TU Delft

Delft University of Technology

Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling

Lee, Juseong; Mitici, Mihaela

DOI
10.1016/j.ress.2021.108123

Publication date
2022

Document Version
Final published version

Published in
Reliability Engineering and System Safety

Citation (APA)
Lee, J., & Mitici, M. (2022). Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling. *Reliability Engineering and System Safety*, *218*, Article 108123. https://doi.org/10.1016/j.ress.2021.108123

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Reliability Engineering and System Safety 218 (2022) 108123

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

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

Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling

Juseong Lee *, Mihaela Mitici

Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, HS 2926, Delft, The Netherlands

ARTICLE INFO

Keywords:
Reliability
Efficiency
Aircraft maintenance
Predictive maintenance
Multi-objective design
Design space exploration
Gaussian process learning

ABSTRACT

Aircraft maintenance design aims to identify strategies that render the aircraft reliable for flight in a cost-efficient manner. These are often conflicting objectives. Moreover, existing studies on maintenance design often limit themselves to only one type of maintenance strategy, overlooking other potentially dominating designs. We propose a framework for aircraft maintenance design with explicit reliability and cost-efficiency objectives. We explore the design space of a variety of maintenance strategies ranging from traditional time-based maintenance to predictive maintenance. To explore this design space, we propose an adaptive algorithm using Gaussian process learning and a novel adaptive sampling method. Gaussian process learning models rapidly pre-evaluate new maintenance designs, while adaptive sampling selects for further exploration only those designs that are expected to improve the available Pareto front of maintenance designs. This framework is illustrated for the maintenance of multi-component aircraft systems with k-out-of-n redundancy. The results show that novel predictive maintenance designs based on Remaining-Useful-Life prognostics dominate other maintenance designs, especially in the knee region of the obtained Pareto front, where the most beneficial balance between conflicting objectives is achieved. Our proposed exploration algorithm also outperforms other state-of-the-art exploration algorithms with respect to the quality of the Pareto front obtained.

1. Introduction

Aircraft maintenance is key for efficient and reliable aircraft operations, with airlines spending approximately 9.5% of the total operational costs for maintenance [1]. Currently, aircraft maintenance is designed based on Maintenance Steering Group-3 (MSG-3) [2], according to which critical aircraft components with respect to safety, economics, or operations are maintained at fixed time intervals. This strategy is referred to as time-based maintenance (TBM) [3]. TBM often relies on short time intervals (high-frequency) of maintenance tasks, in order to timely detect severe degradation of critical components, ensuring high reliability of aircraft. However, this high frequency of tasks may lead to higher costs with maintenance, or equivalently, to a decrease in maintenance efficiency.

To further improve the efficiency of aircraft maintenance, novel technologies such as on-board sensors and aircraft condition monitoring systems have been increasingly utilized. These systems provide (semi) real-time condition monitoring data for aircraft components. These new technologies and datasets promote condition-based maintenance (CBM) and predictive maintenance (PdM) strategies. Under CBM, maintenance tasks are scheduled based on the monitored health condition of components [4]. Under PdM strategies, the health condition data of components are further analyzed to predict their Remaining-Useful-Life (RUL), and to specify optimal maintenance schedules [5,6]. CBM and PdM are expected to improve the reliability and efficiency of aircraft maintenance, as shown in [7,8] for aircraft engine condition monitoring. Given these successes, the aviation industry is working towards the integration of CBM and PdM in the design of next-generation aircraft maintenance [9].

To justify the integration of CBM and PdM strategies in the maintenance of aircraft, their performance needs to be analyzed relative to existing and/or promising maintenance strategies. However, many existing studies limit themselves to optimizing only one specific maintenance strategy (e.g., either a PdM strategy, a CBM strategy, or a TBM strategy). For instance, [10] uses a genetic algorithm to optimize the design variables of a CBM strategy, but the dominance of this CBM strategy is not demonstrated against other maintenance strategies. Also, [11–13] explore only traditional TBM strategies with the aim of optimizing the inspection intervals. Even when studies compare the performance of their proposed maintenance strategies against benchmark strategies, the quantity and diversity of these explored maintenance strategies are limited, e.g., [14–16] consider as benchmark only a simple TBM strategy that replaces components at fixed

* Corresponding author.
E-mail address: J.Lee-2@tudelft.nl (J. Lee).

https://doi.org/10.1016/j.ress.2021.108123
Received 6 May 2021; Received in revised form 4 October 2021; Accepted 5 October 2021
Available online 14 October 2021
0951-8320/© 2021 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

intervals. In the context of designing the next-generation aircraft maintenance, it is necessary to analyze the dominance of novel strategies relative to other existing and/or promising TBM, CBM, PdM strategies.

At the same time, aircraft maintenance is inherently a multi-objective problem with two main, conflicting objectives: (1) to reliably operate the aircraft without incidents related to maintenance, and (2) to minimize maintenance cost [2,17]. Strict maintenance regulations [18,19] and system redundancy are in place to ensure system reliability, while intelligent scheduling of maintenance tasks aims to reduce the cost with maintenance. However, most existing studies focus on optimizing maintenance only from a monetary cost perspective, neglecting maintenance reliability. For example, [15,16] propose maintenance strategies such that component inspection and replacement costs are minimized. Some studies model reliability as a penalty cost: the cost of stopping system to prevent failure [14], the down-time cost during maintenance [20,21], the penalty cost of operating failed components [22], the unavailability cost due to a failed system [23,24]. Ultimately, these reliability-related costs are integrated in a single cost objective [14,20–24]. However, such a single-objective approach hides the potential trade-offs between reliability and monetary costs [25]. In general, existing studies lack a multi-objective approach for the design of (aircraft) maintenance, where both reliability and cost metrics are *explicitly* evaluated.

The main challenge in analyzing the multi-objective dominance between maintenance designs is the computational cost required, which increases significantly when considering many types of maintenance designs. Especially when the computational cost needed to evaluate the objectives of each design is significant, it is important to efficiently select for analysis only those designs that are expected to be dominating. This is known as a *design space exploration* problem [26]. Well-known approaches for this problem are meta-heuristic algorithms such as Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) [27,28]. NSGA-II iteratively selects designs to be analyzed, using operators specific to genetic algorithms such as mutation and crossover. More efficient, novel design space exploration algorithms utilize surrogate models to rapidly pre-estimate the objectives of the designs. For instance, Response Surface-based Pareto Iterative Refinement (ReSPIR) considers surrogate models such as linear regression and radial-basis-functions [29]. Gaussian process (GP) learning models are also often used for an efficient design space exploration [30,31]. For example, Efficient Global Optimization (EGO) utilizes GP models to calculate some infill-criteria, and select for analysis those designs that maximize these infill-criteria [32,33]. However, the global maximum of infill-criteria is often hard to find because of many local maxima, which may not result in the most efficient selection of new designs to be analyzed [32,34]. As such, further algorithmic improvements are needed to be able to efficiently select designs to be explored for aircraft maintenance.

In this paper, we propose a framework to design multi-objective aircraft maintenance with an emphasis on the trade-off between maintenance reliability and cost-efficiency. Our framework considers various types of maintenance strategies, and identifies Pareto optimal aircraft maintenance designs by adaptively exploring the design spaces of the considered maintenance strategies. For this, we construct a generic aircraft maintenance model that is used to evaluate multiple objectives related to the cost-efficiency and reliability of aircraft maintenance designs by means of Monte Carlo simulation. Since this simulation-based evaluation of maintenance designs is computationally expensive, we propose an adaptive design space exploration algorithm that iteratively identifies Pareto optimal maintenance designs using Gaussian process learning models and a novel adaptive sampling method. This adaptive sampling method uses Gaussian process learning models to pre-estimate the objectives of the designs. Using these pre-estimations, we select for further exploration only those designs whose pre-estimated objectives are not dominated by the currently available Pareto front.

Our framework is applied for the maintenance of aircraft landing gear brakes, which is a multi-component aircraft system with k-out-of-n redundancy. We show that the RUL-based PdM design achieves the most beneficial balance between cost-efficiency and reliability objectives, the maximization of the utilization of components being a cost-related objective, and the minimization of the expected number of degradation incidents being a reliability-related objective. The results show that across the domain of the aircraft maintenance design problem, there are both TBM, CBM and PdM strategies that are Pareto optimal, i.e., Pareto optimality is not achieved only by one type of strategy. The results also show that the RUL-based PdM design is located in the *knee region* of the Pareto front, where the most beneficial trade-off between the considered objectives is achieved.

The main contributions of this paper are as follows:

- We explicitly consider reliability-related objectives of aircraft maintenance, which are often neglected in existing studies on (aircraft) maintenance or indirectly considered in cost metrics. Using a multi-objective approach, we analyze the trade-offs between reliability and cost-efficiency metrics of aircraft maintenance.
- We are able to design maintenance by exploring a wide spectrum of types of maintenance strategies, ranging from TBM, CBM, to PdM. With our approach, we show that the reliability–cost-efficiency Pareto front of aircraft maintenance consists of a mix of the TBM, CBM and PdM strategies, rather than restricting Pareto optimality to only one type of strategy.
- We propose an efficient algorithm to explore the design space of aircraft maintenance using a Gaussian process (GP) learning model and a novel adaptive sampling method. Our algorithm is shown to outperform existing design space exploration algorithms in terms of the number of Pareto optimal designs generated.
- Our framework is expected to support decision-makers to quantitatively analyze novel maintenance strategies from both a reliability and cost-efficient perspective, and ultimately to facilitate a discussion on the integration of such novel maintenance strategies into the current paradigm of aircraft maintenance design.

The remainder of this paper is organized as follows. In Section 2, we formulate the problem of multi-objective aircraft maintenance design. In Section 3, we propose a framework to address this problem and an algorithm to adaptively explore the design space of aircraft maintenance. In Section 4, we design the maintenance of landing gear brakes, and discuss the benefits of novel predictive maintenance designs. In Section 5, we compare the performance of our proposed algorithm for the aircraft maintenance design problem, against several algorithms commonly used for multi-objective design problems. Conclusions are provided in Section 6.

2. Problem formulation: Multi-objective design of aircraft maintenance

We consider the multi-objective aircraft maintenance design problem, i.e., we are interested in identifying those maintenance designs that optimize a set of objectives. Formally, an aircraft maintenance design is specified by a tuple $(s, \mathbf{x}^s)$, where s is the strategy *type* and $\mathbf{x}^s$ is its associated *design variables*. As an example, consider two types of maintenance strategies A and B where:

- according to strategy type A, a component is replaced with a brand-new one every D_{Rep} flight cycles.
- according to strategy type B, a component is inspected every D_{Ins} flight cycles. Upon inspection, the component is replaced if its degradation level exceeds a threshold η_{Rep}.

The strategy type $s \in \{A, B\}$ specifies the rules according to which maintenance tasks are executed, and $\mathbf{x}^A = [D_{\text{Rep}}]$ and $\mathbf{x}^B = [D_{\text{Ins}}, \eta_{\text{Rep}}]$ are the vectors of design variables associated with strategy type A and B, respectively. In this example, the set of considered strategy types is $s \in S = \{A, B\}$, and the domains of the design variables are $\mathbf{x}^A \in \mathcal{X}^A = \mathbb{Z}^+$ and $\mathbf{x}^B \in \mathcal{X}^B = \mathbb{Z}^+ \times \mathbb{R}$. We define the *design space*

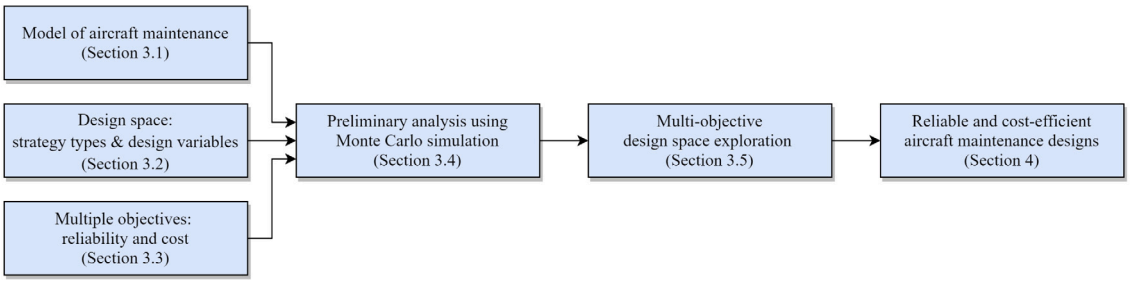

Fig. 1. Framework of multi-objective aircraft maintenance design.

D of aircraft maintenance by the set of maintenance strategy types considered ($s \in S$), and the domain of the design variables of strategy type s ($\mathbf{x}^s \in \mathcal{X}^s$), i.e.,

$$D = \{(s, \mathbf{x}^s) | s \in S, \mathbf{x}^s \in \mathcal{X}^s\}. \tag{1}$$

Here, $\mathbf{x}^s \in \mathcal{X}^s$ consists of both continuous and integer variables, while $s \in S$ is a non-numerical variable.

We explore the design space of aircraft maintenance to identify those maintenance designs $(s, \mathbf{x}^s)$ that result in an optimal maintenance performance. Here, the maintenance performance is defined in terms of $M > 1$ objectives. Let $\mathbf{f}(s, \mathbf{x}^s) = [f_1(s, \mathbf{x}^s), \dots, f_M(s, \mathbf{x}^s)]$ denote the vector of M objectives for a specific maintenance design $(s, \mathbf{x}^s)$. For our study, $\mathbf{f}$ consists of reliability-related objectives and cost-related objectives. Some of these objectives are potentially conflicting.

Then the design space exploration to identify those aircraft maintenance designs that optimize the set of M objectives is formalized as follows:

$$\begin{aligned} &\min/\max\ f_m(s, \mathbf{x}^s), m \in \{1, 2, \dots, M\} \\ &\text{subject to } (s, \mathbf{x}^s) \in D \end{aligned} \tag{2}$$

Since we have multiple objectives, the following Pareto dominance relations apply [35]. For simplicity, in the definition below, we assume that all objectives are minimized. We say that a maintenance design $(s_1, \mathbf{x}^{s_1})$ *dominates* another design $(s_2, \mathbf{x}^{s_2})$, denoted by $\mathbf{f}(s_1, \mathbf{x}^{s_1}) \succ \mathbf{f}(s_2, \mathbf{x}^{s_2})$, if and only if

$$\begin{aligned} &\left(\forall\, m \in \{1, \dots, M\},\ \ f_m(s_1, \mathbf{x}^{s_1}) \le f_m(s_2, \mathbf{x}^{s_2})\right) \\ &\wedge \left(\exists\, m \in \{1, \dots, M\},\ \ f_m(s_1, \mathbf{x}^{s_1}) < f_m(s_2, \mathbf{x}^{s_2})\right). \end{aligned} \tag{3}$$

We say that a maintenance design $(s, \mathbf{x}^s)$ is *Pareto optimal*, if and only if

$$\nexists\, (s', \mathbf{x}^{s'}) \in D \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) \succ \mathbf{f}(s, \mathbf{x}^s). \tag{4}$$

Let $D^* \subset D$ denote the set of Pareto optimal maintenance designs, where

$$D^* = \left\{ (s, \mathbf{x}^s) \in D \middle| \nexists (s', \mathbf{x}^{s'}) \in D \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) \succ \mathbf{f}(s, \mathbf{x}^s) \right\}. \tag{5}$$

Also, the *Pareto front* $\mathcal{F}^*$ is defined as a set of objective vectors of the Pareto optimal maintenance designs, with

$$\mathcal{F}^* = \left\{ \mathbf{f}(s, \mathbf{x}^s) \middle| (s, \mathbf{x}^s) \in D^* \right\}. \tag{6}$$

The set of Pareto optimal maintenance designs D^* and the Pareto front $\mathcal{F}^*$ are the solutions of the multi-objective design space exploration problem in Eq. (2).

3. Framework for multi-objective design of aircraft maintenance

The design of reliable and cost-efficient aircraft maintenance, given in Eq. (2) is a complex problem since there are potentially conflicting objectives. Also, a large number of maintenance designs $(s, \mathbf{x}^s)$ from the design space D need to be analyzed. In this section, we propose a framework to adaptively explore a wide variety of maintenance strategies and their design variables, and identify Pareto optimal aircraft maintenance designs. Fig. 1 shows the overview of our framework.

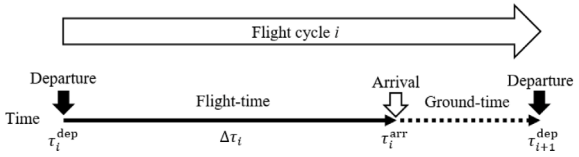

Fig. 2. Operation of an aircraft.

3.1. Aircraft maintenance model

To analyze the reliability and efficiency of an aircraft maintenance design, we first propose a generic aircraft maintenance model [17]. This model considers the operation of aircraft (aircraft arrivals and departures according to a flight schedule), the degradation process of aircraft components, and the manner in which maintenance tasks are performed to address the gradual degradation of components.

Aircraft operation

An aircraft is operated based on a sequence of flight cycles. As shown in Fig. 2, each flight cycle i is defined by a departure time τ_i^{dep}, and a flight-time $\Delta\tau_i$, where $\Delta\tau_i \sim \mathcal{N}(\overline{\Delta\tau_i}, \sigma_i^2)$. After the flight, the aircraft arrives at a destination airport at $\tau_i^{\text{arr}} = \tau_i^{\text{dep}} + \Delta\tau_i$. The time between an arrival and the next departure $[\tau_i^{\text{arr}}, \tau_{i+1}^{\text{dep}}]$ is called *ground-time*. During ground-time, maintenance tasks can be performed. If a maintenance task is not completed until the next scheduled departure time τ_{i+1}^{dep}, then the aircraft departs only after the task is completed, with a delay.

Stochastic degradation of aircraft components

In this study, we focus on the maintenance of aircraft components whose condition degrades over time due to gradual damage monotonically accumulating over time in a sequence of tiny increments. In general, various degradation processes follow such degradation processes, e.g., wear, erosion, fatigue, corrosion, crack growth, degrading health index [36].

We model such a degradation process using a Gamma process [36]. Let $Z(t)$ be the degradation level of a component at time t, which follows a Gamma process. Then, the degradation of component during flight cycle i is:

$$Z(\tau_i^{\text{arr}}) - Z(\tau_i^{\text{dep}}) \sim \text{Gamma}(\alpha, \beta), \tag{7}$$

with α the shape parameter and β the scale parameter of the Gamma distribution.

During ground-time, if no maintenance task is performed, then the degradation level of the components remains the same since the components have not been in use, i.e.,

$$Z(\tau_{i+1}^{\text{dep}}) - Z(\tau_i^{\text{arr}}) = 0. \tag{8}$$

For a brand-new component, $Z(t) = 0$. If, however, the degradation level exceeds a threshold η, i.e., $Z(t) \ge \eta$, we say that the component is *inoperable*.

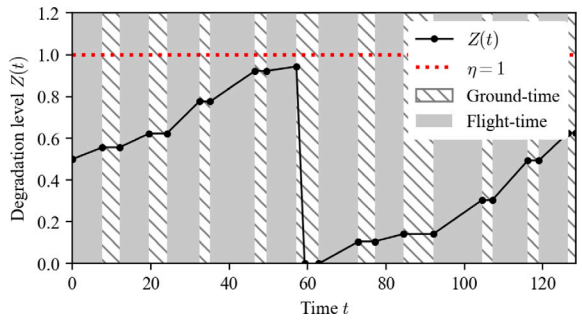

Fig. 3. A realization of the degradation of a component.

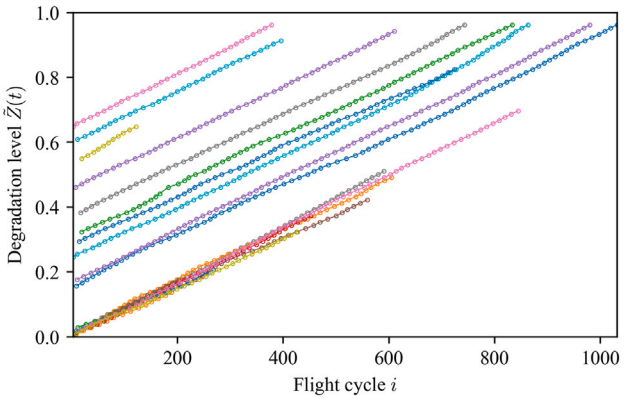

Fig. 4. Degradation level data of the aircraft brake [17].

Following Eqs. (7) and (8), $Z(t)$ becomes a piece-wise Gamma process. A realization of $Z(t)$ for 10 flight cycles is shown in Fig. 3. The degradation level increases during flight-time, and it remains the same during ground-time. During the 5th ground-time, maintenance is performed, the component is replaced with a brand-new one, and the degradation level is reset to zero.

Remark 1. In [17], we have conducted a data-driven analysis of the degradation of aircraft landing gear brakes based on actual measurements, and have shown that this degradation follows a Gamma process. Brake disks of landing gears erode during landing/take-off due to the heat and friction generated. As a result, the thickness of the brake disks reduces gradually over time. The thickness of the brake disks is a direct indicator of the degradation of the brakes. If the thickness of a brake becomes thinner than a threshold, then the brake is replaced with a new one. Fig. 4 shows the degradation level of multiple brakes, where the Y-axis is the scaled degradation level of the brake, with $Z(t) = 0$ indicating a new brake and $Z(t) = 1$ indicating that the brake needs to be replaced ($\eta = 1$). Based on this data, we apply maximum likelihood estimation (MLE) to estimate the parameters α and β of the Gamma distribution in Eq. (7). A Kolmogorov–Smirnov (KS) test verifies our hypothesis that the degradation of the brakes follows a Gamma process.

Maintenance tasks

We consider two types of maintenance tasks: component *replacements* and component *inspections* [17]. These maintenance tasks are performed based on a given maintenance design, i.e., the maintenance design specifies the type (replacement/inspection) and the time when the tasks need to be executed.

Replacements: When a component is replaced with a new one at some time t, the degradation process is reset to $Z(t) = 0$. For example, in Fig. 3, the component is replaced after flight cycle 5. The time spent for a component replacement Δt_{Rep} is modeled as an exponential distribution with mean $\overline{t_{\text{Rep}}}$, which is the average time spent for a component replacement, i.e., $\Delta t_{\text{Rep}} \sim \text{Exp}(\overline{t_{\text{Rep}}})$.

Visual inspections: When a component is visually inspected, the degradation level is observed with a certain level of error. Let $\hat{Z}(t)$ be the degradation level observed upon an inspection. Then,

$$\hat{Z}(t) = Z(t) + \epsilon_{\text{Ins}}, \tag{9}$$

where $\epsilon_{\text{Ins}} \sim \mathcal{N}(0, \sigma^2_{\text{Ins}})$ is the inspection error. The time spent for an inspection Δt_{Ins} is assumed to follow an exponential distribution with mean $\overline{t_{\text{Ins}}}$, which is the average time spent for an inspection, i.e., $\Delta t_{\text{Ins}} \sim \text{Exp}(\overline{t_{\text{Ins}}})$.

Condition monitoring using sensors: For those aircraft equipped with condition monitoring systems, sensors are used to monitor the degradation level of components. Let $\tilde{Z}(t)$ be the degradation level of a component obtained from sensors at time t. Then,

$$\tilde{Z}(t) = Z(t) + \epsilon_{\text{Sen}}, \tag{10}$$

where $\epsilon_{\text{Sen}} \sim \mathcal{N}(0, \sigma^2_{\text{Sen}})$.

Supply management of parts

In our model, for a scheduled replacement, a component is replaced with a brand-new one from the repair shop. We assume that the new component is ordered in advance for this task [37,38]. Let D_{Plan} be the time needed to order a new component, i.e., the supply lead time. Then, an order is placed at least D_{Plan} flight cycles before this scheduled task.

If, however, there is not enough time to order the new component in advance due to, for instance, an unscheduled maintenance, then a new component is leased. An *unscheduled maintenance* occurs when a severe degradation ($\hat{Z}(t) \geq \eta$) is observed during an inspection and the component needs to be promptly replaced in accordance with relevant regulations/manuals [18,19]. In this case, the operator leases a new component with an additional leasing cost and maintenance-related delay [39]. We assume that the time to lease $\Delta t_{\text{Rep,L}}$ follows an exponential distribution.

3.2. The design space of aircraft maintenance: strategy types and associated design variables

Maintenance tasks are scheduled, and executed based on a given aircraft maintenance design, which is defined by a strategy type s and the design variables $\mathbf{x}^s$ associated with this strategy type, i.e., $D = \{(s, \mathbf{x}^s) | s \in \mathcal{S}, \mathbf{x}^s \in \mathcal{X}^s\}$. In general, our framework can consider any finite number of strategy types, and any range for the design variables. We focus on three types of maintenance strategies that are often used for the maintenance of critical components: time-based maintenance (TBM) strategy types [3], condition-based maintenance (CBM) strategy types [4], and predictive maintenance (PdM) strategy types [5].

Under TBM strategies, maintenance tasks are performed at fixed time intervals. For example, an inspection of the aircraft landing gear brakes is performed every 50 flight cycles. The time interval at which an inspection is performed is the design variable of this strategy type [11,13,20]. In this example, the TBM strategy with the choice of performing an inspection every 50 flight cycles defines a maintenance design.

Under CBM strategies, the moment to perform maintenance tasks is determined based on the observed health condition of the components [23,24]. This health condition is identified either using visual inspections and/or using on-board sensors. For example, under CBM, an inspection of the aircraft landing gear brakes is performed only when on-board sensors indicate that the degradation level of the brakes exceeds a threshold of 90% degradation. In this example, the CBM strategy with the choice of performing an inspection once a 90% degradation threshold is exceeded defines a maintenance design.

Under PdM strategies, the Remaining-Useful-Life (RUL) of components is predicted using sensor data analytics. With this information, maintenance tasks are performed in anticipation of a failure [16,21].

Table 1
Summary of the maintenance strategy types (s) and their design variables ($\mathbf{x}^s$) considered in the framework.

	Maintenance strategy type (s)	Design variables (x_i^s)		Range of x_i^s
Time-based maintenance (TBM)	Fixed-interval replacement (FIR)	$x_1^{\text{FIR}} = D_{\text{Rep}}$	Interval of replacement (FC)	[1200, 1500]
	Fixed-interval inspection (FII)	$x_1^{\text{FII}} = D_{\text{Ins}}$	Interval of inspection (FC)	[20, 400]
		$x_2^{\text{FII}} = \eta_{\text{Rep}}$	Degradation threshold to replace	[0.9, 1.0]
Condition-based maintenance (CBM)	Variable-interval inspection (VII)	$x_1^{\text{VII}} = a_{\text{Ins}}$	Parameter of function $D_{\text{Ins}}(\hat{Z})$	[1, 880]
		$x_2^{\text{VII}} = b_{\text{Ins}}$	Parameter of function $D_{\text{Ins}}(\hat{Z})$	[0.9, 1.0]
		$x_3^{\text{VII}} = \eta_{\text{Rep}}$	Degradation threshold to replace	[0.9, 1.0]
	Sensor-based inspection (SBI)	$x_1^{\text{SBI}} = \eta_{\text{Ins}}$	Degradation threshold to inspect	[0.7, 0.9]
		$x_2^{\text{SBI}} = D_{\text{Ins}}$	Interval of inspection (FC)	[20, 400]
		$x_3^{\text{SBI}} = \eta_{\text{Rep}}$	Degradation threshold to replace	[0.9, 1.0]
	Sensor-based replacement (SBR)	$x_1^{\text{SBR}} = \eta_{\text{Rep}}$	Degradation threshold to replace	[0.9, 1.0]
Predictive maintenance (PdM)	Remaining-useful-life-based replacement (RBR)	$x_1^{\text{RBR}} = \rho_{\text{Rep}}$	RUL threshold to replace	[0, 50]

For example, under PdM, an aircraft brake is replaced at a moment indicated by a function of RUL. Also here, the function of RUL is the design variable of this maintenance strategy [15].

For our analysis, we consider 6 types of maintenance strategies: a fixed-interval replacement (FIR) and a fixed-interval inspection (FII) strategy, which are time-based maintenance strategies; a variable-interval inspection (VII), a sensor-based inspection (SBI), and a sensor-based replacement (SBR), which are condition-based maintenance strategies; and a Remaining-Useful-Life-based replacement (RBR) strategy, which is a predictive maintenance strategy. Thus, $S = \{\text{FIR, FII, VII, SBI, SBR, RBR}\}$. For each strategy type $s \in S$, the design variables $\mathbf{x}^s$ and their domains are summarized in Table 1. Below we discuss in detail each of the six strategy types considered.

Fixed-interval replacement (FIR)

The fixed-interval replacement (FIR) strategy replaces components at fixed time interval of D_{Rep} flight cycles. This strategy type uses neither inspection, nor condition monitoring.

The design space of FIR strategy is defined by one design variable D_{Rep}, i.e., $\mathbf{x}^{\text{FIR}} = [D_{\text{Rep}}]$. We consider the domain of D_{Rep} as $1200 \leq D_{\text{Rep}} \leq 1500$ based on the expected life cycle of aircraft landing gear brakes. The expected life cycle of a component following the Gamma process in Eq. (7) is estimated as $1/(\alpha\beta)$. In [17], we have estimated the expected life cycle of an aircraft landing gear brake to be approximately 1249 – 1446 flight cycles. Thus, a maintenance design $(\text{FIR}, [D_{\text{Rep}}])$ consists of the FIR strategy and a specific value for D_{Rep}.

Fixed-interval inspection (FII)

The fixed-interval inspection (FII) strategy relies on periodic inspections. The moment of component replacement is based on the degradation level $\hat{Z}(t)$ observed during an inspection [13,17,20]. The components are inspected every D_{Ins} flight cycles. If $\hat{Z}(t) \geq \eta_{\text{Rep}}$, then the replacement of the component is scheduled after D_{Plan} flight cycles. Here, D_{Plan} is the time required to supply required parts. In this study, we assume $D_{\text{Plan}} = 20$ flight cycles, which is the average number of flight cycles for an aircraft in 10 days [1].

The design space of FII strategy is defined by design two variables D_{Ins} and η_{Rep}, i.e., $\mathbf{x}^{\text{FII}} = [D_{\text{Ins}}, \eta_{\text{Rep}}]$. We assume that $D_{\text{Ins}} \geq D_{\text{Plan}}$ flight cycles since maintenance tasks need to be scheduled D_{Plan} flight cycles ahead of its execution. The upper bound of D_{Ins} is set to 400 flight cycles, so that we can plan to inspect a brake at least 2 times during its average life cycle (1249 – 1446 flight cycles) [17]. In the case of η_{Rep}, its maximum value is $\eta = 1$, which is a scaled degradation threshold of inoperable component (see Section 3.1), and its minimum value is 0.9 assuming a 10% safety margin for replacements. Thus, a maintenance design $(\text{FII}, [D_{\text{Ins}}, \eta_{\text{Rep}}])$ consists of the FII strategy and specific values for D_{Ins} and η_{Rep}.

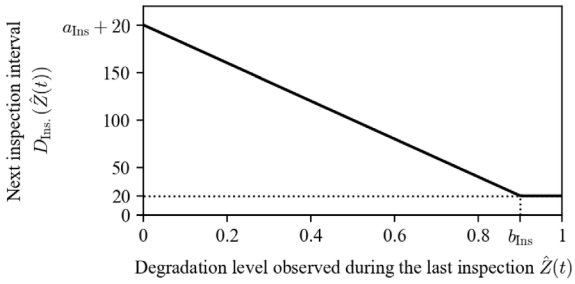

Fig. 5. The next inspection interval $D_{\text{Ins}}(\hat{Z}(t))$ is defined as a function of the last inspection result $\hat{Z}(t)$ under VII strategy [24], $a_{\text{Ins}} = 180$, $b_{\text{Ins}} = 0.9$.

Variable-interval inspection (VII)

The variable-interval inspection (VII) strategy is proposed to reduce the number of inspections when the degradation level of a component is low [24]. The moments of inspections are decided based on the degradation level $\hat{Z}(t)$ observed during the last inspection. Then, the next inspection interval $D_{\text{Ins}}(\hat{Z}(t))$ is determined as follows, [24]:

$$D_{\text{Ins}}(\hat{Z}(t)) = 20 + \max\left((a_{\text{Ins}} - \frac{a_{\text{Ins}}}{b_{\text{Ins}}}\hat{Z}(t)), 0\right), \tag{11}$$

where a_{Ins} determines the first inspection interval ($20 + a_{\text{Ins}}$ flight cycles), and b_{Ins} is the degradation threshold to perform periodic inspections with the minimum interval (20 flight cycles). Fig. 5 illustrates an example of D_{Ins}, given $\hat{Z}(t)$. When $\hat{Z}(t) \geq b_{\text{Ins}}$, inspections are scheduled at minimum interval, which we assume to be 20 flight cycles, or 10 days [1]. Upon inspection, if $\hat{Z}(t) \geq \eta_{\text{Rep}}$, the VII strategy schedules a component replacement after D_{Plan} flight cycles, similar to the FII strategy.

The design space of VII strategy is defined by three design variables, a_{Ins}, b_{Ins}, η_{Rep}, i.e., $\mathbf{x}^{\text{VII}} = [a_{\text{Ins}}, b_{\text{Ins}}, \eta_{\text{Rep}}]$. The lower bound of a_{Ins} is set to be 1 since the VII strategy with $a_{\text{Ins}} = 0$ is identical to the FII strategy with $D_{\text{Ins}} = 20$ flight cycles. The upper bound of a_{Ins} is set to be 880, which renders the VII strategy to perform the first inspection after 1000 flight cycles from its replacement. This is lower than the expected life cycle of aircraft landing gear brakes (1249–1446 flight cycles [17]), and ensures that at least one inspection takes place before a brake becomes inoperable. For b_{Ins} and η_{Rep}, we explore the range $[0.9, 1]$ considering a maximum 10% safety margin for inspections and replacements. Thus, a maintenance design $(\text{VII}, [a_{\text{Ins}}, b_{\text{Ins}}, \eta_{\text{Rep}}])$ consists of the FII strategy and specific values for a_{Ins}, b_{Ins}, and η_{Rep}.

Sensor-based inspection (SBI)

The sensor-based inspection (SBI) strategy uses sensor monitoring data $\tilde{Z}(t)$ to substitute a part of the inspections by sensor monitoring and reduce the number of inspections [17,23]. Unlike FII and FIR strategies, SBI strategy starts the periodic inspections based on the sensor data $\tilde{Z}(t)$ obtained after every flight cycles. If the sensor data

is below a threshold η_{Ins}, i.e., $\tilde{Z}(t) < \eta_{\text{Ins}}$, then inspections are skipped. If $\tilde{Z}(t) \geq \eta_{\text{Ins}}$, a periodic inspection of fixed interval D_{Ins} is started. Upon inspection, if $\hat{Z}(t) \geq \eta_{\text{Rep}}$, the component is replaced after D_{Plan} flight cycles.

The design space of SBI strategy is defined by three design variables η_{Ins}, D_{Ins}, and η_{Rep}, i.e., $\mathbf{x}^{\text{SBI}} = [\eta_{\text{Ins}}, D_{\text{Ins}}, \eta_{\text{Rep}}]$. For η_{Ins}, we explore the range $[0.7, 0.9]$, so that periodic inspections start at a degradation level having a 10%–30% safety margin from the threshold of inoperable components ($\eta = 1$). Also, η_{Rep} has a range $[0.9, 1.0]$ considering a maximum 10% safety margin. For D_{Ins}, the range $[20, 400]$ is considered, similar to the FII strategy. Thus, a maintenance design $(\text{SBI}, [\eta_{\text{Ins}}, D_{\text{Ins}}, \eta_{\text{Rep}}])$ consists of the SBI strategy and specific values for η_{Ins}, D_{Ins}, and η_{Rep}.

Sensor-based replacement (SBR)

The sensor-based replacement (SBR) strategy determines the moment for component replacement based on the last sensor monitoring data $\tilde{Z}(t)$. So, there are no visual inspections performed by mechanics under SBR strategy [17]. A component is replaced if the sensor indicates a degradation level $\tilde{Z}(t)$ higher than a threshold, i.e., if $\tilde{Z}(t) \geq \eta_{\text{Rep}}$. Since we assume imperfect measurements $\tilde{Z}(t)$ with measurement error ϵ_{Sen} (see Eq. (10)), an early replacement may be triggered although the true degradation level $Z(t)$ is below a threshold, or a required replacement may be missed even though the true degradation level $Z(t)$ exceeds a threshold.

The design space of SBR strategy is defined by one design variable η_{Rep}, i.e., $\mathbf{x}^{\text{SBR}} = [\eta_{\text{Rep}}]$. For this, we explore the range $[0.9, 1]$, considering a maximum 10% safety margin from the degradation threshold of an inoperable component ($\eta = 1$). Thus, a maintenance design $(\text{SBR}, [\eta_{\text{Rep}}])$ consists of the SBR strategy and a specific value for η_{Rep}.

Remaining-useful-life-based replacement (RBR)

The Remaining-Useful-Life-based replacement (RBR) strategy is a predictive maintenance design that uses prognostics of the Remaining-Useful-Life (RUL) of components to schedule replacements [16,17]. The RUL prognostics are determined based on an analysis of the data collected by sensors on the degradation of the components [5,6]. The RUL of a component at time t is estimated based on the last available sensor data $\{\tilde{Z}(t') \text{ for } 0 \leq t' \leq t\}$. The following linear model is considered to estimate the degradation level of a component at time $t + \Delta t$ [17]:

$$\tilde{Z}(t + \Delta t) = c_0 + c_1 \Delta t. \tag{12}$$

The coefficients c_0 and c_1 are updated after every flight cycle based on the most recent sensor data using the ordinary least square method [40]. After each flight cycle, the RUL of the component at time t is estimated as follows [17]:

$$\min\{\Delta t | c_0 + c_1 \Delta t \geq \eta\} \tag{13}$$

Finally, if RUL is below a threshold ρ_{Rep}, the component is replaced after D_{Plan} flight cycles.

The design space of the RBR strategy is defined by one design variable ρ_{Rep}, i.e., $\mathbf{x}^{\text{RBR}} = [\rho_{\text{Rep}}]$. For the lower bound, we consider $\rho_{\text{Rep}} \geq 0$. This ensures that we set a non-negative RUL as a threshold to perform predictive maintenance. The upper bound of ρ_{Rep} is set to be 50 flight cycles, considering that a replacement is performed after $D_{\text{Plan}} = 20$ flight cycles from the moment when RUL is predicted to be below the threshold ρ_{Rep}, i.e., 30 flight cycles are considered as a safety margin. Here, 30 flight cycles are twice the standard deviation of the error of RUL prediction following Eq. (13). Thus, a maintenance design $(\text{RBR}, [\rho_{\text{Rep}}])$ consists of the RBR strategy and a specific value for ρ_{Rep}.

3.3. Multiple objectives of aircraft maintenance

We consider the following objectives that have been regarded in literature as key performance indicators for aircraft maintenance [12,13,17,41,42].

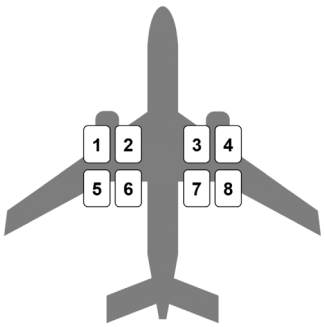

Fig. 6. An example of multi-component system of aircraft, 8 brakes of wide-body aircraft [17].

*Maximization of mean-cycles-to-replacement (*MCTR*)*

Maximizing the utilization time of components is of high interest in aircraft maintenance [12]. The utilization of components is evaluated in terms of the mean-cycles-to-replacement (MCTR), which is defined as the mean number of flight cycles that a component is utilized for before it is replaced. Since a large part of the maintenance cost comes from component replacements, a high MCTR implies that we are exploiting the component longer and that the cost per unit time is reduced [12].

Minimization of the expected number of maintenance tasks (N_{Rep}, N_{Ins})

The cost with maintenance is often evaluated as the number of maintenance tasks performed times the cost of individual maintenance tasks [12,13]. However, the cost of individual aircraft maintenance tasks is specific to each operator [1], and depends on various factors such as the skill of the mechanics that execute the task, the moment when the task is executed, etc. This makes the estimation of the costs for a specific task challenging. In this study, we assume that the maintenance cost is represented by the number of maintenance tasks performed. The goal is to minimize the number of component replacements (N_{Rep}) and component inspections (N_{Ins}).

Minimization of the expected number of unscheduled replacements (N_{Uns})

Unscheduled replacements have a negative impact both on the supply management of parts and on the flight schedule of the aircraft (see Section 3.1). For example, when there is not enough time to supply a new component to replace, this may need to be leased at a higher cost [39]. Also, the task will be performed with delay due to additional time to acquire the leased part [17,41]. The aim is to minimize the expected number of unscheduled replacements (N_{Uns}).

Minimization of the expected number of degradation incidents (N_{Inc})

As an indicator of the reliability of the aircraft maintenance strategy, we consider the expected number of *degradation incidents* (N_{Inc}) [17]. We consider a multi-component system having a k-out-of-n redundancy, i.e., the system consists of n components and requires that at least k components are operable ($0 < k \leq n$). If a system with k-out-of-n redundancy has more than $(n - k)$ inoperable components, then we say that a *degradation incident* occurs. When a degradation incident occurs, the aircraft needs prompt maintenance before it can perform a next flight.

Aircraft landing gear brakes are an example of a k-out-of-n system. Wide-body aircraft are equipped with 8 brakes, 4 on each side, as shown in Fig. 6. According to the minimum-equipment-list (MEL) [18,19], a minimum of 3-out-of-4 brakes on each side need to be operational for the aircraft to be able to fly.

Table 2
Spearman's rank correlation coefficients between pairs of objectives.

	Group-1			Group-2		
	Cost-related objectives			Reliability-related objectives		
	$-$MCTR	N_{Rep}	N_{Ins}	N_{Uns}	N_{Inc}	T_{Delay}
MCTR		0.994	0.373	−0.921	−0.939	−0.907
N_{Rep}	0.994		0.371	−0.919	−0.939	−0.907
N_{Ins}	0.373	0.371		−0.191	−0.532	−0.210
N_{Uns}	−0.921	−0.919	−0.191		0.876	0.982
N_{Inc}	−0.939	−0.939	−0.532	0.876		0.877
T_{Delay}	−0.907	−0.907	−0.210	0.982	0.877	

Minimization of delay due to maintenance (T_{Delay})

Delays due to maintenance are another key performance indicator for maintenance strategies [42]. Delays due to maintenance occur when the time spent to perform tasks (Δt_{Rep}, and Δt_{Ins}) exceeds the scheduled ground-time (see also Fig. 2 for the definition of ground-time), or when a new component needs to be leased and an additional time $\Delta t_{\text{Rep,L}}$ is required to supply this component. The aim is to minimize delays due to maintenance.

3.4. Crude Monte Carlo simulation of maintenance designs and selection of conflicting objectives

In this section, we perform crude Monte Carlo simulation to determine Pareto optimal maintenance designs and analyze the correlation of multiple design objectives.

Crude Monte Carlo simulation of maintenance designs

Using the aircraft maintenance model in Section 3.1, the strategy types presented in Table 1 are simulated by means of Monte Carlo simulation. Although the design variables of each strategy type are defined on continuous ranges, we sample discrete values for these variables using a l-level factorial design (FD) method [43], with $l = 7$. Using a 7-level FD, the total number of maintenance designs to be simulated is $\sum_{s\in S} 7^{N^s}$, where N^s is the number of the design variables of strategy type s. For our analysis, we considered 3 maintenance strategies with one design variable (FIR, SBR, RBR), 1 maintenance strategy with two design variables (FII) and 2 maintenance strategies with three design variables (VII, SBI), see Table 1. This leads to a total of $3 \cdot 7^1 + 7^2 + 2 \cdot 7^3 = 756$ maintenance designs that are simulated.

For each sampled maintenance design, the objectives are evaluated by means of crude Monte Carlo simulation. We simulate each maintenance design $N_{\text{MC}} = 1000$ times, and evaluate the mean values of the objectives observed during the simulations. Each design is simulated for a period of 10 years of aircraft maintenance.

Analyzing the correlation of the design objectives

Following simulation, we analyze the 6 objectives (MCTR, N_{Rep}, N_{Ins}, N_{Inc}, N_{Uns}, and T_{Delay}), of 756 maintenance designs sampled from different strategy types. Table 2 shows the Spearman's rank correlation coefficient [44] of these objectives.

Based on the Spearman's coefficients in Table 2, we group the objectives that are positively correlated: $-$MCTR, N_{Rep}, and N_{Ins} (Group 1), and negatively correlated: N_{Uns}, N_{Inc}, and T_{Delay} (Group 2). Group 1 consists of cost-related objectives since the maintenance cost is reduced as we use components longer (high MCTR), and as we perform fewer tasks (low N_{Rep} and N_{Ins}). Group 2 consists of reliability-related objectives, where we aim to minimize the number of unscheduled tasks, degradation incidents, and delays. The cost-related and reliability-related groups of objectives are conflicting.

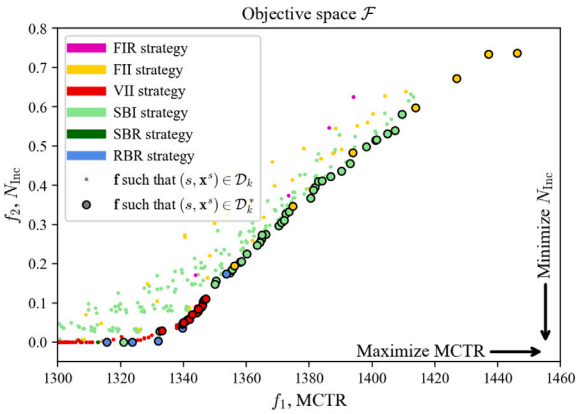

Fig. 7. Pareto optimal aircraft maintenance designs using crude Monte Carlo simulation.

Selection of conflicting objectives

We are interested in maintenance designs that balance cost and reliability objectives. As such, from each group of objectives in Table 2 we select objectives: MCTR and N_{Inc}, i.e.,

maximize f_1=MCTR (cost-related objective)

minimize $f_2 = N_{\text{Inc}}$ (reliability-related objective)

In general, our proposed framework is not limited to only these two objectives, and can be readily applicable for any set of objectives.

Pareto front generated using crude Monte Carlo simulation

Fig. 7 shows the Pareto optimal maintenance designs obtained using crude Monte Carlo simulation of the 756 maintenance designs sampled for the 2 conflicting objectives $f_1 = \text{MCTR}$ and $f_2 = N_{\text{Inc}}$.

However, the Pareto front in Fig. 7 is not generated efficiently. First, this Pareto front is obtained after sampling only a small number of discrete values of the design variables, following 7-level FD method. In general, we are interested in evaluating the entire, continuous range of the design variables in Table 1. Moreover, most of the 756 maintenance designs that have been simulated are dominated. In fact, only 66 out of 756 maintenance designs are actually Pareto optimal. This implies that most of the computational power is wasted to simulate maintenance designs that are dominated. Therefore, we need an efficient algorithm to obtain Pareto optimal maintenance designs by considering the continuous ranges of design variables, and by adaptively simulating those maintenance designs expected to be Pareto optimal.

3.5. An adaptive algorithm for multi-objective design space exploration of aircraft maintenance

In this section, we propose an algorithm to **E**xplore the design space D of the multi-objective aircraft maintenance problem using Gaussian process **L**earning and adaptive **SA**mpling (ELSA). The main merit of ELSA is that it adaptively samples the maintenance designs that are expected to improve the Pareto front of the aircraft maintenance problem, and simulates only these sampled maintenance designs. With this, the total number of simulations that need to be conducted is reduced significantly.

ELSA utilizes Gaussian process (GP) learning models as a surrogate model of the objectives. The GP model is a flexible non-parametric model that does not need prior knowledge on the objectives [30,31]. Moreover, the GP model not only provides predictions, but also estimates the uncertainty of the predictions, which is used in the adaptive sampling step to prevent premature convergence of the design space exploration.

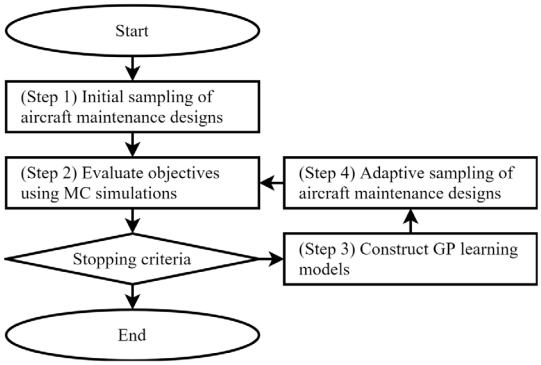

Fig. 8. Overview of ELSA.

Fig. 8 shows an overview of ELSA. We first sample initial maintenance designs $(s, \mathbf{x}^s)$ (Step 1). Iteratively, we evaluate the objective vectors $\mathbf{f}(s, \mathbf{x}^s)$ of the sampled maintenance designs using MC simulation, and update the Pareto front of the aircraft maintenance problem (Step 2). Gaussian process (GP) learning models are constructed in Step 3, based on the information acquired in the previous steps. Next, we adaptively sample new maintenance designs that are expected to be Pareto optimal, using the GP learning models (Step 4). These newly obtained maintenance designs are then simulated (back to Step 2). Steps 2–4 are iterated until some stopping criteria are satisfied. Below we discuss in detail these steps .

Step 1: Initial sampling of maintenance designs

Initial maintenance designs $(s, \mathbf{x}^s)$ are sampled as follows. For each strategy type $s \in \mathcal{S}$, design variables $\mathbf{x}^s$ are sampled from its domain $\mathcal{X}^s$ using an l-level factorial design (FD) method [43]. The FD method initializes ELSA without bias, providing evenly distributed data points over the domain $\mathcal{X}^s$ of the design variables [43].

Step 2: Evaluate the objectives of maintenance designs, update pareto front and training data

In Step 2, we evaluate the objectives $\mathbf{f}(s, \mathbf{x}^s)$ of the sampled maintenance designs (see Step 1) using MC simulations of the aircraft maintenance model in Section 3.1. Let $\mathcal{D}_k$ denote the set of maintenance designs whose objectives have been evaluated during iterations $0, 1, \dots, k$, with $\mathcal{D}_k \subset \mathcal{D}$. Let $\mathcal{F}_k = \{\mathbf{f}(s, \mathbf{x}^s) | (s, \mathbf{x}^s) \in \mathcal{D}_k\}$ denote the set of objective vectors that have been evaluated during iterations $0, 1, \dots, k$.

Having obtained $\mathcal{F}_k$ for the set of evaluated maintenance designs $\mathcal{D}_k$, ELSA identifies those maintenance designs that are Pareto optimal $\mathcal{D}_k^*$. Thus, the following Pareto front $\mathcal{F}_k^*$ is obtained:

$$\mathcal{D}_k^* = \left\{ (s, \mathbf{x}^s) \in \mathcal{D}_k \middle| \nexists (s', \mathbf{x}^{s'}) \in \mathcal{D}_k \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) \succ \mathbf{f}(s, \mathbf{x}^s) \right\}, \tag{14}$$

$$\mathcal{F}_k^* = \{\mathbf{f}(s, \mathbf{x}^s) | (s, \mathbf{x}^s) \in \mathcal{D}_k^*\}. \tag{15}$$

Here, $\mathcal{D}_k^*$ and $\mathcal{F}_k^*$ are approximations of the true Pareto optimal designs $\mathcal{D}^*$ and the true Pareto front $\mathcal{F}^*$, as defined in Eq. (5)–(6). By exploring additional maintenance designs $(s, \mathbf{x}^s) \in (\mathcal{D} - \mathcal{D}_k)$ in further iterations, ELSA refines $\mathcal{D}_k^*$ and $\mathcal{F}_k^*$.

Also, these sampled maintenance designs $\mathcal{D}_k$ and their objective vectors $\mathcal{F}_k$ will be used as training data for Gaussian learning models in Step 3.

Step 3: Construct Gaussian process learning models

In Step 3, we construct Gaussian process (GP) learning models using the Monte Carlo simulation results obtained during iterations $0, 1, \dots, k$, i.e., $\mathcal{D}_k$ and $\mathcal{F}_k$. These GP models are surrogate models that pre-estimate the objective vector of maintenance designs that have not yet been evaluated using MC simulation of the aircraft maintenance model. This pre-estimation is faster than using MC simulations. Therefore, this pre-estimation is further used in Step 4 of ELSA. Below we explain in detail how we construct the GP models.

We construct a GP model $\mathcal{GP}_m^s$ for each strategy type s, and for each objective f_m. This $\mathcal{GP}_m^s$ model assumes that the objective f_m of maintenance design $(s, \mathbf{x}^s)$, follows a Gaussian process specified by its mean function and covariance function [30,31]. Assuming a zero prior mean function, $\mathcal{GP}_m^s$ is defined as:

$$f_m(s, \mathbf{x}^s) \sim \mathcal{GP}_m^s\left(0, \kappa_m^s(\mathbf{x}^s, \mathbf{x}'^s)\right), \tag{16}$$

where κ_m^s is the covariance function, or equivalently a *kernel*.

At iteration k of ELSA, the training data for $\mathcal{GP}_m^s$ consist of X_k^s and $F_{m,k}^s$, where X_k^s is the matrix whose rows are $\mathbf{x}^s$ such that $(s, \mathbf{x}^s) \in \mathcal{D}_k$, and $F_{m,k}^s$ is the vector whose elements are $f_m(s, \mathbf{x}^s)$ such that $(s, \mathbf{x}^s) \in \mathcal{D}_k$. For simplicity, in this section we drop the superscript s and the subscript m and k since the following discussion applies to all strategy types $s \in \mathcal{S}$, objectives $m \in \{1, \dots, M\}$, and iterations $k = \{0, 1, \dots\}$.

Having the GP model specified in Eq. (16), the prior distribution of training data F and a test output f at a test input $\mathbf{x}$ is:

$$\begin{bmatrix} F \\ f \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) & K(X,\mathbf{x}) \\ K(\mathbf{x},X) & K(\mathbf{x},\mathbf{x}) \end{bmatrix}\right), \tag{17}$$

where $K(\cdot,\cdot)$ is the covariance matrix calculated by kernel $\kappa(\cdot,\cdot)$. Then, the posterior distribution of f is [30]:

$$f | X, F, \mathbf{x} \sim \mathcal{N}\left(\mathbb{E}[f], \mathbb{V}[f]\right). \tag{18}$$

Here, the mean $\mathbb{E}[f]$ and variance $\mathbb{V}[f]$ of the posterior distribution are:

$$\mathbb{E}[f] = K(\mathbf{x}, X)\left[K(X,X)\right]^{-1} F, \tag{19}$$

$$\mathbb{V}[f] = K(\mathbf{x},\mathbf{x}) - K(\mathbf{x},X)\left[K(X,X)\right]^{-1} K(X,\mathbf{x}). \tag{20}$$

The kernel should be defined based on the characteristics of the considered problem [30]. We consider the following two characteristics of aircraft maintenance designs. First, the objective values of two maintenance designs with similar $\mathbf{x}$ are correlated, i.e., radial basis correlation. Second, we assume that the training data F contains uncertainty as they are evaluated by MC simulation of the stochastic aircraft maintenance model. Based on these two characteristics, we consider the following compound kernel function:

$$\kappa(\mathbf{x}, \mathbf{x}') = \kappa_{\text{RBF}}(\mathbf{x}, \mathbf{x}') + \kappa_{\text{WN}}(\mathbf{x}, \mathbf{x}'), \tag{21}$$

where κ_{RBF} is a squared exponential radial basis function (RBF) kernel, and κ_{WN} is a white noise (WN) kernel [30].

The RBF kernel κ_{RBF} models the correlation of two vectors of the design variables $\mathbf{x}$ and $\mathbf{x}'$ based on an Euclidean distance as follows:

$$\kappa_{\text{RBF}}(\mathbf{x}, \mathbf{x}') = \sigma_{\text{RBF}}^2 \exp\left(-\frac{1}{2}\sum_{j=1}^{N}\left(\frac{x_j - x_j'}{l_j}\right)^2\right), \tag{22}$$

where $\mathbf{x} = [x_1, \dots, x_j, \dots, x_N]$, l_j is a characteristic length-scale, and $\sigma_{\text{RBF}}^2 > 0$ is a scale parameter of RBF kernel. Depending on l_j, the intensity of the correlation along design variable x_j varies, i.e, κ_{RBF} is an anisotropic kernel. This feature allows us to model the situation when some design variables x_j have a larger impact on the objectives than other design variables [30]. Such an anisotropic correlation is often observed in the simulation of aircraft maintenance [17].

The WN kernel κ_{WN} models the homogeneous noise in the objective values in the training data F [32]. Formally, κ_{WN} is defined as follows:

$$\kappa_{\text{WN}}(\mathbf{x}, \mathbf{x}') = \begin{cases} \sigma_{\text{WN}}^2, & \text{if } \mathbf{x} = \mathbf{x}' \\ 0, & \text{otherwise,} \end{cases} \tag{23}$$

where $\sigma_{\text{WN}}^2 > 0$ is a noise level.

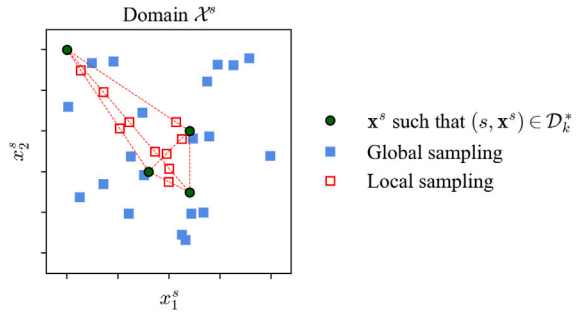

Fig. 9. An example of global and local sampling in a 2-dimensional domain $\mathcal{X}^s$ of strategy type s.

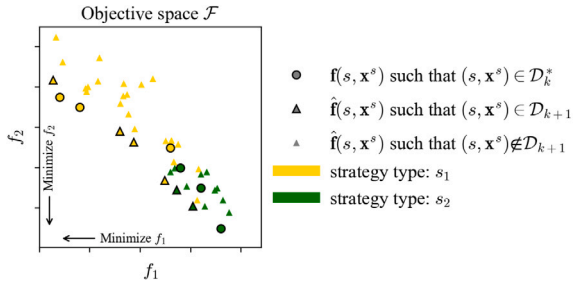

Fig. 10. An example of an objective space $\mathcal{F}$ during Step 4 of ELSA.

In Eq. (22)–(23), l_j, σ^2_{RBF} and σ^2_{WN} are hyper-parameters, and are optimized using the maximum likelihood estimation [40].

Finally, we train the GP models $\mathcal{GP}^s_m$ for all strategy types $s \in S$ and all objectives $m \in \{1, \dots, M\}$. These GP models are further used to rapidly pre-estimate the objective vectors in Step 4 (adaptive sampling) of ELSA.

Step 4: Adaptive sampling of maintenance designs

In Step 4, we select new maintenance designs that can potentially improve the Pareto front $\mathcal{F}^*_k$ obtained in Step 2, using the GP models constructed in Step 3. These newly selected maintenance designs will be simulated in a next iteration $(k+1)$. In general, this selection of new points (maintenance designs) to explore is done by solving infill-criteria maximization problems with genetic algorithms [32]. However, these infill-criteria maximization problems are often hard to solve because of many local maximums and a high computational cost of the infill-criteria [32,34]. Moreover, this approach rarely identifies new designs that are Pareto optimal in some applications [45]. To address these issues, we propose a novel approach to select new maintenance designs using *adaptive sampling* as follows.

We first randomly sample $\mathbf{x}^s \in \mathcal{X}^s$ for all $s \in S$ based on two approaches: (i) some are sampled from the entire domain $\mathcal{X}^s$ uniformly at random (global sampling), and (ii) the others are sampled near the already available Pareto optimal $\mathbf{x}^s$ (local sampling). Global sampling contributes to the *exploration* for new designs, while local sampling *exploits* the current Pareto optimal solutions to generate additional Pareto optimal maintenance designs.

For global sampling, ELSA chooses $\mathbf{x}^s$ independently of the training data obtained up to the current iteration (exploration). Specifically, we sample n_G vectors of the design variables $\mathbf{x}^s$ from $\mathcal{X}^s$, uniformly at random. Using global sampling, ELSA explores the entire domain $\mathcal{X}^s$ even when the Pareto optimal solutions are clustered in a small area, as seen in Fig. 9.

Local sampling is based on the idea that a Pareto optimal design variable $\mathbf{x}^s$ is likely to be located in the vicinity of other Pareto optimal design variables (exploitation). We sample two vectors of design variables $\mathbf{x}^s_1$ and $\mathbf{x}^s_2$ of strategy type s, and consider a random weight w, such that

$$(s, \mathbf{x}^s_1) \in D^*_k,\ (s, \mathbf{x}^s_2) \in D^*_k,\ 0 < w < 1. \tag{24}$$

Then, the convex combination of $\mathbf{x}^s_1$ and $\mathbf{x}^s_2$ with weight w is sampled as:

$$\mathbf{x}^s = w\mathbf{x}^s_1 + (1-w)\mathbf{x}^s_2. \tag{25}$$

Following this approach, we sample n_L vectors of design variables $\mathbf{x}^s$ from $\mathcal{X}^s$. Fig. 9 shows an example of global and local sampling in a 2-dimensional domain $\mathcal{X}^s$.

For all the maintenance designs that have been sampled using global/local sampling, we rapidly pre-estimate their objective vectors $\hat{\mathbf{f}}(s, \mathbf{x}^s)$ using the GP models discussed in Step 3. Here, since this pre-estimation using GP models is much faster than using MC simulation, ELSA can pre-estimate many more design points sampled by global and local sampling. Let $\hat{\mathbf{f}}(s, \mathbf{x}^s) = [\hat{f}_1(s, \mathbf{x}^s), \dots, \hat{f}_M(s, \mathbf{x}^s)]$ be the objective vector pre-estimated by the GP models, where $\hat{f}_m(s, \mathbf{x}^s)$, $1 \le m \le M$, is the objective value pre-estimated by the GP model $\mathcal{GP}^s_m$ constructed in Step 3. For this pre-estimation, ELSA considers both the mean $\mathbb{E}[f_m]$ and the variance $\mathbb{V}[f_m]$ of the GP models' posterior distribution (see Eq. (18)–(20)). Assuming that we want to minimize f_m, we pre-estimate $\hat{f}_m$ as the lower-limit of the confidence interval of the prediction of f_m made by the GP model $\mathcal{GP}^s_m$, i.e.,

$$\hat{f}_m(s, \mathbf{x}^s) \simeq \mathbb{E}[f_m(s, \mathbf{x}^s)] - \zeta \sqrt{\mathbb{V}[f_m(s, \mathbf{x}^s)]}, \tag{26}$$

where $\zeta \ge 0$ is the width of the confidence interval. Since the lower limit of this confidence interval is used, the pre-estimation $\hat{f}_m$ is under-estimated relative to the mean $\mathbb{E}[f_m]$. The larger $\mathbb{V}[f_m]$ is, the larger the under-estimation is. Conversely, if we want to maximize f_m, then the upper-limit of this confidence interval is used. In this case, the pre-estimation $\hat{f}_m$ is over-estimated.

Next, based on these pre-estimated objective vectors $\hat{\mathbf{f}}$, ELSA selects only those maintenance designs that have been sampled, and that are not dominated by the currently available Pareto optimal maintenance designs. These selected maintenance designs will be simulated in the next iteration $(k+1)$, i.e., they are added to D_{k+1}. Formally, a sampled maintenance design $(s, \mathbf{x}^s)$ is selected and added to D_{k+1}, when:

$$\begin{aligned} &(s, \mathbf{x}^s) \in D_{k+1} \Longleftrightarrow \\ &\left(\nexists (s', \mathbf{x}^{s'}) \in D^*_k \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) \succ \hat{\mathbf{f}}(s, \mathbf{x}^s)\right) \wedge \\ &\left(\nexists (s', \mathbf{x}^{s'}) \in D_{k+1} \text{ such that } \hat{\mathbf{f}}(s', \mathbf{x}^{s'}) \succ \hat{\mathbf{f}}(s, \mathbf{x}^s)\right). \end{aligned} \tag{27}$$

Fig. 10 shows an example of the adaptive sampling step when considering the objective space $\mathcal{F}$. Circle-points represent the Pareto optimal objective vectors obtained during iterations $0, 1, \dots, k$. Triangle-points represent the pre-estimated objective vectors $\hat{\mathbf{f}}(s, \mathbf{x}^s)$ of the sampled maintenance designs. Among them, those maintenance designs that are *not* dominated by the current Pareto optimal designs (circle-points) nor by the other designs sampled at this step (triangle-points), are selected for simulation in the next iteration $(k+1)$, i.e., $(s, \mathbf{x}^s) \in D_{k+1}$.

This adaptive sampling step enables ELSA to balance between exploration and exploitation. Here, *exploration* refers to acquiring more training data and thus reducing the uncertainty of the GP models. To explore, ELSA selects maintenance designs that have a high uncertainty $\mathbb{V}[f_m]$. These maintenance designs are often located far from the already evaluated maintenance designs. *Exploitation* refers to improving the current solutions D^*_k and $\mathcal{F}^*_k$ using the available training data set [46]. To exploit, ELSA selects maintenance designs that are expected to dominate the current Pareto optimal maintenance designs. During early iterations when the training data set is limited, the GP models have a high uncertainty $\mathbb{V}[f_m]$. Thus, the pre-estimation $\hat{f}_m$ (see Eq. (26)) is influenced mainly by the uncertainty term $\mathbb{V}[f_m]$ and less by

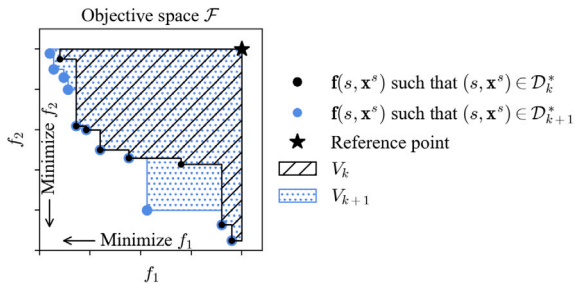

Fig. 11. A visualization of hyper-volume indicator V at iteration k and iteration $(k+1)$.

the mean $\mathbb{E}[f_m]$. Therefore, in these early iterations, those maintenance designs with a high uncertainty are most likely to be chosen. As the amount of training data increases, the uncertainty of the GP models decreases, and thus the pre-estimation $\hat{f}_m$ is influenced mainly by the mean $\mathbb{E}[f_m]$. If we assume that f_m is minimized(maximized), then those maintenance designs with a small(large) mean $\mathbb{E}[f_m]$ are most likely to be chosen. As such, taking into account the level of the uncertainty of the GP models, ELSA balances between exploration and exploitation.

Stopping criteria and quality indicators

At the end of each iteration, ELSA is terminated if one of the following stopping criteria is satisfied. First, we consider the computational cost, i.e., ELSA is terminated when a predefined computational time is exceeded, or when a predefined number of simulations is exceeded. Second, ELSA is terminated when the quality of the solution is satisfactory or converges. Here, we consider the following two quality indicators for ELSA:

(i) Hyper-volume indicator

The *hyper-volume indicator* V_k is the hyper-volume in the objective space $\mathcal{F}$ covered by a reference point and the available Pareto front [47,48]. Fig. 11 provides a visualization of the hyper-volume indicators V_k and V_{k+1} for a 2-dimensional space $\mathcal{F}$ of objectives. The black/blue circle-points denote the Pareto optimal objective vectors obtained after iteration k and $(k+1)$, respectively. The star-point in the upper-right corner is the reference point. The hyper-volume is monotonically increasing as the number of iterations increases, i.e., $V_k \leq V_{k+1}$, because new Pareto optimal objective vectors always increase the hyper-volume [47]. Also V_k is bounded from above by the hyper-volume indicator of the true Pareto front V_∞, i.e., $V_k \leq V_\infty$ [48]. Thus, V_k monotonically converges to V_∞ as the approximated Pareto front at iteration k ($\mathcal{F}_k^*$) approaches the true Pareto front ($\mathcal{F}^*$).

(ii) The number of Pareto optimal maintenance designs

We count the *number of Pareto optimal maintenance designs* obtained after k iterations, i.e., $|D_k^*|$. A large $|D_k^*|$ implies that the generated Pareto front is densely populated. Unlike V_k, $|D_k^*|$ is not monotonically increasing. For example, when a newly identified Pareto optimal maintenance design dominates many solutions of the previous Pareto front, then $|D_k^*|$ will actually decrease from iteration k to iteration $(k+1)$.

We say that V_k represents the degree of *exploration* of the domains, while $|D_k^*|$ represents the degree of *exploitation*. This is because V_k is sensitive to a new objective vector far from the current Pareto front, while $|D_k^*|$ can be increased by a large number of objective vectors close to the current Pareto front. For example, in the upper left corner of Fig. 9, four new solutions are found, but these do not significantly increase V_k compared to the increment made by a single new solution in the lower right corner of Fig. 11.

Table 3
Parameter values of Gamma process for the aircraft brake degradation model [17]. L and R indicate whether the brake is on the left or right side of the wing.

Brake location index	Side	α	β
1	L	3.350	2.063e−4
2	L	4.146	1.836e−4
3	R	3.546	2.217e−4
4	R	3.390	2.171e−4
5	L	4.667	1.715e−4
6	L	4.100	1.856e−4
7	R	3.068	2.329e−4
8	R	2.583	2.852e−4

4. Case study: Designing maintenance for landing gear brakes

In this section, we apply our proposed framework to explore the entire design space of aircraft maintenance, i.e., considering both TBM, CBM and predictive maintenance designs, for landing gear brakes.

4.1. Model parameters

Aircraft landing gear brakes are a k-out-of-n multi-component system. A wide-body aircraft has 8 landing gear brakes, 4 on each side of the wings (see Fig. 6). 3 out of 4 on each side need to be operable during flights to satisfy the manuals/regulations [18,19]. After each take-off and landing, the brake disks degrade, i.e., the thickness of the brake disks reduces. In [17], it is shown that this degradation process follows a Gamma process, and the parameters α and β are estimated using the maximum likelihood estimation (MLE) based on the degradation data of landing gear brakes of a fleet of aircraft (see Table 3). As soon as the thickness of a brake reduces to half of its original thickness, i.e, as soon as the degradation level exceeds a threshold $\eta = 1$, it is required to replace this brake. In this case, the brakes do not completely lose their functionality, but they are required to be replaced.

As maintenance tasks for landing gear brakes, we consider replacements, visual inspections, and condition monitoring using sensors (see Section 3.1). Following interviews with maintenance experts, we assume that the mean time spent for a brake replacement is 3 h ($\overline{t_{\text{Rep}}} = 3\text{hrs}$). For unscheduled maintenance, we assume an additional time $\overline{t_{\text{Rep,L}}} = 6\text{hrs}$ is needed to supply the required component. A visual inspection requires 2 min on average ($\overline{t_{\text{Ins}}} = 2\text{min}$), with $\sigma_{\text{Ins}} = 0.0075$ [17]. For the condition monitoring systems, the sensor error σ_{Sen} is assumed to be 0.0204 [17].

4.2. Pareto front of aircraft maintenance designs

We consider the maximization of the mean-cycles-to-replacements, i.e., max f_1 = MCTR, which is a cost-related objective. For the reliability-related objective, we minimize the expected number of degradation incidents of multi-component systems with k-out-of-n redundancy, i.e., min $f_2 = N_{\text{Inc}}$. The explored design space consists of 6 maintenance strategies (see Section 3.2). The ranges of their design variables are shown in Table 1.

Using ELSA, 195 Pareto optimal designs are identified after simulating 1035 maintenance designs during 19 iterations, i.e., $|D_{19}^*| = 195$, and $|D_{19}| = 1035$. The process of iteratively generating Pareto optimal maintenance designs using ELSA is shown in Fig. 12. Each point in Fig. 12 represents the objective vector $\mathbf{f}(s,\mathbf{x}^s)$ of one maintenance design $(s,\mathbf{x}^s) \in D_k$, where s is the strategy type and $\mathbf{x}^s$ are the associated design variables. The large circles represent the Pareto optimal maintenance designs ($(s,\mathbf{x}^s) \in D_k^*$), while the small dots represent the dominated maintenance designs. The cross-points correspond to the maintenance designs adaptively sampled for the next iteration $k+1$, with their objective vectors being pre-estimated using the GP models.

At the initial iteration $k = 0$, a sparse Pareto front is generated, which consists of 19 Pareto optimal maintenance designs. This front is

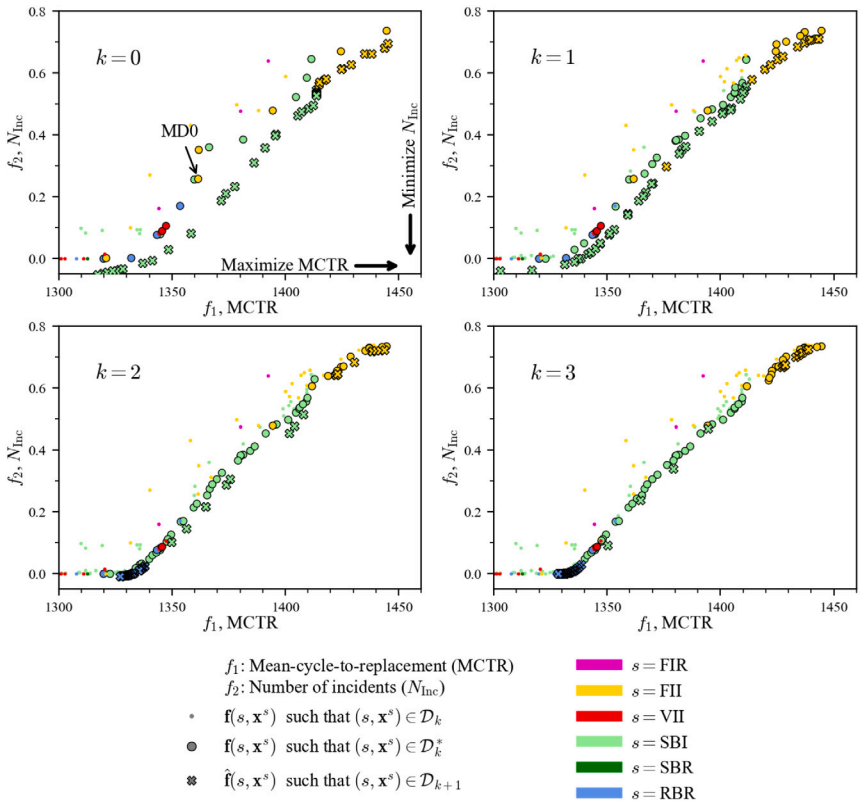

Fig. 12. Development of Pareto front in the first 4 iterations ($k = 0, 1, 2, 3$) of ELSA.

obtained after simulating 85 initial maintenance designs, i.e., $|\mathcal{D}_0^*| = 19$, and $|\mathcal{D}_0| = 85$. As an example, in Fig. 12 the objective vector $\mathbf{f} = [1361.4, 0.2590]$ is annotated as MD0, which corresponds to the maintenance design specified by the strategy s = FII and its design variables $\mathbf{x}^s = [147, 0.9667]$. This means that when using the maintenance design MD0, the aircraft components are expected to be utilized for 1361.4 flight cycles on average, and 0.2590 degradation incidents are expected to occur.

Now, the GP models $\mathcal{GP}_m^s$ are trained with the training data obtained at iteration $k = 0$. Using these updated GP models, ELSA adaptively samples new maintenance designs $(s, \mathbf{x}^s)$. Their pre-estimated objective vectors are shown as cross-points at iteration $k = 0$ in Fig. 12. Here, these pre-estimated objective vectors include the uncertainty of the GP models (see Eq. (27)), which is high at iteration $k = 0$ due to the small training data sets. Thus, the pre-estimated objective vectors are located far from the current Pareto front.

For these maintenance designs selected at iteration $k = 0$, MC simulations are conducted at iteration $k = 1$. As a result, ELSA finds additional Pareto optimal maintenance designs, especially in the region where the GP models predict new Pareto optimal solutions. As such, the Pareto front is pushed towards the lower-right corner as shown for $k = 1$ in Fig. 12. ELSA repeats the same steps of training the GP models, adaptive sampling, and simulation, for the next iterations $k = 1, 2, 3, \ldots, 19$. During the following iterations, the Pareto front is gradually improved, identifying new maintenance designs that dominate the old maintenance designs. For example, after iteration $k = 2$, MD0 that was Pareto optimal at $k = 0, 1$ is dominated by other maintenance designs identified at $k = 2$. Also, it is noted that the gap between the current Pareto front and the pre-estimated objective vectors decreases as the uncertainty of the GP models is reduced by the increased training data sets in iteration $k = 1, 2, 3$. Finally, after simulating 1035 maintenance designs during 19 iterations, ELSA is stopped as the predetermined total number of simulations is reached.

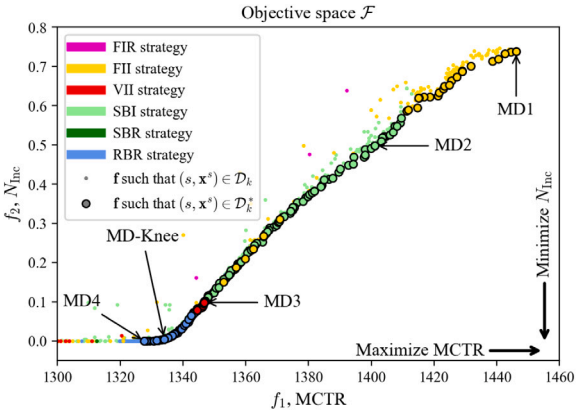

Fig. 13. Pareto optimal designs of aircraft maintenance strategies obtained by the proposed framework.

In Fig. 13, the final Pareto front generated by our framework shows a clear trade-off between cost-related objective (MCTR) and reliability-related objective (N_{Inc}). In the lower-left part of the front, there are maintenance designs that have nearly zero degradation incidents (low N_{Inc}), but achieve this by replacing the components quite early (low MCTR), wasting the useful life of the components. Such maintenance designs achieve high reliability in terms of minimizing N_{Inc}, but utilize the components inefficiently (small MCTR). On the other hand, the maintenance designs in the upper-right corner of Fig. 13 may result in some degradation incidents (high N_{Inc}), but the components are utilized for a longer time (high MCTR). Between these two maintenance designs, there are maintenance designs that balance reliability and cost, i.e., having moderate N_{Inc} and MCTR. Such a trade-off is often the consideration of aircraft maintenance decision-makers.

Table 4
Examples of 5 Pareto optimal maintenance designs $(s, \mathbf{x}^s)$ and their objective vectors $\mathbf{f} = [f_1, f_2]$. The table is sorted by the descending order of MCTR(f_1). The objective vectors are also annotated in the Pareto front in Fig. 13.

Annotation	Strategy type(s)	Design vector ($\mathbf{x}^s$)	MCTR (f_1)	N_{Inc} (f_2)	Selected under the following preference
MD1	FII	[399, 0.9999]	1446.2	0.7387	Maximize MCTR
MD2	SBI	[0.9, 200, 0.9982]	1401.1	0.4978	Maximize MCTR while $N_{\text{Inc}} \leq 0.5$
MD3	VII	[874, 0.9985, 0.9978]	1346.7	0.0985	Maximize MCTR while $N_{\text{Inc}} \leq 0.1$
MD-Knee	RBR	[22.86]	1334.1	0.0049	Knee point
MD4	RBR	[29.19]	1327.7	$< 10^{-4}$	Minimize N_{Inc}

4.3. Selecting reliable and cost-efficient aircraft maintenance designs

With the knowledge of which maintenance designs are dominating in terms of reliability and cost, decision-makers are expected to select a Pareto optimal design that reflects best their preferences. Below we discuss the selection of a Pareto optimal maintenance design.

Extreme aircraft maintenance designs

From the Pareto front in Fig. 13, there are two extreme maintenance designs that only maximize MCTR or only minimize N_{Inc}. These two extreme maintenance designs are annotated as MD1 and MD4 in Fig. 13 and Table 4. MD1 has the highest MCTR (1446.2 flight cycles), but its N_{Inc} is worst among all Pareto optimal designs. Design MD1 results in the highest cost-efficiency by utilizing the components for the longest time, but leads to the lowest reliability as the expected number of degradation incidents is highest. At the other extreme, MD4 is the most reliable but the least cost-efficient maintenance design, having the smallest N_{Inc} and the shortest MCTR. If the sole purpose of maintenance is to reduce the expected degradation incidents regardless of MCTR, then design MD4 is a suitable choice. However, neither MD1 nor MD4 is usually preferred. Rather, a balance between the two objectives is desirable.

Preference-based maintenance designs

In practice, aircraft maintenance is often expected to satisfy a certain level of reliability, e.g., the expected number of degradation incidents should be smaller than a threshold. When such a threshold-based *preference* for one of the objectives is known, then multi-objective decision making is straightforward. From the available Pareto optimal maintenance designs, we choose the one with the largest MCTR while having N_{Inc} below this threshold. For instance, if the decision-maker's preference is to keep $N_{\text{Inc}} \leq 0.5$, then the optimal maintenance design is MD2, which has the largest MCTR among the Pareto optimal designs having $N_{\text{Inc}} \leq 0.5$ (see Fig. 13 and Table 4). Similarly, we annotate the optimal maintenance designs MD3, when the preferences are $N_{\text{Inc}} \leq 0.1$.

Knee point-based maintenance designs

If preferences with respect to the objectives are not known or cannot be specified, then the *knee region* of the Pareto front is recommended for decision-makers [49,50]. The knee region is a convex region of the Pareto front with a strong curvature. The non-dominated solutions in the knee region are generally preferred because they provide the most beneficial trade-off, i.e., an objective is improved significantly at the cost of a slight deterioration of the other objective. Outside of the knee region, an objective is significantly deteriorated to achieve a slight improvement in the other objective.

The knee region of the Pareto front in Fig. 13 is occupied by maintenance designs using the RBR strategy. In this knee region, the RBR strategy dominates other types of strategies. This shows that the RBR strategy balances MCTR and N_{Inc}, which are conflicting objectives.

The point that achieves the maximum trade-off in the knee region is known as the *knee point*, and is defined as follows. Given a Pareto front, two extreme points $\mathbf{f}^1$ and $\mathbf{f}^2$ are obtained. Then, for a Pareto optimal point $\mathbf{f}$, the angle between two lines $L^1(\mathbf{f}^1, \mathbf{f})$ and $L^2(\mathbf{f}^2, \mathbf{f})$ is defined as the bend angle of $\mathbf{f}$. The Pareto optimal point having the maximum bend angle is defined as the *knee point* [49,50]. Fig. 14 visualizes this definition of the knee point.

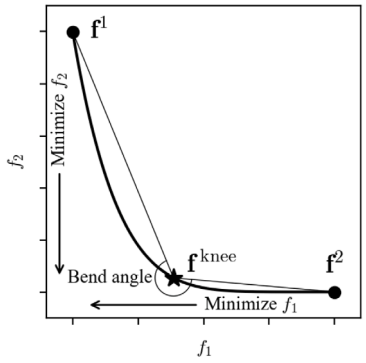

Fig. 14. Knee point defined by the bend angle.

Fig. 13 shows that the knee point MD-Knee is also obtained using the RBR strategy with $\rho_{\text{Rep}} = 22.86$, i.e., a replacement is scheduled when the estimated RUL is smaller than 22.86 flight cycles. Under the maintenance design MD-Knee, the components are utilized for 1334.1 flight cycles on average, and 0.0049 degradation incidents are expected. Compared to MD3, MD-Knee has a slightly shorter MCTR (99% of MD3), but a significantly smaller number of degradation incidents (5% of MD3). Thus, the MD-Knee shows that the RBR strategy enables the most beneficial performance with respect to MCTR and the number of incidents.

Discussion on the benefit of novel predictive strategies for aircraft maintenance

The analysis of the Pareto front shows that novel predictive strategies such as the RBR maintenance strategy has benefits in balancing the reliability (N_{Inc}) and cost-efficiency (MCTR) of aircraft maintenance. In fact, the RBR maintenance strategy results in a maximal trade-off between reliability and cost-efficiency, when compared with TBM strategies (FIR, FII) and CBM strategies (VII, SBI, SBR). Fig. 13 shows that the maintenance designs using the RBR strategy dominate all other strategies in the knee region of the Pareto front.

This performance of the RBR strategy can be explained by the fact that the use of sensors and data-driven RUL prognostics algorithms leads to a higher exploitation of components (MCTR) without generating additional degradation incidents. In contrast, TBM strategies such as FII and CBM strategies such as VII, SBI, and SBR rely less on data analytics to plan component replacements, affecting the exploitation time of the components. This performance of the RBR strategy is obtained even after we assumed that sensor monitoring is less accurate than visual inspections ($\sigma_{\text{Sen}} > \sigma_{\text{Ins}}$).

Maintenance designs based on the RBR strategy (MD-Knee and MD4) show a high performance also for other reliability and cost-related objectives. Fig. 15 shows that design MD-Knee outperforms the other Pareto optimal maintenance designs (see Table 4) in terms of N_{Uns}, N_{Inc}, and T_{Delay}. MD-Knee reduces 99% of unscheduled replacements compared to MD1. Thus, most replacements are scheduled in advance under MD-Knee, which provides early demand information for the supply management of parts. This is achieved only with a small increase in the cost-related objectives, e.g., N_{Rep} of MD-Knee is just 1.3% higher than that of MD3.

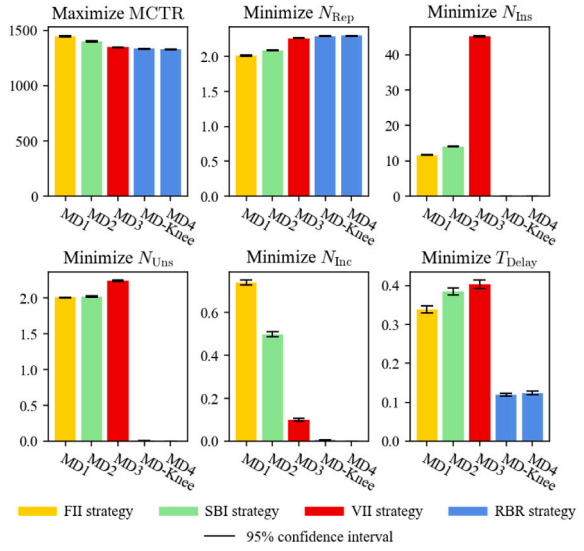

Fig. 15. Multiple objectives of the Pareto optimal designs shown in Table 4.

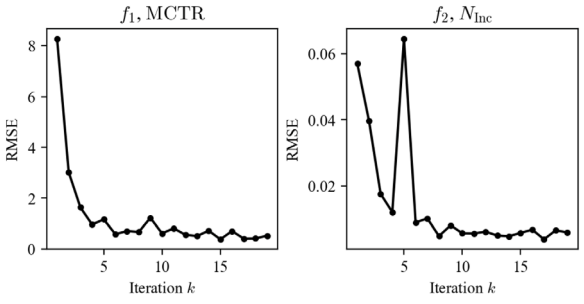

Fig. 16. RMSE of the pre-estimation made by the GP model at each iteration.

5. Quality of the Pareto front

In this section, we analyze the quality of the Pareto front generated by ELSA.

5.1. The quality of the pre-estimations made by the GP models

ELSA relies on the pre-estimation of the objectives $\hat{f}_m(s, \mathbf{x}^s)$ made by the GP model $\mathcal{GP}_m^s$ during Step 4: adaptive sampling (see Section 3.5). At the end of iteration k, we obtain the objective values of the adaptively sampled maintenance designs $f_m(s, \mathbf{x}^s)$ by means of Monte Carlo simulation. With this, we determine the root-mean-square-error (RMSE) between $f_m(s, \mathbf{x}^s)$ and the pre-estimation $\hat{f}_m(s, \mathbf{x}^s)$, i.e.,

$$\text{RMSE} = \sqrt{\mathop{\mathbb{E}}_{\mathcal{D}_k - \mathcal{D}_{(k-1)}} \left(\hat{f}_m(s, \mathbf{x}^s) - f_m(s, \mathbf{x}^s) \right)^2}, \tag{28}$$

where the set $\mathcal{D}_k - \mathcal{D}_{(k-1)}$ denotes the designs that are selected for simulation at iteration k.

Fig. 16 shows the RMSE obtained for objectives MCTR and N_{Inc} during several iterations of ELSA. Overall, the RMSE is small compared to the scales of the two objectives shown in the Pareto front in Fig. 13. This shows that the GP models provide reliable pre-estimations in the adaptive sampling step of ELSA. In addition, the RMSE decreases further in later iterations as the GP models are updated with more training data.

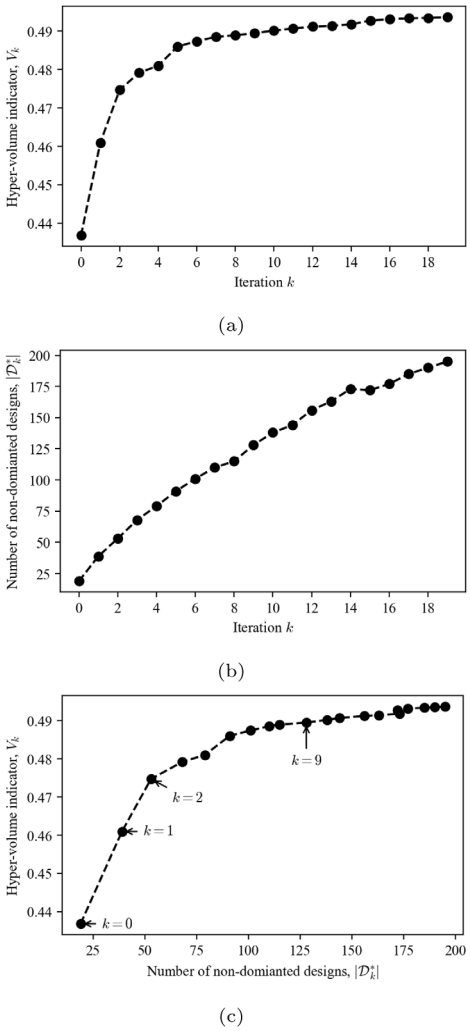

Fig. 17. (a) Hyper-volume indicator (V_k) at iteration k. (b) Number of Pareto optimal maintenance strategies ($|D_k^*|$) at iteration k. (c) The balance between exploration (V_k) and exploitation ($|D_k^*|$).

5.2. The quality of the pareto front obtained using ELSA

Fig. 17 shows the hyper-volume V_k and the number of Pareto optimal designs $|D_k^*|$ obtained using ELSA. Here, V_k indicates the level of exploration achieved by ELSA, while $|D_k^*|$ indicates the level of exploitation achieved.

The results show that the exploration of the design space of aircraft maintenance is largely achieved in the first iteration (see Fig. 17(a)). At the initial iteration $k = 0$, $V_0 = 0.437$. At iteration $k = 1$, the hyper-volume is increased to $V_1 = 0.461$, which is 43% of the total improvement of V_k during the 19 iterations. This improvement can also be seen in Fig. 12.

In contrast to the rapid increase of the hyper-volume in the first iterations, the exploitation of the design space is gradual, as shown in Fig. 17(b). ELSA starts with 19 non-dominated maintenance designs, i.e., $|D_0^*| = 19$. During the following 19 iterations, $|D_k^*|$ increases gradually to a total of 195 maintenance designs that are Pareto optimal. This continuous and gradual increase of $|D_k^*|$ in the later iterations is explained by the fact that ELSA generates many new Pareto optimal maintenance designs with slightly different design variables. For example, ELSA generates maintenance designs (SBI, [0.8034, 169, 0.9997]), (SBI, [0.8033, 169, 0.9997]), and (SBI, [0.8053, 169, 0.9999]) as Pareto optimal solutions at the iteration $k = 16, 17, 19$ respectively, which are all very similar Pareto optimal maintenance designs.

As more training data $\mathcal{D}_k$ and $\mathcal{F}_k$ are available, ELSA shifts from more exploration in the early iterations to more exploitation in the later iterations. This shift is shown in Fig. 17(c). During early iterations $k = 1, 2, 3$, ELSA selects those new maintenance designs that can improve V_k significantly (exploration). In the later iterations $k \geq 5$, ELSA selects those new maintenance designs that can improve $|D_k^*|$ (exploitation), rather than V_k. For example, at iteration $k = 9$, ELSA identifies 13 new Pareto optimal designs, but the increase of V_k is modest. This behavior of ELSA is explained by the fact that the pre-estimation in the adaptive sampling step considers both the mean and the uncertainty of the GP models (see Section 3.5).

5.3. Performance of ELSA vs. other algorithms

In this section, the performance of ELSA is compared against the performance of three state-of-the-art algorithms used to solve multi-objective optimization problems: NSGA-II [27,28], ReSPIR [29], and EGO [32,33].

NSGA-II is an evolutionary algorithm often used to solve multi-objective optimization problems [27,28]. As like traditional genetic algorithms, NSGA-II iteratively improves the Pareto front of the considered problem. At each iteration, NSGA-II evaluates the *non-dominated rank* of the current Pareto optimal solutions. Then, new maintenance designs are generated using the typical operations of genetic algorithms: selection, crossover, and mutation. The objectives of these newly generated maintenance designs are evaluated, and the Pareto front is updated accordingly. Unlike ELSA where GP models is used to rapidly pre-estimate the objectives, NSGA-II does not rely on any surrogate models for the selection of new designs. Thus, NSGA-II selects new designs without prior knowledge of their objective vectors.

ReSPIR is an algorithm that iteratively generates a Pareto front using a surrogate model [29]. Similar to ELSA, at every iteration, ReSPIR (1) simulates several designs, (2) constructs a surrogate model using radial-basis-functions (RBF), (3) uses this surrogate model to pre-estimate the objectives of the designs that have not yet been explored, and (4) among the designs evaluated in step 3, it selects those designs whose pre-estimated objectives dominate the current Pareto optimal designs. Although ReSPIR uses a RBF as a surrogate model, the uncertainty of this surrogate model is not considered. Because of this, ReSPIR may not explore enough the design space, and may converge prematurely to a certain area of the design space. In addition, ReSPIR requires to pre-estimate the objectives of all possible designs, which is not feasible in the case of aircraft maintenance design where an infinite number of designs exists due to the fact that there are continuous design variables. In contrast, ELSA explicitly considers the uncertainty of the GP learning models, and is able to handle continuous design variables since it adaptively samples a finite number of maintenance designs.

EGO is also a surrogate-model-based algorithm that iteratively updates the Pareto front [32,33]. Similar to ELSA, EGO uses GP models as surrogate models. However, while ELSA uses an adaptive sampling step to select new designs to further explore, EGO selects new designs that maximize an infill-criteria. This infill-criteria is evaluated using GP models. For our case study, EGO is implemented using an *expected-improvement-matrix-based infill-criteria*, which has been shown to require the least computational time when compared with other infill-criteria [32,33]. Maximizing this infill-criteria is done using a typical genetic algorithm. As a last step for EGO, only those maintenance designs that maximize this infill-criteria are actually simulated in the next iteration.

For comparison, NSGA-II, ReSPIR and EGO are used to solve the multi-objective aircraft maintenance design problem formulated in Eq. (2). Fig. 18 shows that ELSA outperforms EGO, ReSPIR and NSGA-II by generating a larger hyper-volume V_k and by identifying a larger number of Pareto optimal aircraft maintenance designs $|D_k^*|$. Here, the number of maintenance designs that are simulated $|D_k|$ is used as a metric for the computational cost of the algorithms.

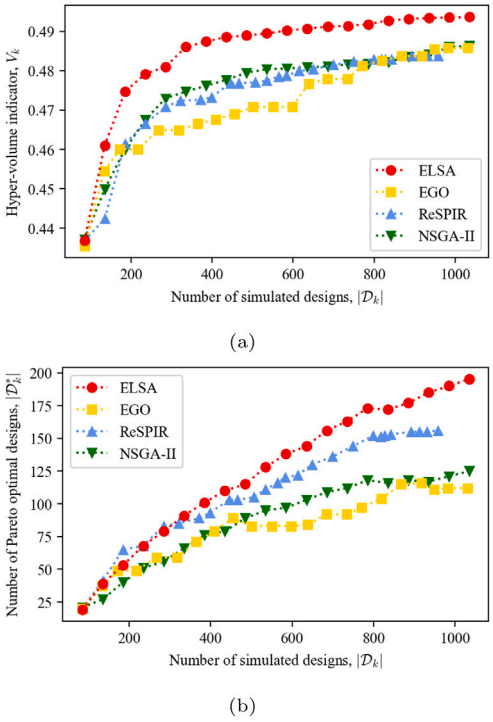

Fig. 18. The performance of ELSA relative to benchmark algorithms EGO, ReSPIR and NSGA-II.

Fig. 18(a) shows that all four algorithms improve V_k rapidly in their early iterations, but ELSA improves V_k the fastest. ELSA achieves $V_3 = 0.479$ after simulating only 235 designs, while EGO, ReSPIR, and NSGA-II achieve the same hyper-volume after simulation many more designs (770, 614, and 485, respectively). This is explained by the fact that ELSA explicitly considers the uncertainty of the GP models in the selection of new designs to be simulated (see Section 3.5). Also, the final V_k obtained by ELSA is the largest, which shows that ELSA explores the design space the most. In fact, EGO, ReSPIR, and NSGA-II achieve around 98% of the V_k achieved by ELSA.

Fig. 18(b) shows that ELSA also outperforms the other three algorithms by generating the most Pareto optimal maintenance designs. Specifically, 195 Pareto optimal designs are identified by ELSA, while EGO, ReSPIR and NSGA-II identify only 112, 156, 125 Pareto optimal designs, respectively. Thus, the other algorithms generate only 57−80% of the Pareto optimal maintenance designs generated by ELSA. Compared to the performance difference in V_k (exploration), the difference in $|D_k^*|$ is larger in general. This is due to ELSA's adaptive sampling step which enables the exploitation of maintenance designs that are close to the other Pareto optimal solutions. In the case of NSGA-II, for example, new designs are often significantly different from the already evaluated designs because of crossover and mutation operations. In the case of EGO, since the infill-criteria measures the level of exploration only, the selection of new maintenance designs may not aim to increase the number of Pareto optimal designs.

6. Conclusion

In this paper, we have proposed a framework to design multi-objective aircraft maintenance, considering the objectives representing cost and reliability. For this, we construct a generic aircraft maintenance model that accommodates a variety of types of maintenance designs. To efficiently select those maintenance designs to be analyzed, we propose a design space exploration algorithm using Gaussian process learning models and a novel adaptive sampling method (ELSA). We

illustrate the proposed framework for the maintenance design of multi-component aircraft systems with k-out-of-n redundancy, considering 6 different maintenance strategy types which range from traditional time-based maintenance strategies to novel predictive maintenance strategies. With our proposed approach, we identify 195 Pareto optimal maintenance designs.

The obtained Pareto front of maintenance designs shows that the RUL-based predictive strategy is beneficial in balancing reliability and cost of maintenance. In particular, this RUL-based strategy dominates other maintenance strategies in the knee region of the Pareto front where conflicting objectives are balanced. This result also provides decision-makers with arguments for transition to novel predictive maintenance strategies.

In addition, we also show that ELSA outperforms other state-of-the-art algorithms by generating a Pareto front with the most non-dominated designs and the largest hyper-volume. This is due to the fact that adaptive sampling method of ELSA balances between exploration and exploitation of the design space.

Last, but not least, our proposed framework is generic and can readily be applied for the maintenance design of other aircraft systems, various types of maintenance strategies, and other objectives, depending on the need of the decision-makers.

As future work, apart from the two objectives considered for the maintenance of the brakes, we plan to further explore the predictive maintenance strategies by considering additional objectives that reflect the interests of the decision-makers such as reliability-related objectives considering different severity levels of the degradation incidents, and cost-related objectives that explicitly integrate airline-specific cost models. Furthermore, we aim to apply our proposed aircraft maintenance model for other aircraft systems and structures. In these cases, we plan to investigate the functional dependency of k-out-of-n redundant systems, and the impact of operational conditions on the degradation process.

CRediT authorship contribution statement

Juseong Lee: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Mihaela Mitici:** Conceptualization, Methodology, Validation, Investigation, Writing - review & editing, Supervision, Project administration, Funding acquisition.

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.

Acknowledgment

This research has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 769288.

References

[1] Force IMCT. Airline maintenance cost executive commentary (FY2016 data). Technical report, IATA's Maintenance Cost Task Force; 2017.
[2] Association AT. ATA MSG-3, operator/manufacturer scheduled maintenance volume 1 - fixed wing aircraft, vol. 1 - fixed. Air Transport Association of America; 2013.
[3] Wang H. A survey of maintenance policies of deteriorating systems. European J Oper Res 2002;139(3):469–89. http://dx.doi.org/10.1016/S0377-2217(01)00197-7.
[4] Alaswad S, Xiang Y. A review on condition-based maintenance optimization models for stochastically deteriorating system. Reliab Eng Syst Saf 2017;157:54–63. http://dx.doi.org/10.1016/j.ress.2016.08.009.
[5] Sprong J, Jiang X, Polinder H. A deployment of prognostics to optimize aircraft maintenance - a literature review. In: Proceedings of the annual conference of the prognostics and health management society, PHM. 11, (1):2019, p. 1–12. http://dx.doi.org/10.36001/phmconf.2019.v11i1.776.
[6] Si XS, Wang W, Hu CH, Zhou DH. Remaining useful life estimation - A review on the statistical data driven approaches. European J Oper Res 2011;213(1):1–14. http://dx.doi.org/10.1016/j.ejor.2010.11.018.
[7] Zhao Z, Bin Liang, Wang X, Lu W. Remaining useful life prediction of aircraft engine based on degradation pattern learning. Reliab Eng Syst Saf 2017;164(457):74–83. http://dx.doi.org/10.1016/j.ress.2017.02.007.
[8] Li X, Ding Q, Sun JQ. Remaining useful life estimation in prognostics using deep convolution neural networks. Reliab Eng Syst Saf 2018;172(June 2017):1–11. http://dx.doi.org/10.1016/j.ress.2017.11.021.
[9] European Aviation Safety Agency (EASA). Aircraft health monitoring (AHM) integration in MSG-3, IP180. In: International maintenance review board policy board (IMRBPB) issue paper (IP). 2, 2018, p. 1–33.
[10] Compare M, Martini F, Zio E. Genetic algorithms for condition-based maintenance optimization under uncertainty. European J Oper Res 2015;244(2):611–23. http://dx.doi.org/10.1016/j.ejor.2015.01.057.
[11] Ferreira RJ, de Almeida AT, Cavalcante CA. A multi-criteria decision model to determine inspection intervals of condition monitoring based on delay time analysis. Reliab Eng Syst Saf 2009;94(5):905–12. http://dx.doi.org/10.1016/j.ress.2008.10.001.
[12] Cavalcante CA, Lopes RS, Scarf PA. Inspection and replacement policy with a fixed periodic schedule. Reliab Eng Syst Saf 2021;208(November 2020):107402. http://dx.doi.org/10.1016/j.ress.2020.107402.
[13] Kallen M, van Noortwijk J. Optimal maintenance decisions under imperfect inspection. Reliab Eng Syst Saf 2005;90(2–3):177–85. http://dx.doi.org/10.1016/j.ress.2004.10.004.
[14] Curcuru G, Galante G, Lombardo A. A predictive maintenance policy with imperfect monitoring. Reliab Eng Syst Saf 2010;95(9):989–97. http://dx.doi.org/10.1016/j.ress.2010.04.010.
[15] Nguyen KT, Medjaher K. A new dynamic predictive maintenance framework using deep learning for failure prognostics. Reliab Eng Syst Saf 2019;188(September 2018):251–62. http://dx.doi.org/10.1016/j.ress.2019.03.018.
[16] Lu B, Chen Z, Zhao X. Data-driven dynamic predictive maintenance for a manufacturing system with quality deterioration and online sensors. Reliab Eng Syst Saf 2021;212(March):107628. http://dx.doi.org/10.1016/j.ress.2021.107628.
[17] Lee J, Mitici M. An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets. Reliab Eng Syst Saf 2020;202:107052. http://dx.doi.org/10.1016/j.ress.2020.107052.
[18] FAA Flight Operations Evaluation Board (FOEB). Master minimum equipment list (MMEL) airbus A350-900 series, all models, vol. 1. 1, 2017.
[19] FAA Flight Operations Evaluation Board (FOEB). Master Minimum Equipment List BOEING 787. U.S. Department of Transportation Federal Aviation Administration; 2015.
[20] Huynh K, Barros A, Bérenguer C, Castro IT. A periodic inspection and replacement policy for systems subject to competing failure modes due to degradation and traumatic events. Reliab Eng Syst Saf 2011;96(4):497–508. http://dx.doi.org/10.1016/j.ress.2010.12.018.
[21] Huynh KT. An adaptive predictive maintenance model for repairable deteriorating systems using inverse Gaussian degradation process. Reliab Eng Syst Saf 2021;213(April). http://dx.doi.org/10.1016/j.ress.2021.107695.
[22] Chen N, Ye ZS, Xiang Y, Zhang L. Condition-based maintenance using the inverse Gaussian degradation model. European J Oper Res 2015;243(1):190–9. http://dx.doi.org/10.1016/j.ejor.2014.11.029.
[23] Do P, Voisin A, Levrat E, Iung B. A proactive condition-based maintenance strategy with both perfect and imperfect maintenance actions. Reliab Eng Syst Saf 2015;133:22–32. http://dx.doi.org/10.1016/j.ress.2014.08.011.
[24] Grall A, Dieulle L, Bérenguer C, Roussignol M. Continuous-time predictive-maintenance scheduling for a deteriorating system. IEEE Trans Reliab 2002;51(2):141–50. http://dx.doi.org/10.1109/TR.2002.1011518.
[25] Syan CS, Ramsoobag G. Maintenance applications of multi-criteria optimization: A review. Reliab Eng Syst Saf 2019;190(May):106520. http://dx.doi.org/10.1016/j.ress.2019.106520.
[26] Kang E, Jackson E, Schulte W. An approach for effective design space exploration. In: Monterey workshop. LNCS, vol. 6662, Berlin: Springer; 2010, p. 33–54. http://dx.doi.org/10.1007/978-3-642-21292-5_3.
[27] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 2002;6(2):182–97. http://dx.doi.org/10.1109/4235.996017.
[28] Palesi M, Givargis T. Multi-objective design space exploration using genetic algorithms. In: Hardware/software codesign - proceedings of the international workshop. 2002, p. 67–72. http://dx.doi.org/10.1145/774801.774804.
[29] Palermo G, Silvano C, Zaccaria V. ReSPIR: A response surface-based pareto iterative refinement for application-specific design space exploration. IEEE Trans Comput-Aided Des Integr Circuits Syst 2009;28(12):1816–29. http://dx.doi.org/10.1109/TCAD.2009.2028681.

[30] Rasmussen CE, Williams CK. Gaussian processes for machine learning. MIT Press; 2006.
[31] Frazier PI. A tutorial on Bayesian optimization. (Section 5):2018, p. 1–22, arXiv arXiv:1807.02811.
[32] Rojas-Gonzalez S, Van Nieuwenhuyse I. A survey on kriging-based infill algorithms for multiobjective simulation optimization. Comput Oper Res 2020;116:104869. http://dx.doi.org/10.1016/j.cor.2019.104869.
[33] Zhan D, Cheng Y, Liu J. Expected improvement matrix-based infill criteria for expensive multiobjective optimization. IEEE Trans Evol Comput 2017;21(6):956–75. http://dx.doi.org/10.1109/TEVC.2017.2697503.
[34] Couckuyt I, Deschrijver D, Dhaene T. Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization. J Global Optim 2014;60(3):575–94. http://dx.doi.org/10.1007/s10898-013-0118-2.
[35] Simon D. Evolutionary optimization algorithms. Wiley; 2013.
[36] van Noortwijk J. A survey of the application of gamma processes in maintenance. Reliab Eng Syst Saf 2009;94:2–21. http://dx.doi.org/10.1016/j.ress.2007.03.019.
[37] Zhu S, van Jaarsveld W, Dekker R. Spare parts inventory control based on maintenance planning. Reliab Eng Syst Saf 2020;193(August 2019):106600. http://dx.doi.org/10.1016/j.ress.2019.106600.
[38] Braglia M, Grassi A, Montanari R. Multi-attribute classification method for spare parts inventory management. J Qual Maint Eng 2004;10(1):55–65. http://dx.doi.org/10.1108/13552510410526875.
[39] de Pater I, Mitici M. Predictive maintenance for multi-component systems of repairables with Remaining-Useful-Life prognostics and a limited stock of spare components. Reliab Eng Syst Saf 2021;214:107761. http://dx.doi.org/10.1016/j.ress.2021.107761.
[40] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, douard Duchesnay E. Scikit-learn: Machine learning in python. J Mach Learn Res 2011;12(1):2825—-2830. http://dx.doi.org/10.1145/2786984.2786995.
[41] Sheng J, Prescott D. A coloured Petri net framework for modelling aircraft fleet maintenance. In: Reliability engineering and system safety, vol. 189. Elsevier Ltd; 2019, p. 67–88. http://dx.doi.org/10.1016/j.ress.2019.04.004.
[42] Zámková M, Prokop M, Stolín R. Factors influencing flight delays of a European airline. Acta Univ Agric Silvicult Mendelianae Brunensis 2017;65(5):1799–807. http://dx.doi.org/10.11118/actaun201765051799.
[43] Kleijnen JP. Design of experiments: Overview. In: Proceedings of the 2008 Winter Simulation Conference. IEEE; 2008, p. 479–88. http://dx.doi.org/10.1109/WSC.2008.4736103.
[44] Almotahari A, Yazici A. A computationally efficient metric for identification of critical links in large transportation networks. Reliab Eng Syst Saf 2021;209(January):107458. http://dx.doi.org/10.1016/j.ress.2021.107458.
[45] He X, Chen X, Xiong C, Zhu Z, Zhang L. Optimal time-varying pricing for toll roads under multiple objectives: A simulation-based optimization approach. Transp Sci 2017;51(2):412–26. http://dx.doi.org/10.1287/trsc.2015.0661.
[46] Chen J, Xin B, Peng Z, Dou L, Zhang J. Optimal contraction theorem for exploration-exploitation tradeoff in search and optimization. IEEE Trans Syst Man Cybern A 2009;39(3):680–91. http://dx.doi.org/10.1109/TSMCA.2009.2012436.
[47] Zitzler E, Thiele L, Laumanns M, Fonseca CM, Da Fonseca VG. Performance assessment of multiobjective optimizers: An analysis and review. IEEE Trans Evol Comput 2003;7(2):117–32. http://dx.doi.org/10.1109/TEVC.2003.810758.
[48] Fleischer M. The measure of Pareto optima applications to multi-objective metaheuristics. In: International conference on evolutionary multi-criterion optimization. Berlin: Springer; 2003, p. 519–33. http://dx.doi.org/10.1007/3-540-36970-8_37.
[49] Greco S, Ehrgott M, Figueira JR. Multiple criteria decision analysis. Springer New York; 2016, http://dx.doi.org/10.1007/978-1-4939-3094-4.
[50] Deb K, Gupta S. Understanding knee points in bicriteria problems and their implications as preferred solution principles. Eng Optim 2011;43(11):1175–204. http://dx.doi.org/10.1080/0305215X.2010.548863.