1, \\ \dfrac{P(\Delta_n = \mathbf{d})}{\sum_{\mathbf{h} \in E_\Delta^2} P(\Delta_n = \mathbf{h})}, & \text{if } \sigma = 2 \text{ and } \mathbf{d} \in E_\Delta^2, \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

As the next step, we derive one-step transition probabilities of the OP, $P(\Omega_{n+1} = \sigma' | \Omega_n = \sigma)$, in terms of transition probabilities into states of CP in $E_\Delta^{\sigma'}$ for all admissible state–action pairs, (σ, a) as follows:

Case 1: For $\{(\sigma, a) : \forall n, \sigma < 2 \text{ and } a = 0\}$,

For a self-transition into the regeneration state, which follows from (11) and (2):

$$P(\Omega_{0^+} = 0 | \Omega_0 = 0) = P(\Delta_{0^+} = \mathbf{0} | \Delta_0 = \mathbf{0}) = (1 - \alpha)^C, \tag{14}$$

and, for all other transitions:

$$P(\Omega_{n+1} = \sigma' | \Omega_n = \sigma) = \sum_{\mathbf{h} \in E_\Delta^{\sigma'}} P(\Delta_{n+1} = \mathbf{h} | \Omega_n = \sigma) \tag{15}$$

where $\sigma' = 1$ and $\sigma = 0$ for $n = 0$, and $\sigma' = 1$ or 2 and $\sigma = 1$ for $n \geq 1$; with the transition probabilities into states of CP given that the observed process is in σ:

$$\begin{aligned} & P(\Delta_{n+1} = \mathbf{d} | \Omega_n = \sigma) \\ &= \sum_{\mathbf{h} \in S_\Delta} P(\Delta_{n+1} = \mathbf{d} | \Delta_n = \mathbf{h}, \Omega_n = \sigma) \times P(\Delta_n = \mathbf{h} | \Omega_n = \sigma), \qquad (16) \\ &= \sum_{\mathbf{h} \in E_\Delta^\sigma} P(\Delta_{n+1} = \mathbf{d} | \Delta_n = \mathbf{h}) \times P(\Delta_n = \mathbf{h} | \Omega_n = \sigma), \qquad (17) \end{aligned}$$

due to the implications in (11). Note that probability expressions in (17) are provided in (2)–(4) and (13).

Case 2: For $\{(\sigma, a) : \forall n, \sigma \in S_\Omega^n \text{ and } a > 0\}$,

$$\begin{gathered} P(\Omega_{n^+} = 0 | \Omega_n = \sigma) = 1, \\ \text{when } (\sigma = 0, n = 0) \text{ or } (\sigma = 1, 1 \leq n \leq K - 1) \text{ or } (\sigma \geq 1, n \geq K). \end{gathered} \tag{18}$$

We illustrate the signal transition dynamics in terms of one-step transition probabilities (14)–(18) for each time index n in Fig. 1.

3.2. Optimization model

Next, we provide the balance equations and long-run expected cost rate function below. All notations we used in both mathematical and optimization models are presented in Table 1. Let $\Pi_{\sigma,n,a}$ be the limiting probability of being in state σ and taking decision a at period n of a cycle in the non-homogeneous process OP. Then, the balance equations for the long-run state–action distribution (19)–(21) and the normalization (22) and non-negativity (23) conditions are:

$$\sum_{a \in A} \Pi_{1,n,a} = \sum_{\sigma \in \{0,1\}} P(\Omega_n = 1 | \Omega_{n-1} = \sigma) \times \Pi_{\sigma,n-1,0}, \text{ for } n \geq 1 \tag{19}$$

$$\sum_{a \in A \setminus \{0\}} \Pi_{2,n,a} = P(\Omega_n = 2 | \Omega_{n-1} = 1) \times \Pi_{1,n-1,0}, \text{ for } n \geq K \tag{20}$$

$$\sum_{a \in A} \Pi_{0,0,a} = P(\Omega_{0^+} = 0 | \Omega_0 = 0) \times \Pi_{0,0,0} + \sum_{a \in A \setminus \{0\}} 1 \times \Pi_{0,0,a} + \sum_{n \in \mathbb{Z}^+} \sum_{\sigma \in S_\Omega^n} \sum_{a \in A \setminus \{0\}} P(\Omega_{n^+} = 0 | \Omega_n = \sigma) \Pi_{\sigma,n,a} \tag{21}$$

$$\sum_{n \in \mathbb{Z}^{\text{nonneg}}} \sum_{\sigma \in S_\Omega^n} \sum_{a \in A} \Pi_{\sigma,n,a} = 1 \tag{22}$$

$$\Pi_{\sigma,n,a} \geq 0, \qquad \forall n \in \mathbb{Z}^{\text{nonneg}}, \forall \sigma \in S_\Omega^n, \text{and}, \forall a \in A \tag{23}$$

On the other hand, the long-run expected cost rate function is:

$$\begin{aligned} Z(\Pi_{\sigma,n,a} = \sum_{n \in \mathbb{Z}^+} \sum_{\sigma \in S_\Omega^n} \sum_{a \in A \setminus \{0\}} \Pi_{\sigma,n,a} \Big(c_\sigma + a c_t + \sum_{\mathbf{d}} \big[P(\Delta_n = d | \Omega_n = \sigma)(y(\mathbf{d}) c_r + (a - y(\mathbf{d}))^+ c_e \\ + (y(\mathbf{d}) - a)^+ c_s) \big] \Big). \end{aligned} \tag{24}$$

In Eq. (24), the first component given in parentheses represents the expected cost of having a site visit when the signal type is σ whereas the second component stands for the expected cost of having a shortage and an excess in the number of spare parts brought to the site visit.

The above set of equations is sufficient to develop a linear programming formulation to obtain optimal values for $\Pi_{\sigma,n,a}$ that subsumes the optimal set of actions for all σ and n. The linear programming formulation we implement is to *minimize* (24) *subject to* (19)–(23). Note that our model satisfies the standard conditions outlined in Puterman (2014), ensuring the existence of an optimal deterministic stationary policy. Hence, there is no reason to consider randomized policies for our model.

3.3. Computational implementation

The optimization model presented in the previous section is defined by the transition probabilities for the observed process. Since these transition probabilities are not readily available, they have to be generated from the transition probabilities of the core processes, as discussed in Section 3.1. Henceforth, we are going to call this part of our procedure "*pre-processing*".

The state space of the OP defined in the previous section is the set of all non-negative integer numbers. Hence, it is not computationally feasible to generate all one-step transition probabilities for the process in the pre-processing stage. However, the probability of reaching the time index n before a regeneration, $P\{S_\Omega^n\}$, is non-increasing in the time index n. We propose to stop the generation of the one-step transition probabilities when $P\{S_\Omega^n\} \leq \epsilon$, where ϵ is an arbitrarily small number. That is, we generate the transition probabilities up to a time index U that satisfies the given equation:

$$U = \arg\min_n (P\{S_\Omega^n\} \leq \epsilon). \tag{25}$$

In other words, we obtain the truncation level U by setting the probability of reaching the counter index U below a given tolerance level, ϵ. Given that in the optimal solution of LP the counter index U is visited with probability zero, we obtain an optimal solution for the unrestricted

Table 2
Parameter sets for the numerical experiments.

Parameter	*Set*	*Parameter*	*Set*	*Parameter*	*Set*
$(1-\alpha)$	$\{0.65, 0.75, 0.85, 0.95\}$	c_1	$\{100\}$	c_2	$\{100, 200, 400, 800\}$
K	$\{2, 3, 5\}$	c_t	$\{30\}$	c_r	$\{50\}$
C	$\{1, 2, 4\}$	c_e	$\{30\}$	c_s	$\{30, 60, 90\}$

problem. Otherwise, we keep decreasing ϵ, consequently increasing U, until the previous condition is satisfied.

The optimization model presented in Eqs. (19)–(24), needs to be adjusted upon the state-space truncation we propose. Note that it is actually not possible to directly use the given formulation, with any solver, even if we were able to generate the transition probabilities of infinite cardinality. The adjustment is implemented by redefining the sets of the time index in the optimization model as

$$\mathbb{Z}^{\text{nonneg}} = \{0, 1, 2, \dots, U\}, \tag{26}$$

$$\mathbb{Z}^{+} = \{1, 2, \dots, U\}, \tag{27}$$

and by adding a new constraint given by

$$\Pi_{1,U,0} = 0. \tag{28}$$

Eq. (28) forces the OP to regenerate at time index U, which ensures that the system does not exceed the time index limit. Upon executing the LP numerically in a solver environment (Gurobi Optimizer version 10.0.2 in Phyton for our case), if the given optimal solution entails the restoration of the system before the time index U, our solution is guaranteed to be optimal for the original LP of Eqs. (19)–(24), as well. For all numerical instances that we covered in our study, this was the case using an $\epsilon = 0.01$. If this were not the case, we would have to decrease the value of ϵ until $\sum_{\sigma \in S_{\Omega}^{U}} \sum_{a \in A} \Pi_{\sigma,U,a} = 0$ is satisfied (see [49,50] for similar implementations).

4. Numerical analysis

In this section, we introduce our experimental design. For each experiment, we report performance and policy metrics and compare the performance of the optimal policy with the naive heuristic policies we propose. Additionally, we discuss the effect of system parameters based on our metrics. We complete the section by providing a set of experiments that illustrates the scalability of our methodology. Note that all numerical results we obtained with the proposed procedure were verified by a simulation model.

4.1. Parameters

In our numerical experiments, we consider 5 different alternatives for the probability that the deterioration level increases by a single unit (α), 3 different alternatives for the last deterioration level (K), 3 different alternatives for the number of components in the system (C), 4 different alternatives for the corrective maintenance cost (c_2), and 3 different alternatives for the cost of bringing a single component to the site with an extra transfer option (c_s). On the other hand, we set the preventive maintenance cost (c_1), the transfer cost (c_t), the cost of returning a single component back (c_e), and the replacement cost (c_r) as 100, 30, 30, and 50, respectively. To examine how system parameters affect the system performance measures, we consider a full-factorial experimental design with all these alternatives (see, Table 2). That is, in total, we create a total of $4^2 \times 3^3 = 432$ different numerical instances. We defer the justification for the alternatives for the parameters to [7], where they are discussed in detail and justified within the context of a real-life case study. Note that the alternatives given in Table 2 are set in parallel with those of [7].

4.2. Heuristic policies & performance and policy metrics

In this work, we propose 6 different heuristic policies that are described in Table 3. Under the first group of heuristic policies (from 1 to 3), the service provider intervenes in the system when she receives a total of $(K-1)$ yellow signals. Note that it is not possible for the system to emit a red signal before observing $(K-1)$ yellow signals. This policy guarantees the system to be restored only via preventive maintenance interventions. So, in the literature, these types of policies are known as *preventive policies*. Under the second group of heuristic policies (from 4 to 6), the service provider intervenes in the system only upon receiving a red signal. This implies that the policy restores the system only via corrective maintenance interventions, which is known as *corrective policies* in the literature [2,7].

Table 3
List of heuristic policies considered to assess the performance of the optimal policy.

Policy code (H)	*Policy description*
1:	Intervene at $K-1$ taking an optimal number of parts
2:	Intervene at $K-1$ taking a single component
3:	Intervene at $K-1$ taking C components
4:	Intervene only upon receiving a red signal taking an optimal number of parts
5:	Intervene only upon receiving a red signal taking a single component
6:	Intervene only upon receiving a red signal taking C components

Both groups consist of three policies that differ in terms of the number of spare parts to be taken along with the service provider at the time of the intervention. Specifically, (*i*) for policies 1 and 4, the policy is to take the optimal number of parts, (*ii*) for policies 2 and 5, the policy is to take a single component, and (*iii*) for policies 3 and 6, the policy is to take the total number of components in the system C.

For each problem instance, we collect the optimal value of the objective function as defined in Eq. (24) and henceforth denoted as z^*. As an additional performance metric, we keep track of the long-run fraction of the time that the system is up under the optimal policy, which is represented by UP. Furthermore, the optimal preventive maintenance points, n^* are also recorded. Note that n^* denotes the optimal number of yellow signal counts to be received before invoking a preventive maintenance intervention. This count may not be reached if the system generates a red signal beforehand.

We also utilize the following performance indicator to assess the value of using the optimal policy compared to a particular benchmark heuristic policy:

$$\%I_H = \frac{z_H^* - z^*}{z^*} \times 100, \tag{29}$$

where z_H^* is the long-run average cost obtained for each problem instance with the use of the corresponding benchmark policy H and z^* is the long-run average cost obtained for each problem instance with the use of the optimal policy. Note that we collect this performance metric for each of the six heuristic policies, separately. Intuitively, the higher the value of $\%I_H$ means the worse the performance of the heuristic policy is compared to the optimal policy.

4.3. Experiments on policies and their performances

This section summarizes a wide set of numerical experiments that we employ to assess how system characteristics affect the performance/policy metrics and the benefit the optimal policy provides over the heuristic policies we proposed. In the following sections, we report the results of experiments in tables. Part (a) of each table provides the minimum, the maximum, and the average for the performance/policy

Table 4
Effect of total number of components on the policies.

(a) Performance/policy metrics for the optimal policy

Number of components (C)	z^*			n^*			UP		
	Mean	Min	Max	Mean	Min	Max	Mean	Min	Max
Low (1)	18.27	1.84	46.67	20.56	1.00	133.00	0.97	0.86	1.00
Medium (2)	31.01	3.37	76.77	17.52	1.00	104.00	0.96	0.81	1.00
High (4)	52.40	6.65	123.91	14.76	1.00	82.00	0.94	0.76	1.00

(b) Average cost degradation percentages for the heuristics

Number of components (C)	$\%\overline{I_1}$	$\%\overline{I_2}$	$\%\overline{I_3}$	$\%\overline{I_4}$	$\%\overline{I_5}$	$\%\overline{I_6}$
Low (1)	55.94	55.94	55.94	39.45	39.45	39.45
Medium (2)	57.11	58.17	84.03	28.76	32.29	31.24
High (4)	49.96	54.97	118.80	21.06	29.32	29.43

metrics across all experiments for the given factor level. Furthermore, in part (b) of each table, the averages for the performance indicators for the six heuristic policies are presented.

4.3.1. Effect of total number of components

Table 4 shows that as C increases, the long-run average cost increases since it is more costly to maintain systems with more units. Larger systems are also more likely to fail earlier since the failure of any component causes the system to become inoperative and to emit a red signal. Consequently, it is better to intervene earlier for larger systems. Under the optimal policy even though intervention is preponed, the system becomes more likely to experience failure before it is subject to preventive maintenance. In accordance, the gap between the corrective maintenance heuristics (policies 4–6) and the optimal policy decreases with the number of components (see Table 4). It should be noted that for a single-component system, there is no difference within the heuristic families (1–3) and (4–6), since the interventions all involve a single component change. The preventive maintenance heuristic that proposes to take the total number of components underperforms significantly in larger systems. At the preventive maintenance point, $(K-1)$, it is unlikely that many components start their degradation. This is not the case for the corrective maintenance heuristic. Overall, there is a significant benefit in applying the optimal policy instead of the heuristic.

4.3.2. Effect of the number of yellow states

An increase in K means effectively a higher resolution for the counter space. Table 5 indicates that the optimal policy takes advantage of this, delivering a drastic improvement in terms of costs. Since the system is likely to operate longer before failure, the optimal policy reacts by postponing preventive interventions, i.e., increasing n^*. Additionally, with a higher resolution for the counter space, the system is better equipped to protect itself from failures, thereby leading to a higher UP time (see Table 5). As the number of degradation levels increase, the preventive maintenance based heuristics rapidly become costlier, whereas the corrective maintenance based heuristics tend to cost less. This is in line with the intervention point being postponed and, therefore, the optimal policy resembling more and more the corrective maintenance heuristics. To take more parts along during the intervention works better for the corrective maintenance heuristics, whereas the opposite is true for the preventive maintenance heuristics. Again, the optimal policy brings to table significant improvements over the heuristics.

4.3.3. Effect of corrective maintenance cost

With an increase in the corrective maintenance cost, the service provider attempts to protect the system via earlier predictive interventions so as not to experience costly failures. One can confirm this strategy by observing the decrease in the n^* column of Table 6. Since this strategy results in a decrease in the failures and an increase in the up times is illustrated in the UP column of Table 6. The performance of the heuristic policies is also in line with the same observation. As the corrective maintenance cost increases, the optimal policy prepones the interventions, becoming closer to the preventive maintenance heuristics and further apart from the corrective maintenance heuristics. This is also reflected in the cost figures given in Table 6.

Additionally, we analyzed the effect of unit shortage cost (c_s) on our performance measures. However, it is observed that our measures were not sensitive to a change in the unit shortage cost. For the sake of succinctness, we do not provide the relevant results in this manuscript.

4.3.4. Effect of component deterioration characteristic

As α increases, the deterioration of components slows down. In this case, the service provider has no incentive to rush into a preventive intervention as manifested in the n^* column of Table 7. Since times between interventions get longer, the maintenance costs per unit time decrease and the up times increase.

On the other hand, Table 7 indicates that the gap between the corrective maintenance heuristics (policies from 4 to 6) and the optimal policy decreases as components become more reliable, that is, as α increases. With a reliable set of components, the service provider is better off postponing preventive maintenance interventions as much as possible. Correspondingly, the optimal policy begins to resemble the corrective maintenance heuristics, while significantly outperforming the preventive maintenance heuristics.

4.4. Experiments on scalability

The main contribution of the paper is to propose a scalable procedure that can be applied to more realistic problem sizes, compared to previous procedures available in the literature. To illustrate how the proposed procedure scales when the system is composed of a larger number of components and the components deteriorate according to a more detailed process with a larger state space, we provide the following set of experiments, in which $C \in \{2,3,4,5,6\}$ and $K \in \{3,5,7,9,11,13,15\}$. Note that the other system parameters are kept constant and each of them is set to the median of the corresponding levels given in Table 2. So, in total, we consider $5 \times 7 = 35$ different problem instances. In Table 8, for each problem instance, we present the execution times (in seconds) for the pre-processing stage (P_{time}) and the linear programming optimization (O_{time}). Additionally, in the same table, we provide the truncation levels (U) for each problem instance. Each problem instance is run using the Phyton code we developed, on a virtual machine provided by Google Colab (Intel(R) Xeon(R) CPU @2.20 GHz, 12 GB RAM). The optimization model is built within the Python code using Gurobi API and then solved via Gurobi optimizer.

The execution times for pre-processing stage increase with the number of components, as well as the size of the deterioration state-space, as we expected. However, it is interesting that the same increases do not seem to have any significant impact on the optimization times.

Table 5
Effect of the number of yellow states on the policies.

(a) Performance/policy metrics for the optimal policy

Number of yellow states $(K-1)$	z^*			n^*			UP		
	Mean	Min	Max	Mean	Min	Max	Mean	Min	Max
Low (2)	45.36	4.61	123.91	7.93	1.00	46.00	0.94	0.76	1.00
Medium (3)	35.23	3.07	107.55	15.56	2.00	78.00	0.96	0.84	1.00
High (5)	21.07	1.84	67.43	29.36	5.00	133.00	0.97	0.91	1.00

(b) Average cost degradation percentages for the heuristics

Number of yellow states $(K-1)$	$\%\overline{I_1}$	$\%\overline{I_2}$	$\%\overline{I_3}$	$\%\overline{I_4}$	$\%\overline{I_5}$	$\%\overline{I_6}$
Low (2)	20.10	20.15	57.57	41.59	43.24	49.45
Medium (3)	44.29	45.38	75.27	24.14	27.93	26.74
High (5)	98.62	103.56	125.92	23.54	29.89	23.92

Table 6
Effect of corrective maintenance cost on the policies.

(a) Performance/policy metrics for the optimal policy

Corrective maintenance cost (c_2)	z^*			n^*			UP		
	Mean	Min	Max	Mean	Min	Max	Mean	Min	Max
Very Low (100)	24.15	1.84	92.49	28.28	4.00	133.00	0.92	0.76	0.99
Low (200)	32.16	2.74	116.38	26.29	4.00	104.00	0.92	0.76	0.99
Medium (400)	38.64	3.69	123.91	9.63	1.00	50.00	0.99	0.95	1.00
High (800)	40.62	4.40	123.91	6.27	1.00	36.00	1.00	0.99	1.00

(b) Average cost degradation percentages for the heuristics

Corrective maintenance cost (c_2)	$\%\overline{I_1}$	$\%\overline{I_2}$	$\%\overline{I_3}$	$\%\overline{I_4}$	$\%\overline{I_5}$	$\%\overline{I_6}$
Very Low (100)	118.09	120.77	161.36	0.00	5.28	4.82
Low (200)	59.07	61.23	92.51	0.13	4.28	3.80
Medium (400)	25.51	27.24	52.09	23.27	26.58	26.30
High (800)	14.67	16.21	39.06	95.63	98.60	98.58

Table 7
Effect of component deterioration characteristic on the policies.

(a) Performance/policy metrics for the optimal policy

Component deterioration characteristic (α)	z^*			n^*			UP		
	Mean	Min	Max	Mean	Min	Max	Mean	Min	Max
Unreliable (0.65)	54.26	12.02	123.91	6.27	1.00	19.00	0.93	0.76	1.00
Fair (0.75)	42.45	8.80	99.40	8.52	1.00	26.00	0.95	0.81	1.00
Reliable (0.85)	28.20	5.39	69.77	14.13	1.00	44.00	0.96	0.86	1.00
Very Reliable (0.95)	10.66	1.84	29.88	41.55	1.00	133.00	0.99	0.95	1.00

(b) Average cost degradation percentages for the heuristics

Component deterioration characteristic (α)	$\%\overline{I_1}$	$\%\overline{I_2}$	$\%\overline{I_3}$	$\%\overline{I_4}$	$\%\overline{I_5}$	$\%\overline{I_6}$
Unreliable (0.65)	29.11	33.08	42.04	35.89	40.69	38.13
Fair (0.75)	40.02	42.85	59.91	31.99	36.15	35.10
Reliable (0.85)	57.64	58.94	89.89	27.99	31.64	32.06
Very Reliable (0.95)	90.58	90.58	153.18	23.15	26.26	28.20

Table 8
Pre-processing (P_{time}) and optimization (O_{time}) times (in seconds).

	$C=2$			$C=3$			$C=4$			$C=5$			$C=6$		
	P_{time}	O_{time}	U	P_{time}	O_{time}	U	P_{time}	O_{time}	U	P_{time}	O_{time}	U	P_{time}	O_{time}	U
$K=3$	0.005	0.030	20	0.007	0.030	17	0.080	0.030	15	0.180	0.030	14	1.666	0.030	13
$K=5$	0.004	0.030	35	0.045	0.030	31	0.609	0.030	28	2.547	0.030	26	62.955	0.030	25
$K=7$	0.011	0.030	50	0.052	0.030	44	1.577	0.030	41	24.349	0.030	39	1174.531	0.030	37
$K=9$	0.013	0.030	64	0.210	0.030	58	4.098	0.030	54	142.760	0.030	51	–	–	–
$K=11$	0.017	0.030	78	0.280	0.030	71	15.535	0.030	67	1086.269	0.030	64	–	–	–
$K=13$	0.031	0.030	93	0.498	0.030	85	43.532	0.030	80	–	–	–	–	–	–
$K=15$	0.047	0.030	107	0.862	0.030	98	115.792	0.030	93	–	–	–	–	–	–

This phenomenon is due to the fact that the optimization model is based on the observed process whose state space is independent of the set of core states. Although the cardinality of the set of core states grows exponentially with the number of components given by the formula $(K+1)^C$, the size of the observed process is $2\times(U+1)$ linearly increasing in the truncation level. Hence, the optimization effort does not suffer from the curse of dimensionality of the original problem (see, the given O_{times} in Table 8). However, with the limited RAM that was at our disposal, there were some problem instances for which we could not complete the pre-processing stage. These instances are denoted by dashes in Table 8. For such instances, it may be worth working further on the pre-processing part of our approach as future research.

Our procedure neutralizes the complexity of the original problem in the pre-processing stage, unburdening the optimization part of our procedure. In contrast, the optimization efforts in the previously available procedures increase exponentially as C or K grows. Consequently, our procedure is capable of handling the problems that were simply impossible to approach with the existing approaches. The previous state-of-the-art for the same problem can obtain an optimal solution for an instance with $K = 2$ and $C = 3$ in 154 h using a supercomputer with 12 cores and a total RAM capacity of 20.00 GB [7]. Note that our approach can solve similar problem sizes within milliseconds on a computer with regular specifications. We do strongly believe that we extend the current state-of-the-art significantly by providing an efficient solution procedure for conditional-based maintenance problems in partially observable multi-component systems.

5. Conclusion

In this work, we study maintenance planning for a partially observable multi-component system. The system periodically emits three-level signals that categorize its condition in terms of reliability. The control model proposed in the study addresses when to perform maintenance interventions and how many spare components to take along for the off-site intervention.

The work is based on a novel approach that introduces a process for the observed signals, the observed process (OP), and then constructs a Markov Decision Process (MDP) for the OP. The construction requires the derivation of conditional distribution of the underlying core states given the state of the observed process. The core states represent the joint condition of each component's deterioration level. The work supplies a computational procedure to obtain all the relevant distributions. Once the distributions are computed, the transition probabilities for the observed process as well as the period costs for the MDP can also be numerically obtained. The MDP is then solved using a linear programming (LP) approach.

The resulting MDP and LP are rather concise. The number of states in our approach is independent of the number of deterioration levels as well as the number of components while the state space of the core process grows exponentially. Hence, the optimization of the control model is incomparably efficient compared to previous approaches for similar systems [7]. This is a major contribution to the literature that enables application to considerably larger and more complex systems. In the manuscript, we used this to contribute insights for practitioners by conducting extensive numerical experiments based on a case study. Note that such extensive numerical experiments were computationally too costly with earlier approaches. Finally, the approach is inherently flexible with the potential for many other similar settings. We additionally conduct a set of scalability experiments to demonstrate the efficacy of our methodology in addressing challenges associated with increased problem scales. The previous state-of-the-art for the same problem can obtain an optimal solution for an instance with $K = 2$ and $C = 3$ in 154 h using a supercomputer with 12 cores and a total RAM capacity of 20.00 GB [7]. We would like to highlight that our methodology exhibits incomparable proficiency in addressing similar problem instances, providing a solution within milliseconds on a computing system with regular specifications.

The numerical experiments we present in this work reveal that when the cost ratio of preventive maintenance to corrective maintenance gets lower, the advantage of harnessing the optimal policy instead of corrective maintenance heuristics increases. Conversely, under the same conditions, the optimal policy's edge over preventive maintenance heuristics becomes less pronounced. The main reason behind this is that with an increase in the cost associated with corrective maintenance, the optimal policy prepones the interventions and thus becomes closer to the preventive maintenance heuristics. The numerical results also indicate that, when dealing with a reliable set of components, the service provider is more advantageous in deferring preventive maintenance interventions to the maximum extent possible. So, the optimal policy begins to resemble the corrective maintenance heuristics, while significantly outperforming the preventive maintenance heuristics. Last but not least, we observe that when the number of components increases, the long-run average maintenance cost the service provider should bear increases. Larger systems are also more likely to fail earlier since the failure of any component causes the system to become inoperative and emit a red signal. To avoid this, the service provider opts to intervene in such systems earlier, thereby increasing the frequency of maintenance interventions and the relevant average cost. Another important observation is that the optimization of the number of parts to take during intervention is more critical for systems with a larger number of components. We do believe that these observations hold significant implications for reliability engineers, especially during the design stage of new systems.

Our work can be extended in several ways. First of all, we currently posit the presence of an adequate inventory of spare parts at all times. Extending this work so as to cover inventory decisions would enable us to have a better understanding of how these types of decisions affect the system and the relevant performance metrics and provide better insights for practitioners. However, this might be a challenging extension with higher computational complexity. Second, in this work, we assume that in case of a maintenance intervention, all deteriorated components in the system are replaced with as-good-as-new components. This is likely not to be the best practice to reduce the cost as some mildly-deteriorated components can be used for some more time and then be replaced during the subsequent interventions. However, incorporating this kind of decision into the current model would be challenging not only in terms of computational complexity but also in terms of memory requirements necessary to keep up with the larger system state-space. Developing an algorithm that is capable of addressing this decision would be an interesting topic for future research. Finally, it would be valuable to render our model more data-oriented and adaptive to changes in the environment. Such an adaptation would be significantly instrumental for real world deployment by practitioners.

CRediT authorship contribution statement

Oktay Karabağ: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Önder Bulut:** Writing – review & editing, Writing – original draft, Methodology, Data curation, Conceptualization. **Ayhan Özgür Toy:** Writing – review & editing, Writing – original draft, Data curation, Conceptualization. **Mehmet Murat Fadıloğlu:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation.

Declaration of competing interest

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

Data availability

No data was used for the research described in the article.

References

[1] Bousdekis A, Lepenioti K, Apostolou D, Mentzas G. A review of data-driven decision-making methods for industry 4.0 maintenance applications. Electronics 2021;10(7):828.

[2] Alaswad S, Xiang Y. A review on condition-based maintenance optimization models for stochastically deteriorating system. Reliab Eng Syst Saf 2017;157:54–63.
[3] De Jonge B, Scarf PA. A review on maintenance optimization. European J Oper Res 2020;285(3):805–24.
[4] Zhang Q. Case study of cost benefits of condition-based maintenance used in medical devices. In: 2013 Proceedings annual reliability and maintainability symposium. IEEE; 2013, p. 1–5.
[5] Keizer MCO, Flapper SDP, Teunter RH. Condition-based maintenance policies for systems with multiple dependent components: A review. European J Oper Res 2017;261(2):405–20.
[6] Quatrini E, Costantino F, Di Gravio G, Patriarca R. Condition-based maintenance—An extensive literature review. Machines 2020;8(2):31.
[7] Karabağ O, Eruguz AS, Basten R. Integrated optimization of maintenance interventions and spare part selection for a partially observable multi-component system. Reliab Eng Syst Saf 2020;200:106955.
[8] Bakir I, Yildirim M, Ursavas E. An integrated optimization framework for multi-component predictive analytics in wind farm operations & maintenance. Renew Sustain Energy Rev 2021;138:110639.
[9] Zhang M, Revie M. Continuous-observation partially observable semi-Markov decision processes for machine maintenance. IEEE Trans Reliab 2016;66(1):202–18.
[10] Qazizada ME, Pivarčiová E. Reliability of parallel and serial centrifugal pumps for dewatering in mining process. Acta Montanistica Slovaca 2018;23(2).
[11] Botman L. Machine failure diagnostics and prognostics using error information and sensor data (Ph.D. thesis), Eindhoven University of Technology, Eindhoven; 2017.
[12] Yeter B, Garbatov Y, Soares CG. Risk-based maintenance planning of offshore wind turbine farms. Reliab Eng Syst Saf 2020;202:107062.
[13] Zhou X, van Gelder P, Liang Y, Zhang H. An integrated methodology for the supply reliability analysis of multi-product pipeline systems under pumps failure. Reliab Eng Syst Saf 2020;204:107185.
[14] Karabağ O, Bulut Ö, Toy AÖ. Markovian decision process modeling approach for intervention planning of partially observable systems prone to failures. In: International conference on intelligent and fuzzy systems. Springer; 2022, p. 497–504.
[15] Lovejoy WS. Computationally feasible bounds for partially observed Markov decision processes. Oper Res 1991;39(1):162–75.
[16] Shi Y, Zhu W, Xiang Y, Feng Q. Condition-based maintenance optimization for multi-component systems subject to a system reliability requirement. Reliab Eng Syst Saf 2020;202:107042.
[17] Zhang N, Si W. Deep reinforcement learning for condition-based maintenance planning of multi-component systems under dependent competing risks. Reliab Eng Syst Saf 2020;203:107094.
[18] Xu J, Liang Z, Li Y-F, Wang K. Generalized condition-based maintenance optimization for multi-component systems considering stochastic dependency and imperfect maintenance. Reliab Eng Syst Saf 2021;211:107592.
[19] Cheng W, Zhao X. Maintenance optimization for dependent two-component degrading systems subject to imperfect repair. Reliab Eng Syst Saf 2023;240:109581.
[20] Yousefi N, Tsianikas S, Coit DW. Reinforcement learning for dynamic condition-based maintenance of a system with individually repairable components. Qual Eng 2020;32(3):388–408.
[21] Liu B, Pandey MD, Wang X, Zhao X. A finite-horizon condition-based maintenance policy for a two-unit system with dependent degradation processes. European J Oper Res 2021;295(2):705–17.
[22] Liu X, Sun Q, Ye Z-S, Yildirim M. Optimal multi-type inspection policy for systems with imperfect online monitoring. Reliab Eng Syst Saf 2021;207:107335.
[23] Arismendi R, Barros A, Grall A. Piecewise deterministic Markov process for condition-based maintenance models—Application to critical infrastructures with discrete-state deterioration. Reliab Eng Syst Saf 2021;212:107540.
[24] Andersen JF, Andersen AR, Kulahci M, Nielsen BF. A numerical study of Markov decision process algorithms for multi-component replacement problems. European J Oper Res 2022;299(3):898–909.
[25] Zhang N, Tian S, Cai K, Zhang J. Condition-based maintenance assessment for a deteriorating system considering stochastic failure dependence. IISE Trans 2023;55(7):687–97.
[26] Hao Y, Zhu X, Kuo W. Optimization of condition-based maintenance with multiple times of component reallocation using Markov decision process. IEEE Trans Reliab 2023;1–11.
[27] Zhang N, Deng Y, Liu B, Zhang J. Condition-based maintenance for a multi-component system in a dynamic operating environment. Reliab Eng Syst Saf 2023;231:108988.
[28] Hu J, Xu A, Li B, Liao H. Condition-based maintenance planning for multi-state systems under time-varying environmental conditions. Comput Ind Eng 2021;158:107380.
[29] Li S, Yang Z, He J, Li G, Yang H, Liu T, et al. A novel maintenance strategy for manufacturing system considering working schedule and imperfect maintenance. Comput Ind Eng 2023;185:109656.
[30] Sanoubar S, De Jonge B, Maillart LM, Prokopyev OA. Optimal condition-based maintenance via a mobile maintenance resource. Transp Sci 2023.
[31] Guo C, Liang Z. A predictive Markov decision process for optimizing inspection and maintenance strategies of partially observable multi-state systems. Reliab Eng Syst Saf 2022;226:108683.
[32] Wang J, Zhu X. Joint optimization of condition-based maintenance and inventory control for a k-out-of-n: F system of multi-state degrading components. European J Oper Res 2021;290(2):514–29.
[33] Zheng M, Ye H, Wang D, Pan E. Joint optimization of condition-based maintenance and spare parts orders for multi-unit systems with dual sourcing. Reliab Eng Syst Saf 2021;210:107512.
[34] Zheng M, Lin J, Xia T, Liu Y, Pan E. Joint condition-based maintenance and spare provisioning policy for a K-out-of-N system with failures during inspection intervals. European J Oper Res 2023;308(3):1220–32.
[35] Uit het Broek MA, Teunter RH, De Jonge B, Veldman J. Joint condition-based maintenance and condition-based production optimization. Reliab Eng Syst Saf 2021;214:107743.
[36] Uit het Broek MA, Teunter RH, De Jonge B, Veldman J. Joint condition-based maintenance and load-sharing optimization for two-unit systems with economic dependency. European J Oper Res 2021;295(3):1119–31.
[37] Zhao X, He Z, Wu Y, Qiu Q. Joint optimization of condition-based performance control and maintenance policies for mission-critical systems. Reliab Eng Syst Saf 2022;226:108655.
[38] Koopmans M, De Jonge B. Condition-based maintenance and production speed optimization under limited maintenance capacity. Comput Ind Eng 2023;179:109155.
[39] Azizi F, Salari N. A novel condition-based maintenance framework for parallel manufacturing systems based on bivariate birth/birth–death processes. Reliab Eng Syst Saf 2023;229:108798.
[40] Zhang N, Cai K, Deng Y, Zhang J. Joint optimization of condition-based maintenance and condition-based production of a single equipment considering random yield and maintenance delay. Reliab Eng Syst Saf 2024;241:109694.
[41] Sun Q, Ye Z-S, Zhu X. Managing component degradation in series systems for balancing degradation through reallocation and maintenance. IISE Trans 2020;52(7):797–810.
[42] Wang X, Zhao X, Wu C, Lin C. Reliability assessment for balanced systems with restricted rebalanced mechanisms. Comput Ind Eng 2020;149:106801.
[43] Havinga MJ, De Jonge B. Condition-based maintenance in the cyclic patrolling repairman problem. Int J Prod Econ 2020;222:107497.
[44] Tseremoglou I, Santos BF. Condition-based maintenance scheduling of an aircraft fleet under partial observability: A deep reinforcement learning approach. Reliab Eng Syst Saf 2023;241:109582.
[45] Hoss M, Ten Caten CS. Lean schools of thought. Int J Prod Res 2013;51(11):3270–82.
[46] Berk E, Toy AÖ. Quality control chart design under Jidoka. Nav Res Logist 2009;56(5):465–77.
[47] Puterman ML. Markov decision processes: Discrete stochastic dynamic programming. John Wiley & Sons; 2014.
[48] Bertsekas DP. Dynamic programming and optimal control 4^{th} Edition, Volume II. Athena Scientific; 2015.
[49] Karabağ O, Gökgür B. Integrated optimisation of pricing, manufacturing, and procurement decisions of a make-to-stock system operating in a fluctuating environment. Int J Prod Res 2023;61(24):8423–50.
[50] Tan B, Karabağ O, Khayyati S. Production and energy mode control of a production-inventory system. European J Oper Res 2023;308(3):1176–87.