

Predictive Aircraft Maintenance

Integrating Remaining-Useful-Life Prognostics into Maintenance Optimization

Lee, J.

DOI

[10.4233/uuid:b219e3e7-285b-44cd-a0ff-47f03fb9f3ac](https://doi.org/10.4233/uuid:b219e3e7-285b-44cd-a0ff-47f03fb9f3ac)

Publication date

2022

Document Version

Final published version

Citation (APA)

Lee, J. (2022). *Predictive Aircraft Maintenance: Integrating Remaining-Useful-Life Prognostics into Maintenance Optimization*. [Dissertation (TU Delft), Delft University of Technology].
<https://doi.org/10.4233/uuid:b219e3e7-285b-44cd-a0ff-47f03fb9f3ac>

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.

PREDICTIVE AIRCRAFT MAINTENANCE

INTEGRATING REMAINING-USEFUL-LIFE PROGNOSTICS
INTO MAINTENANCE OPTIMIZATION

PREDICTIVE AIRCRAFT MAINTENANCE

INTEGRATING REMAINING-USEFUL-LIFE PROGNOSTICS INTO MAINTENANCE OPTIMIZATION

DISSERTATION

for the purpose of obtaining the degree of doctor
at Delft University of Technology,
by the authority of the Rector Magnificus, Prof.dr.ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates,
to be defended publicly on
Wednesday 7 December 2022 at 15:00 o'clock

by

Juseong LEE

Master of Science in Aerospace Engineering,
KAIST (Korea Advanced Institute of Science and Technology), South Korea,
born in Suwon, South Korea.

This dissertation has been approved by the promotor.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof. dr. ir. M. Mulder,	Delft University of Technology, promotor
Dr. M.A. Mitici,	Utrecht University, copromotor

Independent members:

Prof. dr. N.P. Avdelidis	Cranfield University
Prof. dr. J.H. Choi	Korea Aerospace University
Prof. dr. ir. T. Tinga	University of Twente
Dr. R.M. Groves	Delft University of Technology
Prof. dr. ir. L.L.M. Veldhuis	Delft University of Technology
Prof. dr. G.C.H.E. de Croon	Delft University of Technology, reserve member

This research work is part of ReMAP project, which received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 769288.



Keywords: Aircraft Maintenance, Predictive Maintenance,
Remaining-Useful-Life Prognostics,
Scheduling, Optimization, Modeling and Simulation

Printed by: U2pi BV, den Haag, the Netherlands

Front & Back: Image generated by DALL-E 2, and designed by Y. Kim

Copyright © 2022 by J. Lee

ISBN 978-94-9329-948-1

An electronic version of this dissertation is available at
<http://repository.tudelft.nl/>.

*We choose to go to the Moon in this decade, and do the other things,
not because they are easy, but because they are hard.*

John F. Kennedy

ACKNOWLEDGEMENTS

Even the best pilot cannot fly alone. A flight is accomplished only by the magical collaboration of all people working together — co-pilots, cabin crews, air traffic controllers, mechanics, and much more. The journey to PhD is similar to a flight. It is full of excitement as I am exploring a new world that no one has reached so far. However, it is also full of uncertainty. I was able to complete four years of my journey due to the support from my respectful supervisors, dedicated collaborators, inspiring colleagues, precious friends, and loved family.

First and foremost, I appreciate my knowledgeable supervisor **Dr. Mihaela Mitici**. I could finish my PhD because of your guidance and advice. All your dedication to our research plans, manuscripts, and debates raised me as an independent researcher. You will be my memorable supervisor, and all your lessons will be the valuable foundation of my future research. Also, I would like to thank my respected promotor **Prof. Max Mulder**. Your positive words gave me the power to continue, especially when I struggled with the uncertainty and challenges of PhD life. I was so lucky to have you on board.

Also, I would like to express special thanks to all the partners of ReMAP project. Their knowledge, experience, and feedback enriched my research with realistic case studies. KLM partners brought abundant field knowledge. I particularly appreciate **Floris Freeman**, who put unimaginable effort to fill the gap between academia and industry. ONERA partners, especially **Dr. Pierre Bieber**, supplemented my research by providing comments from new perspectives.

I also would like to make an acknowledgement to my previous supervisor, **Prof. Jaemyung Ahn**. With your encouragement and guidance, I was able to start a new challenge abroad. Throughout my PhD, you have been my role model as a researcher. It is my great luck to have you as my first supervisor.

In my research group, Air Transport & Operations, I had amazing colleagues. Having them, I enjoyed every hours at ATO without being tired. **Matthieu Vert**, you are my best discussion partner for all sorts of imaginable topics in science and philosophy. You always inspired me to see the world from novel perspectives. **Hao Ma**, I enjoyed all your “short questions,” which often took me hours to answer. Forever you will be remembered as my best tea time friend. **Chengpeng Jiang**, we started together four years ago. As you have been my running mate, I could continue through this long journey. Thank you. **Ingeborg de Pater**, you have been the best collaborator in my research. Your sharp critics and kind comments added great value to my work. **Mike Zoutendijk**, and **Simon van Oosterom**, thank you for proofreading my dissertation. Also, thank you for all your tips that helped me survive in the Netherlands. I appreciate all ATO students — **Marie Bieber**, **Malte von der Burg**, **Mahdi Noorafza**, **Ilias Parmaksizoglou**, **Thomas Pioger**, **Iordanis Tseremoglou**, **Jorick Kamphof**, **Dr. Wenhua Qu**, and **Dr. Gülçin Ermiş**. The coffee and beers we drank together were the main propellants for my journey. Also, I wish to extend my special thanks to the ATO alumni who helped me settle down in Delft.

Thank you **Dr. Vis Dhanisetty**, **Dr. Stef Janssen**, **Hemmo Koornneef**, **Dr. Qichen Deng**, **Dr. Alessandro Bombelli**, **Dr. Raissa Li**, and **Dr. Vinh Ho-Huu**. Lastly, thanks to the professors and staff of ATO: **Prof. Henk Blom**, **Dr. Bruno Santos**, **Ir. Paul Roling**, **Ir. Elise Bavelaar**, **Dr. Márcia Baptista**, and **Nathalie Zoet**.

Besides from research, my life in Delft has been full of joy, thanks to my precious friends. **Jaekyum Lee**, I am glad we found each other on the first day at TU Delft. You have been my best drinking buddy, dependable consultant, and friendly brother. **Yeun Kim**, thank you for taking care of the aesthetic aspect of my dissertation. **Dr. Dongil Shin**, I enjoyed discussing various topics with you. **Soyeon Kim**, looking at your endless energy motivated me to enjoy life with enthusiasm. **Hanjun Kim**, playing with you relieved me when I was exhausted. **Jeseung Moon**, I learned a positive mind and warm heart from you. I sincerely appreciate all my friends who shared my joy and sorrow.

I also would like to make a thankful note to all my friends in Korea, who have encouraged and delighted me from the other side of the world. 8,600 km distance and 8 hours time difference couldn't be a barrier to our friendship. Thank you **Jae-youl Ko**, **Eunkwang Lee**, **Yongtak Kim**, **Dr. Eric Won Keun Chang**, **Dr. Haisol Kim**, **Woosang Park**, **Hyungjun Park**, and **Dr. Youngju Song**.

Above all, I am grateful for my best supporter **Minjung Jun**. You willingly followed me wherever I traveled. You thoughtfully listened to me, whatever I said. You took care of me whenever I was tired of life. Thanks for being my best friend, rational discussion partner, strongest advocate, and sweetheart. Meeting you in the far Netherlands was the most appreciated thing of my Dutch life.

Finally, I would like to give all my honor to my family. My parents, **Kang Young** and **Jeongsin**, have given me devoted love, unconditional support, and heartfelt advice. When I decided to start a new challenge in the far Netherlands, they worried me with warm hearts. When I despaired and wanted to give up, they encouraged me to stay strong. When I published my papers, they celebrated me by trying to understand these. If there is one who will be happier than me when I am promoted to a PhD, that would be my father and mother. I am sincerely grateful for their love. Of course, I am thankful to my dependable brother, **Juwon**. As he cares for my loved family, I could focus on my work and complete all these works.

Thank you all.

Juseong Lee

Delft, 3 November, 2022

SUMMARY

Current aircraft maintenance ensures safe and reliable flight operations based on inspections repeated at fixed time intervals. The time interval between inspections is often much shorter than the average life of aircraft components, in an effort to timely detect potential failures. While this approach successfully prevents most potential failures, it is not the most efficient since airlines frequently need to ground aircraft for visual inspections. Furthermore, most inspections do not find any fault; thus, nothing is actually repaired after these frequent inspections.

Predictive aircraft maintenance (PdAM) is a newly emerging approach to maintenance which is expected to be more efficient, while providing the same or higher levels of reliability. PdAM uses the data produced by the plethora of on-board sensors installed on modern aircraft to monitor the health condition of aircraft components, without the need to ground these aircraft for visual inspections. These health condition data are analyzed to predict the Remaining-Useful-Life (RUL) of aircraft components. The core idea of PdAM is to plan maintenance tasks based on the estimated RUL. PdAM is currently not fully implemented in practice, however. Regulatory bodies have only recently started to discuss the integration of aircraft health monitoring (AHM) systems into aircraft maintenance process.

This dissertation aims to identify and address the challenges in implementing PdAM. The first challenge is the lack of mathematical models to assess the performance of PdAM. Before implementing PdAM in actual aircraft, the expected performance needs to be quantified to understand the impact on reliability and cost-efficiency. Although a few studies have proposed aircraft maintenance models, these studies only consider cost as a single performance metric. However, it is clear that aircraft maintenance should also be evaluated in terms of reliability and other key performance indicators (KPIs) representing the various (and often conflicting) interests of all stakeholders involved. In this dissertation, we construct a mathematical model of PdAM to evaluate the balance in maximizing various KPIs altogether. Our model captures the stochastic degradation and failure of aircraft components, and the interactions between stakeholders during the maintenance decision making process.

The second challenge for PdAM is the lack of optimization frameworks to plan PdAM considering RUL prognostics. In the last decades, most researchers have focused on predicting the RUL of aircraft components, but only a few studies address the question of *how* to actually integrate RUL prognostics into maintenance planning. Aircraft maintenance planning is a very complex process that should consider different aircraft components, a fleet of aircraft, their flight schedules, the limited hangar availability, tight safety margins, and strict regulations. Considering all these together in a single optimization framework is overly demanding. Hence, in this dissertation, the optimization of the PdAM planning is performed at three levels: component level, fleet level, and strategy level.

At the component level, we propose probabilistic RUL prognostics and a deep reinforcement learning (DRL) approach for predictive maintenance planning. The probabilistic RUL prognostics estimate the probability distribution of RUL, instead of a point-estimation. This approach quantifies the uncertainty associated with RUL prognostics. Based on the estimated RUL distribution, the DRL approach determines the optimal moment to replace an aircraft component. In the case study for the maintenance of aircraft turbofan engine, the proposed DRL approach reduces the total maintenance cost by 29.3% and prevents 94.3% of unscheduled maintenance, compared to the case when the point-estimation of RUL is used.

At the fleet level, PdAM is planned by simultaneously considering a fleet of aircraft having multiple components. The main interest of fleet-level PdAM is to integrate RUL prognostics and operational requirements, such as the flight schedules and the limited hangar availability. We formulate these in an integer linear programming problem that minimizes the cost of fleet-level PdAM. This approach reduces the usage of hangars by grouping the schedule of maintenance tasks when the RUL of the components are similar. Considering the maintenance of aircraft landing gear brakes for a fleet of aircraft, the total maintenance cost is reduced by 20% compared to the traditional maintenance strategies.

At the maintenance strategy level, we optimize the design parameters of PdAM, such as safety margins and thresholds of RUL, considering multiple objectives: cost-efficiency and reliability. Since this multi-objective optimization problem is computationally intensive, we propose an efficient search algorithm using Gaussian process (GP) learning models to identify Pareto optimal design parameters of PdAM. Compared to other state-of-the-art multi-objective optimization algorithms, the proposed GP learning-based algorithm identifies more Pareto optimal solutions within the same computational time. The identified Pareto front shows that PdAM using RUL prognostics dominates traditional maintenance strategies by achieving the beneficial balance between efficiency and reliability indices. With only a 1% reduction in the efficiency index, the Pareto optimal PdAM strategy achieves a 95% improvement in the reliability index.

The three optimization frameworks at the three different levels of PdAM are proposed and illustrated for case studies on the maintenance of aircraft engines and landing gear brakes. These case studies show three main benefits of PdAM: 1) the maintenance cost is minimized by scheduling maintenance tasks only when necessary; 2) failures and unscheduled maintenance are prevented by considering RUL prognostics; and 3) Pareto optimal performance is achieved considering the balance between reliability and efficiency.

Finally, this dissertation identifies the emerging challenges associated with the introduction of PdAM. Such challenges are often attributable to the introduction of new technologies, such as aircraft health monitoring systems, RUL prognostics algorithms, and decision support systems to plan PdAM. Based on structured brainstorming sessions with domain experts and end-users, three major challenges of future PdAM are identified: 1) the (often unknown) reliability of new technologies, 2) the timeliness and accuracy of communication between the stakeholders of the new PdAM, and 3) the end-users' trust in the new technologies.

Throughout this dissertation, we have focused on decision support systems of PdAM,

in the form of optimization frameworks. These frameworks provide substantial support for the implementation of PdAM in practice. Even so, it remains future work to build users' trust in PdAM, to integrate it into strict aviation legislation, and to adopt PdAM at the business level. The strongest support for trust, legislation, and business regarding PdAM should be based on mathematical models and optimization frameworks. Therefore, this dissertation is a starting point for an informed discussion on the future of predictive aircraft maintenance.

ABBREVIATIONS

ABM	Agent-based Model
AC	Aircraft
ACMS	Aircraft Condition Monitoring System
ACS	Air Conditioning System
AHM	Aircraft Health Monitoring
AMP	Aircraft Maintenance Program
BLR	Bayesian Linear Regression
CBM	Condition-based Maintenance
CDF	Cumulative Distribution Function
C.I.	Confidence Interval
C-MAPSS	Commercial Modular Aero-Propulsion System Simulation
CNN	Convolutional Neural Network
CR	Flight crew
DCNN	Deep Convolutional Neural Network
DM	Data Management team
DRL	Deep Reinforcement Learning
DY	Day
EASA	European Union Aviation Safety Agency
EGO	Efficient Global Optimization
ELSA	Exploring design space using Gaussian process Learning and adaptive Sampling
FC	Flight Cycle
FD	Factorial Design
FH	Flight Hour
FII	Fixed Interval Inspection strategy
FIR	Fixed Interval Replacement strategy
FMEA	Failure Mode and Effects Analysis
GP	Gaussian Process
GS	Ground Station
HAZOP	Hazard and Operability Study
HUMS	Health and Usage Monitoring System
ILP	Integer Linear Programming
IPN	Interaction Petri Nets
KPI	Key Performance Indicator
KS test	Kolmogorov–Smirnov test
LPN	Local Petri Net
MBE	Mean-bias-error
MC	Monte Carlo

MCTR	Mean-Cycles-to-Replacement
ME	Mechanics team
MEL	Minimum Equipment List
MLE	Maximum Likelihood Estimation
MOC	Maintenance Operation Center
MSG	Maintenance Steering Group
NSGA-II	Non-Dominated Sorting Genetic Algorithm-II
OM	Opportunistic Maintenance
PdM	Predictive Maintenance
PdAM	Predictive Aircraft Maintenance
PM	Preventive Maintenance
RBF	Radial-basis-function
RBR	RUL-based Replacement strategy
ReSPIR	Response Surface-based Pareto Iterative Refinement
RMSE	Root-mean-squared-error
RUL	Remaining-Useful-Life
SBI	Sensor-based Inspection strategy
SBR	Sensor-based Replacement strategy
SDCPN	Stochastically and Dynamically Colored Petri Net
TBM	Time-based Maintenance
TG	Task Generating team
TP	Task Planning team
TRPCS	Tail Rotor Pitch Change Shaft
VII	Variable Interval Inspection strategy
WN	White Noise

CONTENTS

Acknowledgements	vii
Summary	ix
Abbreviations	xiii
1 Introduction	1
1.1 Research Context	1
1.2 Research Gaps	2
1.3 Research Objectives.	4
1.4 Research Methodology	4
1.5 Overview of Dissertation	7
2 Stochastic Petri Nets to Model Predictive Aircraft Maintenance	11
2.1 Introduction	12
2.2 Predictive Aircraft Maintenance Process:Agent-based Modeling Approach .	13
2.3 Formalization of the Agent-based Model of Predictive Aircraft Maintenance Process using Stochastic Petri Nets	17
2.3.1 Stochastically and dynamically colored Petri nets	17
2.3.2 Formalization of the agents using SDCPNs.	20
2.3.3 Assessment of maintenance strategies using simulation of agent- based model	29
2.4 Assessment of Maintenance Strategies for Aircraft Landing Gear Brakes . .	29
2.4.1 Problem description	30
2.4.2 Degradation model of the aircraft landing gear brakes	32
2.4.3 Maintenance strategies for aircraft landing gear brakes	34
2.4.4 Safety and efficiency indicators of maintenance for aircraft landing gear brakes.	37
2.4.5 Estimation of the model parameters	39
2.4.6 Monte Carlo simulation results	41
2.4.7 Discussion	47
2.5 Conclusions.	49
3 Multi-Objective Analysis of Predictive Aircraft Maintenance	53
3.1 Introduction	54
3.2 Methodology	54
3.2.1 Multi-component aircraft maintenance model	54
3.2.2 Aircraft maintenance strategies and parameters	57
3.2.3 Multiple objectives of aircraft maintenance	58
3.2.4 Discrete event simulation of aircraft maintenance	59

3.3	Simulation Results: Multi-Objective Analysis	59
3.3.1	Relation between multiple objectives	59
3.3.2	Trade-off between aircraft maintenance objectives	61
3.4	Conclusion	63
4	Predictive Aircraft Maintenance at Component Level using RUL Prognostics and Deep Reinforcement Learning	67
4.1	Introduction	68
4.2	Estimating the Distribution of RUL using CNN with Monte Carlo dropout	70
4.2.1	Data description and pre-processing.	70
4.2.2	Architecture of the multi-channel CNN with Monte Carlo dropout	71
4.2.3	Results: Probabilistic RUL prognostics for turbofan engines	73
4.3	Planning Predictive Aircraft Maintenance using Deep Reinforcement Learning and Probabilistic RUL Prognostics	77
4.3.1	Scheduling engine replacements considering updated probabilistic RUL prognostics	77
4.3.2	Formulating predictive maintenance planning as a deep reinforcement learning problem	79
4.3.3	Training DRL agent for predictive maintenance planning	81
4.4	Results: DRL for predictive aircraft maintenance with probabilistic RUL prognostics	86
4.4.1	Training DRL agent	86
4.4.2	Predictive maintenance using DRL.	86
4.5	Predictive Aircraft Maintenance using DRL vs. Other Maintenance Strategies	91
4.6	Conclusions.	92
5	Predictive Aircraft Maintenance at Fleet Level using RUL Prognostics and Integer Linear Programming	97
5.1	Introduction	98
5.2	RUL Prognostics for Aircraft Landing Gear Brakes.	99
5.2.1	Maintenance of aircraft landing gear brakes	99
5.2.2	Condition monitoring of aircraft landing gear brakes	99
5.2.3	RUL prognostics of aircraft landing gear brakes	99
5.2.4	Performance of the RUL prognostics.	101
5.3	Integration of RUL Prognostics into Opportunistic Maintenance Scheduling	102
5.3.1	Problem description	102
5.3.2	Rolling horizon for RUL-driven OM	103
5.3.3	Integer Linear Programming of RUL-driven OM	104
5.4	Results: RUL-driven Opportunistic Maintenance of Aircraft Landing Gear Brakes.	106
5.4.1	Illustration of RUL-driven OM strategy.	106
5.4.2	Benchmarks: traditional maintenance strategies.	107
5.4.3	RUL-driven OM vs. benchmark maintenance strategies	107
5.5	Conclusion	109

6	Multi-Objective Optimization of Predictive Aircraft Maintenance at Strategy Level using Gaussian Process Learning	113
6.1	Introduction	114
6.2	Problem Formulation: Multi-objective Design of Aircraft Maintenance	117
6.3	Framework for multi-objective design of aircraft maintenance	118
6.3.1	Aircraft Maintenance Model	119
6.3.2	The design space of aircraft maintenance: strategy types and associated design variables	122
6.3.3	Multiple objectives of aircraft maintenance	126
6.3.4	Crude Monte Carlo simulation of maintenance designs and selection of conflicting objectives	127
6.3.5	An adaptive algorithm for multi-objective design space exploration of aircraft maintenance	129
6.4	Case Study I: Designing Maintenance for Landing Gear Brakes	136
6.4.1	Model parameters	136
6.4.2	Pareto front of aircraft maintenance designs	137
6.4.3	Selecting reliable and cost-efficient aircraft maintenance designs	139
6.5	Quality of the Pareto front	143
6.5.1	The quality of the pre-estimations made by the GP models	143
6.5.2	The quality of the Pareto front obtained using ELSA	144
6.5.3	Performance of ELSA vs. other algorithms	144
6.6	Case Study II: Predictive Maintenance for Landing Gear Brakes	147
6.6.1	Probabilistic RUL prognostics for landing gear brakes	148
6.6.2	Predictive maintenance using probabilistic RUL prognostics	148
6.6.3	Model parameters	148
6.6.4	Generating Pareto front of predictive maintenance designs	149
6.7	Conclusion	151
7	Emerging Challenges of Predictive Aircraft Maintenance	159
7.1	Introduction	160
7.2	Process Identification of Data-driven Predictive Aircraft Maintenance	161
7.3	Hazard Identification for Data-driven Predictive Aircraft Maintenance	164
7.3.1	Methodology	164
7.3.2	Analysis of brainstorming results	166
7.4	Validation of the Identified Hazards using Reported Aircraft Incidents	170
7.5	Emerging Challenges of Predictive Aircraft Maintenance	175
7.6	Conclusions	177
8	Conclusion	181
8.1	Review of Research Objectives	181
8.2	Conclusions	183
8.3	Novelty and Scientific Contributions	184
8.4	Limitations and Recommendation for Future Work	186
	Curriculum Vitae	189
	List of Publications	191

1

INTRODUCTION

1.1. RESEARCH CONTEXT

Aircraft maintenance is crucial for safe and reliable flights. To ensure its safety and reliability, strict regulations and manuals guide the aircraft maintenance program (AMP), which is a list of tasks [1]. Most tasks of the AMP are inspections of aircraft components, which need to be performed at fixed time intervals. For instance, the brakes of aircraft landing gears must be checked after every N flight cycles. The engines must be inspected after every T flight hours. Based on the inspection results, further tasks such as lubrication, servicing, restoration, or replacement are performed. This maintenance approach, relying on the tasks being repeated at fixed time intervals, is referred to as time-based maintenance (TBM).

Under time-based aircraft maintenance, the time intervals between tasks are often short to timely detect potential failures. For example, the inspection interval of landing gear brakes (around 30-50 flight cycles) is generally much shorter than the average life of the brakes (around 1,000-1,200 flight cycles) [2]. Although such frequent inspections effectively identify potential failures, the associated maintenance cost is high as aircraft are frequently grounded for inspections. Moreover, most inspections do not find any fault, leading to no further tasks. In fact, around 90% of the scheduled maintenance results in no-fault-found [3]. In other words, a large portion of the maintenance cost is spent to ground the aircraft and perform inspections, but nothing is actually repaired. Therefore, it is of interest to study whether maintenance costs can be reduced.

Current aircraft maintenance can be more efficient if aircraft components are continuously monitored and if their failures can be predicted. Nowadays, new aircraft are equipped with aircraft health monitoring (AHM) systems that continuously monitor the condition of aircraft components without the need to ground the aircraft and perform inspections [4]. For example, in the case of a Boeing 787, the remaining thickness of landing gear brakes is automatically measured every flight cycle [5]. For more sophisticated aircraft components, such as the aircraft engines, on-board sensors generate around 20 terabytes of condition data per flight hour [6]. The large amount of condition data are analyzed to predict the expected time left until component failure, the so-called

Remaining-Useful-Life (RUL) [7]. The predicted RULs of aircraft components and the increasing availability of condition data provide experts with insights into the degradation of these components and support them in making decisions about aircraft maintenance. This new maintenance approach using AHM systems and RUL prognostics is called *predictive maintenance* [8]. Predictive aircraft maintenance (PdAM) is expected to contribute to increase both reliability and efficiency by reducing inspections, predicting potential failures, and planning more efficient maintenance [7].

However, in practice, PdAM is not fully implemented. Current regulations only allow using condition monitoring data and RUL prognostics results for a few limited maintenance actions [1]. For example, the landing gear brakes of Boeing 787 are still replaced based on visual inspections performed every N flight cycles, without using the sensor measurements of the brakes. Only recently, regulatory bodies and aviation industries initiated discussions on integrating AHM systems into aircraft maintenance [9]. Given this context, the main goal of this dissertation is to support the transition from current time-based aircraft maintenance to predictive aircraft maintenance.

1.2. RESEARCH GAPS

Whereas predictive maintenance for aircraft is a relatively new concept, the basic idea of using RUL prognostics for maintenance planning has been applied to other systems such as batteries, rolling machines, wind turbines, and trains. Here, predictive maintenance has significantly reduced system downtime and safety incidents [10, 11]. However, the potential benefits of using RUL prognostics for aircraft maintenance has been validated only for a few cases. Therefore, further research is required to address the following two research gaps.

UNDERSTANDING THE PROCESS OF PREDICTIVE AIRCRAFT MAINTENANCE

First, a clear understanding of the process of predictive aircraft maintenance (PdAM) is needed. PdAM cannot be tested in a real environment due to strict regulations and high costs. Thus, mathematical models are needed to model and quantify the performance of PdAM [12]. Such a mathematical model of PdAM should consider state-of-the-art RUL prognostics algorithms [13] and the interactions of multiple humans and machines involved in the aircraft maintenance [14]. Also, the model should be generic and extendable to facilitate the implementation of PdAM for various components/systems. Although some studies have proposed aircraft maintenance models [12, 13, 14], there is a lack of models to assess the performance of various advanced PdAM approaches.

Moreover, multiple stakeholders of aircraft maintenance, such as airlines, regulatory bodies and manufacturers, consider different key performance indicators (KPIs) for PdAM. These KPIs often conflict with each other, and a deterioration of one KPI might be necessary to improve another KPI. For instance, typical KPIs of landing gear brake maintenance are maintenance costs and the number of brake failures. If the brakes are inspected more frequently, the cost increases, but the probability of having a brake failure decreases. This shows a clear trade-off between cost and reliability. However, many existing studies assume cost as a single objective [15, 16]. Thus, a multi-objective analysis identifying the KPIs and their trade-offs, is a prerequisite to implementing PdAM that

properly reflects the interests of all stakeholders.

INTEGRATING RUL PROGNOSTICS INTO MAINTENANCE OPTIMIZATION

The second research gap is the lack of optimization frameworks to plan PdAM based on RUL prognostics results. In the last decades, academia and industry have focused on data collection using AHM systems and the development of data-driven RUL prognostics algorithms [5, 6, 7]. However, not many studies have considered how to actually plan aircraft maintenance using these advanced RUL prognostics. A few studies that optimize predictive maintenance planning assume simple degradation trends, such as Wiener processes [17] and Poisson processes [18]. To utilize the full potential of RUL prognostics, decision support frameworks are needed to plan maintenance by considering RUL prognostics.

RUL prognostics can be used for aircraft maintenance planning at various levels. In the simplest case, the RUL of a single aircraft component can be used to determine a deadline to replace it, i.e., *component-level* PdAM. Here, it is important to note that RUL prognostics are subjected to uncertainty originating from models and data [19]. Thus, setting a deadline exactly at the predicted RUL can be deceptive; instead, the associated uncertainty should be considered in component-level PdAM.

At the next level, maintenance is simultaneously planned for multiple components and multiple aircraft [13], i.e., *fleet-level* PdAM. At the fleet level, PdAM should be planned considering operational requirements such as flight schedules and the availability of the maintenance hangar [13, 20]. These operational requirements, together with the deadlines of tasks, make the fleet-level PdAM a complex optimization problem with multiple constraints.

Ultimately, the current aircraft maintenance program (AMP) needs to be updated to predictive maintenance strategies, i.e., *strategy-level* PdAM. To switch to PdAM, new parameters need to be optimized, such as thresholds of RUL and safety margins. For instance, under PdAM, a component is replaced when its RUL is smaller than a threshold A , or a component is repaired when its failure probability is larger than a safety margin B . Optimizing such parameters for multiple objectives (KPIs) is computationally intensive. Thus, an efficient multi-objective optimization algorithm is required to optimize the new parameters of PdAM strategies.

IDENTIFYING EMERGING CHALLENGES

In addition, PdAM adopts new technologies (e.g., AHM systems, and data-driven RUL prognostics algorithms) and changes some roles of maintenance stakeholders, which also introduces emerging challenges and risks. Existing studies discussing the challenges and risks of aircraft maintenance are outdated [21, 22], as these were performed before the first discussion on the introduction of PdAM in 2018 [9]. Hence, the emerging challenges of PdAM should be investigated based on updated PdAM models.

1.3. RESEARCH OBJECTIVES

From the [Research Gaps](#) the following research objectives are derived:

- Obj.1** Construct mathematical models of predictive aircraft maintenance and assess its performance.
- Obj.2** Identify the key performance indicators (KPIs) of predictive aircraft maintenance, and analyze their trade-offs.
- Obj.3** Integrate RUL prognostics into predictive aircraft maintenance planning.

Objective **Obj.3** considers the model of PdAM constructed in **Obj.1** and the KPIs identified in **Obj.2**. **Obj.3** is further divided into three levels, namely component-level, fleet-level, and strategy-level PdAM.

- Obj.3.1** Optimize a predictive maintenance plan for an aircraft component based on RUL prognostics and associated uncertainty (Component-level PdAM).
- Obj.3.2** Optimize a predictive maintenance plan for a fleet of aircraft with multiple components, considering operational requirements (Fleet-level PdAM).
- Obj.3.3** Design a predictive maintenance strategy by optimizing parameters such as thresholds of RUL, considering multiple objectives (Strategy-level PdAM).

The last research objective is to identify emerging challenges based on the updated models of PdAM:

- Obj.4** Identify emerging challenges of predictive aircraft maintenance.

1.4. RESEARCH METHODOLOGY

The [Research Objectives](#) are addressed following the research methodology illustrated in Figure 1.1. It consists of three phases.

PHASE I: MODELING THE PROCESS OF PdAM

Phase I addresses research objectives **Obj.1** and **Obj.2**. First, a formal model of generic process of predictive aircraft maintenance (PdAM) is constructed (**Obj.1**). The aircraft maintenance process involves complex interactions between multiple agents who monitor and analyze the health condition of components to make decisions on maintenance tasks. These agents and their interactions are identified through a literature survey including regulations and aircraft maintenance manuals. The monitoring of health conditions of components and their degradation processes are modeled by stochastic processes, which is validated using actual condition monitoring data. The flow of information and decision making procedures are modeled using Petri nets. The final model is validated through interviews with field experts from airlines, aircraft maintenance, and regulatory bodies.

Next, the key performance indicators (KPIs) of PdAM are identified (**Obj.2**). To optimize PdAM, it is essential to set objectives that represent the perspectives of multiple

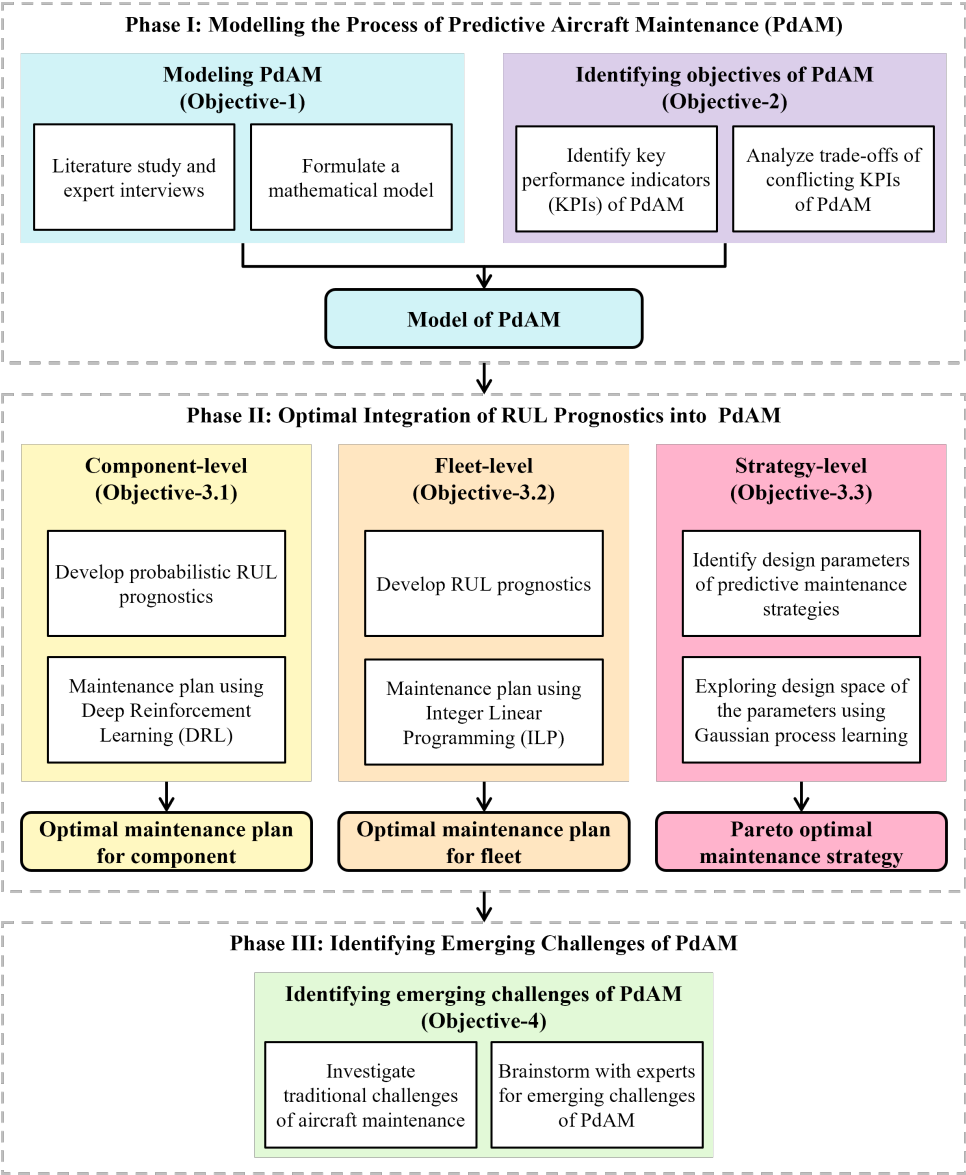


Figure 1.1: Research methodology of the dissertation.

stakeholders involved in PdAM. Based on the literature study and expert interviews, the KPIs of the stakeholders of aircraft maintenance are identified. Using the model of PdAM developed in **Obj.1**, the correlations and the trade-offs between these KPIs are analyzed. This preliminary analysis derives a set of objectives of PdAM, which will be used in the next phase of this research.

PHASE II: OPTIMAL INTEGRATION OF RUL PROGNOSTICS INTO PdAM

Phase II addresses objective **Obj.3**, the optimization of PdAM. First, the component-level PdAM is considered (**Obj.3.1**). In particular, the predictive maintenance of an aircraft engine is considered since it is a crucial component for safety. For these engines, data-driven RUL prognostics are developed to estimate the probability distribution of RUL (probabilistic RUL prognostics). Unlike a point-estimation of RUL, the estimated RUL distribution contains the information of uncertainty, which is not straightforward to interpret for humans. Thus, a deep reinforcement learning approach is introduced to plan predictive maintenance based on the estimated RUL distributions. As a result, a framework to optimize component-level PdAM considering probabilistic RUL prognostics is proposed.

Second, the fleet-level PdAM is considered (**Obj.3.2**). The predictive maintenance is planned simultaneously considering a fleet of aircraft, where each aircraft is equipped with 8 brakes. The RULs of the brakes are estimated using Bayesian regression based on the sensor monitoring data. These RUL prognostics, flight schedules, and the limited hangar availability are formulated into integer linear programming (ILP). By solving the ILP problem, optimal maintenance is planned for a fleet of aircraft with multiple components.

Finally, the optimization of PdAM is considered at the strategy-level (**Obj.3.3**). A predictive maintenance strategy is defined by a set of design parameters such as thresholds of RUL and safety margins. These parameters need to be optimized considering multiple objectives. An efficient algorithm is proposed to explore the high dimensional space of the design parameters of PdAM strategies. Here, Gaussian process (GP) learning models are used to adaptively sample new design parameters and to identify the Pareto optimal design parameters. This novel algorithm is illustrated using two case studies on predictive maintenance for landing gear brakes. The first case study shows the benefit of predictive maintenance compared to other traditional maintenance strategies, including time-based maintenance. The second case study optimizes a PdM strategy that integrates probabilistic RUL prognostics.

PHASE III: IDENTIFYING EMERGING CHALLENGES OF PdAM

In Phase III, the emerging challenges of PdAM are identified (**Obj.4**). Since predictive maintenance has been introduced fairly recently and only for a few tasks, operational data to identify the challenges of PdAM are not enough. Thus, a structured brainstorming is conducted with maintenance experts to reveal emerging challenges for PdAM [23]. Here, the model of PdAM developed for **Obj.1** is used to facilitate and guide the brainstorming. The brainstorming results and the lessons from past accidents/incidents are analyzed together to identify key challenges for future PdAM.

1.5. OVERVIEW OF DISSERTATION

For ease of navigation, this dissertation is divided into 8 chapters. The overview of the dissertation is given in Figure 1.2. Chapters 2 and 3 correspond to Phase I of the research methodology. Chapters 4 to 6 address Phase II of the research methodology, and form the core of this dissertation. Chapter 4 optimizes the PdAM at the component level, considering the maintenance of an aircraft engine. Chapter 5 optimizes the PdAM at the fleet level, considering the maintenance of landing gear brakes for a fleet of aircraft. Chapter 6 designs Pareto optimal PdAM strategies considering multiple objectives. Phase III of the research methodology is covered in Chapter 7. Finally, Chapter 8 reviews the findings and novelty of this dissertation and provides an outlook on future work.

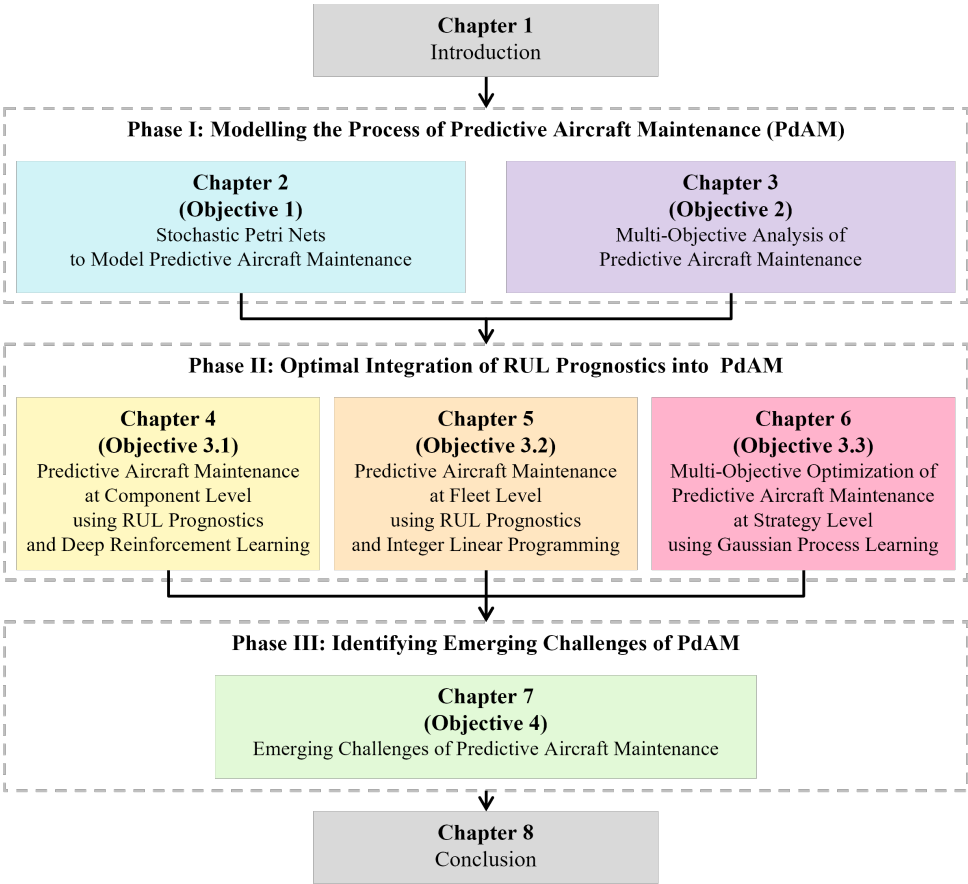


Figure 1.2: Overview of the dissertation.

REFERENCES

- [1] Air Transportation Association. *ATA MSG-3, Operator/Manufacturer Scheduled Maintenance Volume 1 - Fixed Wing Aircraft*. Vol. 1 - Fixed. Air Transport Association of America, 2013.
- [2] J. Lee and M. Mitici. “An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets”. In: *Reliability Engineering and System Safety* 202 (2020), p. 107052. ISSN: 0951-8320. DOI: [10.1016/j.ress.2020.107052](https://doi.org/10.1016/j.ress.2020.107052).
- [3] O. Weiss. “Maintenance of Tomorrow: The AHM path from Airbus’ Perspective”. In: *5th Paperless Aircraft Operations and RFID Conference*. 2018.
- [4] J. B. Maggiore. “Remote Management of Real-Time Airplane Data”. In: *Boeing AERO* (2007), pp. 22–27.
- [5] J. Hale. “Boeing 787 from the Ground Up”. In: *Aero* (2006), pp. 17–23.
- [6] V. E. Badea, A. Zamfiroiu, and R. Boncea. “Big Data in the Aerospace Industry”. In: *Informatica Economica* 22.1/2018 (2018), pp. 17–24. ISSN: 14531305. DOI: [10.12948/issn14531305/22.1.2018.02](https://doi.org/10.12948/issn14531305/22.1.2018.02).
- [7] J. Sprong, X. Jiang, and H. Polinder. “Deployment of Prognostics to Optimize Aircraft Maintenance – A Literature Review”. In: *Journal of International Business Research and Marketing* 5.4 (2020), pp. 26–37. ISSN: 18498558. DOI: [10.18775/jibrm.1849-8558.2015.54.3004](https://doi.org/10.18775/jibrm.1849-8558.2015.54.3004).
- [8] T. Zonta, C. A. da Costa, R. da Rosa Righi, M. J. de Lima, E. S. da Trindade, and G. P. Li. “Predictive maintenance in the Industry 4.0: A systematic literature review”. In: *Computers and Industrial Engineering* 150.April 2019 (2020), p. 106889. ISSN: 03608352. DOI: [10.1016/j.cie.2020.106889](https://doi.org/10.1016/j.cie.2020.106889).
- [9] International Maintenance Review Board Policy Board (IMRBPB). *Aircraft Health Monitoring (AHM) integration in MSG-3, IP180*. 2018.
- [10] Y. Hu, X. Miao, Y. Si, E. Pan, and E. Zio. “Prognostics and health management: A review from the perspectives of design, development and decision”. In: *Reliability Engineering and System Safety* 217.May 2021 (2022), p. 108063. ISSN: 09518320. DOI: [10.1016/j.ress.2021.108063](https://doi.org/10.1016/j.ress.2021.108063).
- [11] W. Tiddens, J. Braaksma, and T. Tinga. “Exploring predictive maintenance applications in industry”. In: *Journal of Quality in Maintenance Engineering* (2020). ISSN: 13552511. DOI: [10.1108/JQME-05-2020-0029](https://doi.org/10.1108/JQME-05-2020-0029).
- [12] A. Regattieri, A. Giazzi, M. Gamberi, and R. Gamberini. “An innovative method to optimize the maintenance policies in an aircraft: General framework and case study”. In: *Journal of Air Transport Management* 44-45 (2015), pp. 8–20. ISSN: 09696997. DOI: [10.1016/j.jairtraman.2015.02.001](https://doi.org/10.1016/j.jairtraman.2015.02.001).
- [13] L. Lin, B. Luo, and S. S. Zhong. “Multi-objective decision-making model based on CBM for an aircraft fleet with reliability constraint”. In: *International Journal of Production Research* 56.14 (2018), pp. 4831–4848. ISSN: 1366588X. DOI: [10.1080/00207543.2018.1467574](https://doi.org/10.1080/00207543.2018.1467574).

- [14] J. Sheng and D. Prescott. “A coloured Petri net framework for modelling aircraft fleet maintenance”. In: *Reliability Engineering and System Safety*. Vol. 189. Elsevier Ltd, 2019, pp. 67–88. DOI: [10.1016/j.ress.2019.04.004](https://doi.org/10.1016/j.ress.2019.04.004).
- [15] B. Lu, Z. Chen, and X. Zhao. “Data-driven dynamic predictive maintenance for a manufacturing system with quality deterioration and online sensors”. In: *Reliability Engineering and System Safety* 212. March (2021), p. 107628. ISSN: 09518320. DOI: [10.1016/j.ress.2021.107628](https://doi.org/10.1016/j.ress.2021.107628).
- [16] K. T. Nguyen and K. Medjaher. “A new dynamic predictive maintenance framework using deep learning for failure prognostics”. In: *Reliability Engineering and System Safety* 188. September 2018 (2019), pp. 251–262. ISSN: 09518320. DOI: [10.1016/j.ress.2019.03.018](https://doi.org/10.1016/j.ress.2019.03.018).
- [17] Y. Zhang, C. C. de Visser, and Q. P. Chu. “Online aircraft damage case identification and classification for database information retrieval”. In: *AIAA Atmospheric Flight Mechanics Conference, 2018* 0.209999 (2018), pp. 1–12. DOI: [10.2514/6.2018-1020](https://doi.org/10.2514/6.2018-1020).
- [18] N. C. Caballé, I. T. Castro, C. J. Pérez, and J. M. Lanza-Gutiérrez. “A condition-based maintenance of a dependent degradation-threshold-shock model in a system with multiple degradation processes”. In: *Reliability Engineering and System Safety* 134 (2015), pp. 98–109. DOI: [10.1016/j.ress.2014.09.024](https://doi.org/10.1016/j.ress.2014.09.024).
- [19] O. Fink, Q. Wang, M. Svensén, P. Dersin, W. J. Lee, and M. Ducoffe. “Potential, challenges and future directions for deep learning in prognostics and health management applications”. In: *Engineering Applications of Artificial Intelligence* 92. January (2020), p. 103678. ISSN: 09521976. DOI: [10.1016/j.engappai.2020.103678](https://doi.org/10.1016/j.engappai.2020.103678). arXiv: [2005.02144](https://arxiv.org/abs/2005.02144).
- [20] Q. Feng, X. Bi, X. Zhao, Y. Chen, and B. Sun. “Heuristic hybrid game approach for fleet condition-based maintenance planning”. In: *Reliability Engineering and System Safety* 157 (2017), pp. 166–176. ISSN: 09518320. DOI: [10.1016/j.ress.2016.09.005](https://doi.org/10.1016/j.ress.2016.09.005).
- [21] A. Hobbs and A. Williamson. “Associations between errors and contributing factors in aircraft maintenance”. In: *Human Factors* 45.2 (2003), pp. 186–201. ISSN: 00187208. DOI: [10.1518/hfes.45.2.186.27244](https://doi.org/10.1518/hfes.45.2.186.27244).
- [22] Y. H. Chang and Y. C. Wang. “Significant human risk factors in aircraft maintenance technicians”. In: *Safety Science* 48.1 (2010), pp. 54–62. ISSN: 09257535. DOI: [10.1016/j.ssci.2009.05.004](https://doi.org/10.1016/j.ssci.2009.05.004).
- [23] H. A. P. Blom, S. H. Stroeve, and T. Bosse. “Modelling of potential hazards in agent-based safety risk analysis”. In: *Europe Air Traffic Management Research and Development Seminar* (2013).

2

STOCHASTIC PETRI NETS TO MODEL PREDICTIVE AIRCRAFT MAINTENANCE

In this chapter, we model predictive aircraft maintenance (PdAM) and assess its impact on reliability and cost-efficiency. The model of PdAM formulates the interactions between the stakeholders of aircraft maintenance using stochastically and dynamically colored Petri nets (SDCPNs). Using Monte Carlo simulation of this model, key performance indicators (KPIs) showing reliability and cost-efficiency of PdAM are evaluated. As a case study, we consider the maintenance of aircraft landing gear brakes. This case study shows that applying predictive maintenance reduces the number of inspections, while ensuring reliability. The model constructed in this chapter is used throughout this dissertation.

Parts of this chapter have been published in the following research articles:

J. Lee and M. Mitici, “An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets,” *Reliability Engineering and System Safety*, vol. 202, p. 107052, 2020.

J. Lee and M. Mitici, “Predictive aircraft maintenance: modeling and analysis using stochastic Petri nets,” in *Proceedings of the 31st European Safety and Reliability Conference*, pp. 146-153, Angers, France, September 19–23, 2021.

2.1. INTRODUCTION

Aircraft maintenance is crucial for safe and efficient operations of aircraft, and thus, airlines spend almost 9.5% of their operational costs for maintenance [1, 2]. While striving for cost-efficient maintenance, safety remains a priority for aircraft operators. However, attaining safety and efficiency in aircraft maintenance is not straightforward, especially due to the complexity of the maintenance process. Some of the drivers of the maintenance complexity are the large number of stakeholders and the necessary cooperation between them, the inevitable human-machine interaction, the high costs with unscheduled maintenance, the dependency between systems, and the strict and specific maintenance regulations [3, 4, 5, 6].

Given the criticality and complexity of the aircraft maintenance process, stakeholders often make use of conservative maintenance strategies. Here, a *maintenance strategy* implies a set of procedures and rules to follow in order to generate, plan, and execute maintenance tasks. In practice, many maintenance tasks are performed at fixed time intervals, i.e., following a time-based maintenance (TBM) strategy [7, 8]. Under TBM strategies, shorter time intervals of tasks increase the chance to detect severe degradation/failures. Thus, shorter time intervals contribute to safety. On the other hand, shorter time intervals require more frequent maintenance tasks, increasing the cost of maintenance. As such, many studies on TBM optimize the maintenance time intervals [9, 10, 11]. Recently, condition-based maintenance (CBM) strategies have been proposed to further decrease the number of maintenance tasks while preserving safety [12, 13]. CBM strategies specify the moment of maintenance by utilizing component/system condition data collected by sensors. Furthermore, predictive maintenance (PdM) strategies have been proposed to use Remaining-Useful-Life (RUL) prognostics in maintenance planning. In this line, many studies propose optimal CBM/PdM strategies to achieve a minimum maintenance cost [3, 12, 13, 14, 15, 16, 17].

Yet, only a few studies consider the safety of maintenance, and even here the authors use indirect metrics such as high penalties for system failure [3, 14, 15], system availability [16], and reliability constraints [17]. Using such indirect metrics makes it hard to distinguish the safety aspects from the efficiency aspects, especially if the improved efficiency compensates for the reduced safety. A clear distinction between these metrics should be made so that the impact on safety and efficiency can be explicitly quantified.

The simulation of maintenance models provides direct quantification of the safety and efficiency of maintenance strategies. In particular, Monte Carlo simulation can capture the impact of uncertainties involved in maintenance, such as stochastic degradation of components, errors in inspection, etc. Thus, several studies perform Monte Carlo simulation for their maintenance models [18, 19, 20, 21, 22, 23, 24].

Methods generally used to model maintenance systems are Petri nets or agent-based modeling (ABM). Petri nets provide a formal visualization for mathematical models of discrete event systems [25, 26]. This method has been used to model the maintenance of complex systems such as railway [18, 19], bridges [20], wind turbines [21, 22], and a fleet of aircraft [23]. However, these models focus on events and processes, without considering the interplay between stakeholders. Because aircraft maintenance involves multiple stakeholders, their interaction needs to be explicitly considered. ABM is another technique to represent maintenance systems, focusing on the interplay between stake-

holders [27, 28]. ABM has been used to model interactive maintenance systems such as production lines [24], and repair service companies [29]. While ABM is effective in formalizing the interaction between multiple stakeholders, the comprehensibility of ABM can be further improved by graphical representations such as Petri nets. This synergy between Petri nets and ABM is used in other domains such as air traffic management [30, 31], but has not been used in the above studies on maintenance. Thus, the synergy between Petri nets and ABM can be used to achieve a comprehensive multi-agent model of aircraft maintenance.

In this chapter, an integrated framework is proposed to assess the safety and efficiency of various aircraft maintenance strategies. An ABM of an end-to-end aircraft maintenance process is developed, where the main maintenance stakeholders are considered. This ABM is formalized by means of stochastically and dynamically colored Petri nets (SDCPNs). Based on the SDCCPN formalization of the ABM, Monte Carlo simulations are conducted for several maintenance strategies. This framework is illustrated for the maintenance of the aircraft landing gear brakes. Here, the degradation of the brakes is modeled by means of a Gamma process. As maintenance strategies for the brakes, a sensor-driven CBM strategy, a prognostic-driven PdM strategy, and two TBM strategies are proposed. Safety and efficiency indicators for these strategies are evaluated using this framework. Overall, this framework is generic in that it supports a safety and efficiency analysis of various aircraft maintenance strategies. Most importantly, our framework supports the assessment of novel strategies, ahead of their implementation in practice.

The remainder of this chapter is organized as follows. Section 2.2 describes the aircraft maintenance process and identifies the agents involved in this process. Section 2.3 formalizes the agent models by means of SDCCPNs. A brief explanation of SDCCPNs is given first. Then, detailed SDCCPN models for each agent are developed. Section 2.4 presents a case study on the maintenance of aircraft landing gear brakes. Finally, Section 2.5 provides conclusions and recommendations for future work.

2.2. PREDICTIVE AIRCRAFT MAINTENANCE PROCESS: AGENT-BASED MODELING APPROACH

We model the predictive aircraft maintenance process by means of agents and interactions between these agents [27]. An *agent* is defined as an independent entity that makes decisions based on a set of rules, interacts with other agents and has its own goals [27, 28]. We identify the agents considering the following four properties [28]: a) an agent is identifiable, having its own characteristics, decision-making rules, and physical or conceptual boundaries that the others can distinguish (Modularity), b) an agent can independently make decisions to change states and to take actions (Autonomy), c) an agent has states that determine its autonomy and that vary over time (Conditionality), and d) an agent interacts with other agents (Sociality).

Among multiple stakeholders involved in the predictive aircraft maintenance process, we focus on the maintenance organization and the aircraft operator. The maintenance organization is a company that keeps the airworthiness of aircraft by means of maintenance, repair, and overhaul. The aircraft operator is a commercial airline which

flies with the aircraft according to a flight schedule. These two stakeholders are represented by several agents. Considering the four properties of an agent mentioned above, we identify the following six key agents that are representative for a maintenance organization and an aircraft operator:

- i) Aircraft (AC)
- ii) Data Management team (DM)
- iii) Task Generating team (TG)
- iv) Task Planing team (TP)
- v) Mechanics team (ME)
- vi) Flight Crews (CR)

Figure 2.1 shows the agents involved in the predictive aircraft maintenance process and the interactions between them. The boxes denote the agents and the arrows denote the interactions between these agents.

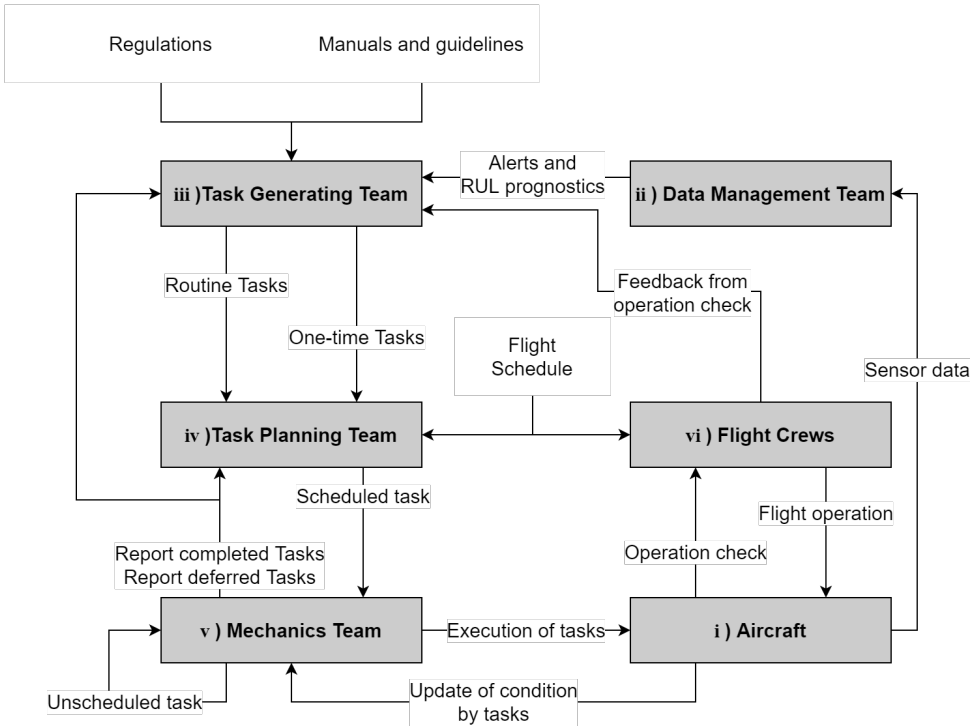


Figure 2.1: Main agents of the predictive aircraft maintenance process.

AIRCRAFT (AC)

Aircraft (AC) is a central agent in the maintenance process, given that the purpose of the aircraft maintenance is to ensure the airworthiness of the aircraft during its operation [1]. Here, we assume that an aircraft operates in terms of flight cycles (see Figure 2.2). A flight cycle is defined as the time period between a departure and the subsequent departure. After the aircraft has departed from a gate at time τ_i^{dep} , we say that the agent AC is in state *in-flight*. *Flight-time* is the period of time between gate departure at time τ_i^{dep} until the arrival time τ_i^{arr} at the gate. When the aircraft stops at the gate, we say that the agent AC is in state *on-ground*. The time between the arrival τ_i^{arr} , and the subsequent departure, τ_{i+1}^{dep} is referred to as *ground-time*. A set of flight cycles is called a flight schedule. The agent AC is operated by flight crews, following a given flight schedule.

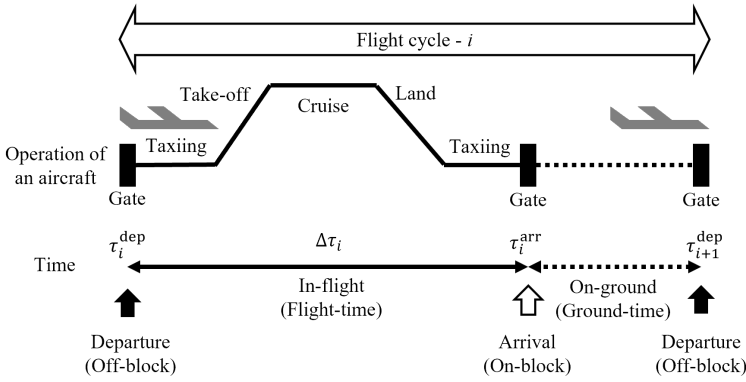


Figure 2.2: Flight cycle of an aircraft.

An aircraft consists of multiple components. These components degrade as the aircraft is in use. When the degradation of a component is significant, malfunctions or failures occur, which renders the aircraft un-airworthy. Airworthiness is sustained by maintenance tasks such as operational checks, inspections, lubrication, restoration, replacement, or discard [1, 32]. Maintenance Steering Group-3, which provides guidelines for aircraft maintenance, suggests four main types of tasks [32]: 1) Inspection, which is the task to find failures or degradation; 2) Lubrication, which is the task that maintains the inherent design capability; 3) Restoration, which is the task to return a system to a specific standard in order to avoid failure, and 4) Replacement, which is the task to discard a component currently in use and install a new one. Inspections, lubrication and restoration tasks are executed by mechanics, while operational checks are executed by flight crews.

Modern aircraft are equipped with sensors that monitor the condition of components [33, 34]. The data collected during condition monitoring is delivered to the data management team for further analysis and maintenance decision support.

DATA MANAGEMENT TEAM (DM)

Data management team (DM) is an agent who playing a key role under a PdM strategy. The agent DM collects the sensor data on the condition of a component, and analyze

these data to estimate the RUL of a component. Considering the estimated RUL and the sensor data, the agent DM alerts the tasks generating team to trigger necessary maintenance tasks.

TASK GENERATING TEAM (TG)

Task generating team (TG) is an agent that specifies which maintenance tasks need to be performed and at which intervals of time. The agent TG takes into account maintenance regulations, manuals and guidelines provided by aircraft manufacturers, as well as feedback from flight crews, mechanics, and sensor data. Using such inputs, the agent TG generates a task by specifying the target component, the type of task, the interval of time at which this task must be executed, and the procedure required to execute the task. The process of task generation reflects the type of maintenance strategy adopted. Under a TBM strategy, the agent TG keeps generating the same tasks at fixed time intervals. Under a PdM strategy, the agent TG generates tasks based on the estimated RUL and sensor data transferred from the agent DM. In this case, the generated tasks are specified with a deadline. The deadlines under a PdM strategy and the intervals under a TBM strategy are specified in the form of flight cycles (FCs), flight hours (FHs), and/or calendar days (DYs). The generated tasks and corresponding maintenance intervals/deadlines are further delivered to the task planning team.

TASK PLANNING TEAM (TP)

Task planning team (TP) is an agent that receives generated tasks from the agent TG and plans these tasks in time. The agent TP receives as input a) the flight schedules, and b) the tasks with their associated intervals/deadlines. Based on the flight schedules, the agent TP evaluates the availability of the aircraft for maintenance. Finally, the agent TP plans maintenance tasks during aircraft ground-time, while making sure that the specified intervals/deadlines for task execution are not exceeded. We refer to this as a *scheduled task*.

MECHANICS TEAM (ME)

Mechanics team (ME) is an agent that executes the scheduled tasks. Once a task is executed, the agent ME may decide whether additional tasks are necessary. If this is the case, the agent ME addresses it immediately by executing *unscheduled tasks*. If the maintenance strategy and the regulations allow, the agent ME can also postpone the execution of additional tasks and just report them to the agent TP or TG.

FLIGHT CREW (CR)

Flight crew (CR) is an agent that operates the aircraft, following a flight schedule. The agent CR checks the condition of the aircraft components before and/or after a flight. We call this activity an *operational check*. If the agent CR observes a component/system failure, then the agent CR reports this to the agent TG.

2.3. FORMALIZATION OF THE AGENT-BASED MODEL OF PREDICTIVE AIRCRAFT MAINTENANCE PROCESS USING STOCHASTIC PETRI NETS

In this section, we formalize the agent models of the aircraft maintenance described in Section 2.2. Stochastically and dynamically colored Petri nets (SDCPNs) are used to graphically model the behavior of the agents [25]. In Section 2.3.1, we introduce the concept of SDCPNs. Then, five agents are modeled by means of SDCPNs in Section 2.3.2. Finally, in Section 2.3.3 we explain how to assess maintenance strategies by means of simulation of agent-based modeling (ABM).

2.3.1. STOCHASTICALLY AND DYNAMICALLY COLORED PETRI NETS

SDCPNs are extension of Petri nets that allow for the modeling of stochastic and dynamic systems [25]. More precisely, SDCPNs are graphs that consist of two sets of nodes: *places* (\mathcal{P}) and *transitions* (\mathcal{T}), as well as a set of *arcs* (\mathcal{A}). These arcs connect the nodes. In addition, SDCPNs may have *tokens* in a place.

Figure 2.3 shows a graphical representation of SDCPN elements. The places represent the possible states of a SDCPN. The location of a token defines the current state of the SDCPN. When additional information is needed to describe the current state, a *color* is assigned to a token. The color of a token can be a continuous or a discrete variable or a set of variables. The locations of the tokens are changed when a transition fires, i.e., a transition updates the status of a SDCPN. We consider three types of transitions. The *immediate transitions* (\mathcal{T}_I) fire immediately if there is at least one token in each *input place* connected by an *incoming ordinary arc* (\mathcal{A}_o) and each *enabling place* connected by an *enabling arc* (\mathcal{A}_e), and there is no token in each *inhibitor place* connected by an *inhibitor arc* (\mathcal{A}_i). Figure 2.4 shows some examples of transitions in a SDCPN. In Figure 2.4, the immediate transitions in (a), (c), and (e) fire immediately. The transition in (b) does not fire because one of its input places has no token. The transition in (d) does not fire because its enabling place has no token. The one in (f) does not fire because its inhibitor place has a token. The *delay transitions* (\mathcal{T}_D) require the same conditions as discussed above, but they fire after a stochastic delay time, as shown in (g) of Figure 2.4. The *guard transitions* (\mathcal{T}_G) fire only if the colors of the tokens in the input places and the enabling places satisfy its guard function (\mathcal{G}). For instance, in (h) and (i) of Figure

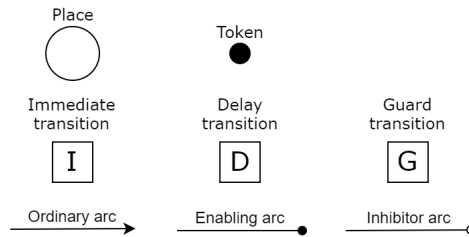


Figure 2.3: Graphical representation of SDCPN elements.

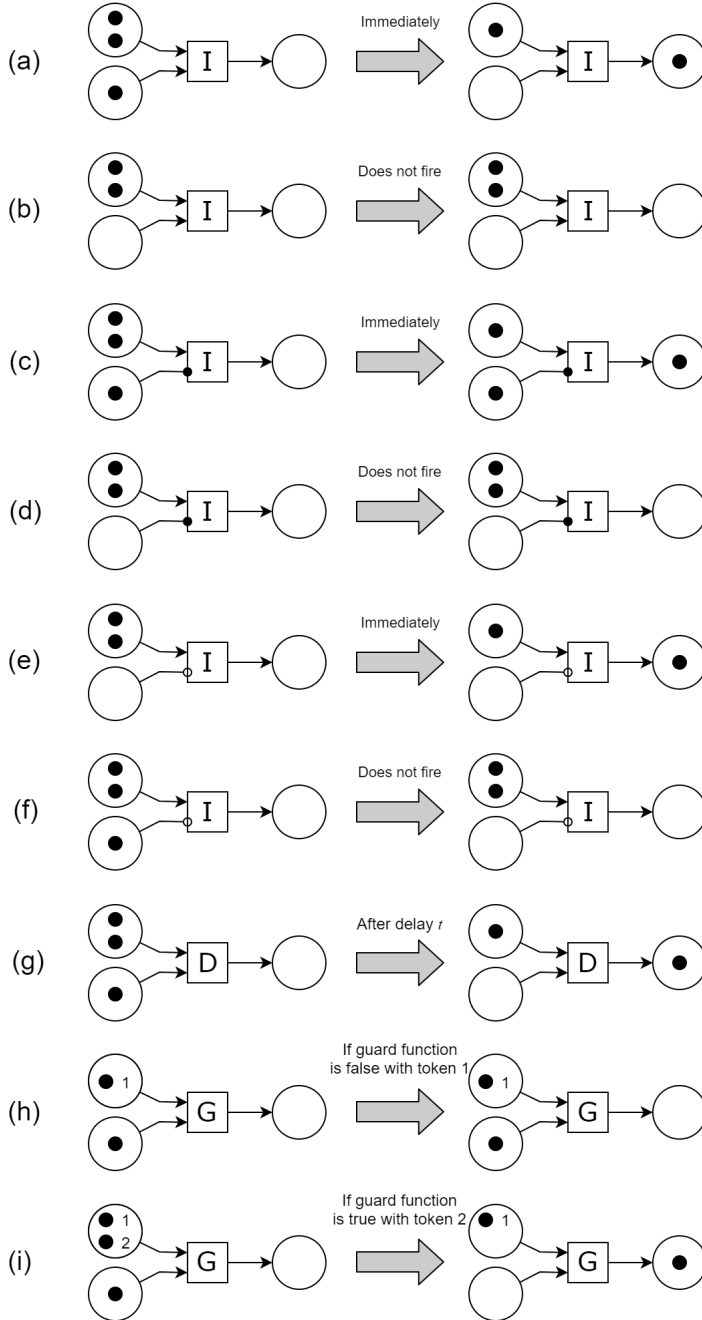


Figure 2.4: Transitions in SDCPN.

2.4, if token 1 renders the guard function false and token 2 renders it true, only the guard transition in (i) fires. When a transition fires, it removes one token from each of its input places, but not from the enabling places (see (a), (c), and (e)). Especially in the case of guard transitions, the token satisfying the guard function is removed (see (i)). Also, a new token is generated in each *output place* that is connected by an *outgoing ordinary arc*. The color of the new token is determined by the *firing function* (\mathcal{F}) of the transition.

In order to make the agent models consistent and comprehensible, we consider the following analogy between an agent and a SDCPN. The possible states of the agents are represented by places. The actions and interactions between agents are represented by transitions and arcs. The places and the transitions needed to model a specific role of an agent are grouped together. This group is called a local Petri net (LPN). A LPN is constructed in such a way that the number of tokens residing in the LPN is not directly changed by another LPN [25]. The interactions between LPNs are modeled by enabling arcs (\mathcal{A}_e) or inhibitor arc (\mathcal{A}_i), which do not change the number of tokens. We also model interactions between LPNs using interaction Petri nets (IPNs), which consist of places and transitions that do not belong to any LPN [25].

As an example, a SDCPN formalization of two agents is given in Figure 2.5. Agent A has two states (places), P-1 and P-2, and it can take two actions (transitions) T-1 and T-2. Agent B has two roles modeled by two LPNs. The guard transition T-1 is fired when the color of the token in P-3 satisfies its guard function, i.e., agent A takes action T-1 if a certain condition of agent B is satisfied. By T-1, the state of agent A becomes P-2. T-3 cannot fire when its inhibitor place P-2 has a token, i.e., agent B cannot take action T-3 while agent A is in P-2. The delay transition T-2 fires after a stochastic delay, returning agent A to P-1. T-2 also fires a token in P-5, and then T-4 immediately fires, which removes the token in P-5 and updates the color of the token in P-4, i.e., agent B immediately takes action T-4 each time agent A takes action T-3. Place P-5 does not belong to any LPN, and it is called an IPN.

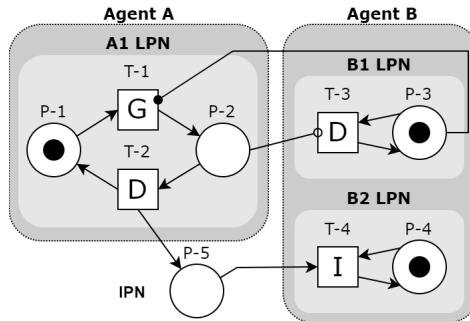


Figure 2.5: Example of SDCPN formalization of two agents.

Table 2.1: LPNs of agents for the aircraft maintenance process

Agent	LPN
i) Aircraft (AC)	Operation Component- ξ Sensor- ξ
ii) Data Management Team (DM)	Prognostics Alert System
iii) Task Generating Team (TG)	Task Generation
iv) Task Planning Team (TP)	Task Planning
v) Mechanics Team (ME)	Task Execution
vi) Flight Crew (CR)	Operation

2.3.2. FORMALIZATION OF THE AGENTS USING SDCPNs

Based on the aforementioned analogy and definition of LPNs, in this section, we model the five agents introduced in Section 2.2 using SDCPNs. Table 2.1 lists the LPNs of each agent considered for the aircraft maintenance process.

AIRCRAFT (AC)

The agent aircraft (AC) is operated by the agent flight crew (CR), following a flight schedule. When the agent CR triggers a departure, the aircraft is pushed back from the gate, i.e., off-block. This changes the state of the agent AC to *in-flight*. The *in-flight* state includes the taxi, take-off, cruise, and landing phases of the operation of an aircraft (see Figure 2.2). When the agent AC arrives at the gate, i.e., on-block, the state of the aircraft from this moment on is *on-ground*.

The LPN **AC Operation** in Figure 2.6 models the operation of the agent AC. Two places In-flight and On-ground represent the two operational states of the aircraft. The transition Off-block changes the state of the aircraft immediately from On-ground to In-flight, when the place Trigger off-block gets a token. This token is generated when the agent CR performs a departure. The transition Off-block also fires a token to the place Use of component- ξ . The color of this token accounts for $u(\Delta\tau)$, which is the amount of time that the component is used during the flight-time $\Delta\tau$. This token triggers the degradation of the component. Depending on the characteristics of the component, $u(\Delta\tau)$ can be represented in different formats. For example, the amount of use of an aircraft engine can be represented as flight-time, i.e., $u(\Delta\tau) = \Delta\tau$. On the other hand, the number of flight cycles better represents the amount of use of aircraft landing gear brakes, i.e., $u(\Delta\tau) = 1$. The transition On-block works in a similar way as the transition Off-block.

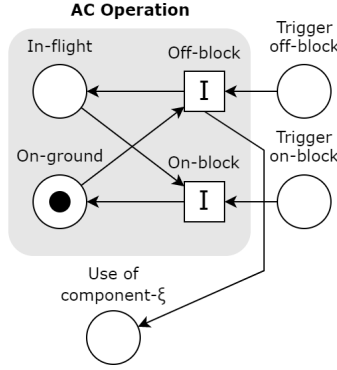


Figure 2.6: LPN: Operation of the agent AC.

When the agent CR completes a flight, the place Trigger on-block gets a token. Then, the transition On-block moves the token from the place In-flight to the place On-ground. We model the degradation of a component as a stochastic process. Here, we assume that aircraft components degrade during in-flight, while the degradation during on-ground is assumed to be negligible. Let $Z(t)$ be the degradation level of a component at time t . Modeling the degradation process $\{Z(t)\}$ should consider different degradation trends for different types of components. However, considering the nature of the degradation, we require the following properties of $\{Z(t)\}$. Firstly, $Z(t) = 0$ if the component is new and has no degradation. Secondly, $Z(t)$ is monotonically increasing unless maintenance is performed, and thus $Z(t) \geq 0$. This is based on the fact that degradation is never recovered spontaneously without maintenance. Thirdly, there is an unacceptable level of degradation, η . If $Z(t) > \eta$, the component is regarded as an unsafe or failed. Finally, we consider the following increment of the degradation process $\{Z(t)\}$:

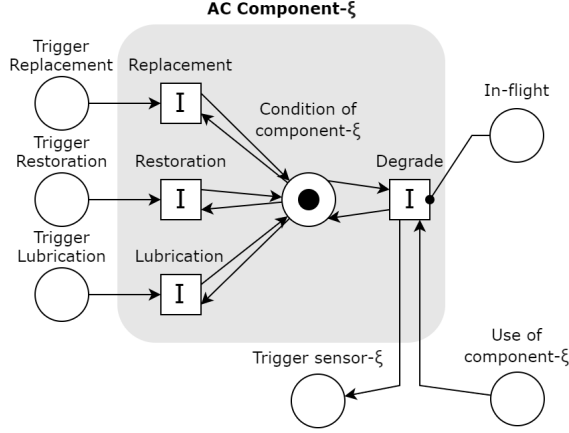
$$Z(t + \Delta t) - Z(t) \sim f_Z(Z; u(\Delta t), \Theta) \quad (2.1)$$

where $\Delta t > 0$, f_Z is the probability density function of the degradation increment, and Θ is the set of parameters of f_Z .

The LPN **AC Component- ξ** in Figure 2.7 models the condition of component- ξ changed by degradation, replacement, restoration and lubrication. The place Condition of component- ξ has a token describing the degradation process of component- ξ , i.e., the token is colored by S_C :

$$S_C = \left(\xi, Z^\xi(t), \Theta^\xi \right) \quad (2.2)$$

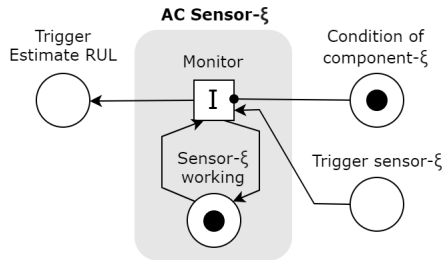
where ξ is the identifier of the component, $Z^\xi(t)$ is the degradation level of component- ξ at time t , and Θ^ξ is a set of parameters describing the degradation process of component- ξ . The transition Degrade fires when the operational state of the aircraft is In-flight and the place Use of component- ξ got a token from the transition Off-block in Figure 2.6. The transition Degrade updates $Z^\xi(t)$ of the color of the token in place Condition of component- ξ , following Equation (2.1). The transition Degrade also fires a token to the place Trigger sensor- ξ , which triggers sensor- ξ to start monitoring the component- ξ .

Figure 2.7: LPN: Component- ξ of the agent AC.

On the other hand, maintenance tasks such as replacement, restoration, and lubrication change the degradation level of the component $Z^\xi(t)$ and/or the trend of the degradation Θ^ξ . After a replacement, $Z^\xi(t)$ is updated to be the degradation level of the new component, Z_{new}^ξ . If the new component is faultless, then $Z_{\text{new}}^\xi = 0$. If the new component already has a level of degradation for some reason, Z_{new}^ξ can be modeled as a constant ($0 \leq Z_{\text{new}}^\xi < 1$) or a random variable with a certain distribution. Restoration tasks update $Z^\xi(t)$ to a specific standard, Z_{res}^ξ , which can be assumed to be a constant or a random variable. We consider lubrication as a task that changes the rate of the degradation process. Thus, lubrication updates Θ^ξ , the parameters of the probability density function in Equation (2.1).

All these tasks are executed by the agent ME, following a maintenance schedule given by the agent TP. In Figure 2.7, the transitions, Replacement, Restoration, and Lubrication fire when there is a token in the places Trigger Replacement, Trigger Restoration, and Trigger Lubrication, respectively. These places get a token when the agent ME executes the corresponding maintenance task on component- ξ . These three transitions update the color S_C of the token in the place Condition of component- ξ .

The LPN **AC Sensor- ξ** in Figure 2.8 models the sensor- ξ that monitors the condition

Figure 2.8: LPN: Sensor- ξ of the agent AC.

of component- ξ . When the sensor is working, the place Sensor- ξ working has a token colored by S_S :

$$S_S = (\xi, \tilde{Z}^\xi(t)), \quad (2.3)$$

where $\tilde{Z}^\xi(t)$ is the degradation level of component- ξ monitored by sensor- ξ . The transition Monitor is triggered by the token in the place Trigger sensor- ξ . Assuming real-time monitoring, the place Trigger sensor- ξ gets a token every time the transition Degrade fires (see Figure 2.7). The token in the place Condition of component- ξ is needed for the transition Monitor. The transition Monitor updates the color S_S of the token in the place Sensor- ξ working. Specifically, $\tilde{Z}^\xi(t)$ is updated as follows:

$$\tilde{Z}^\xi(t + \delta_S) = Z^\xi(t) + \epsilon_S, \quad (2.4)$$

where $\delta_S \sim \text{Exp}(\bar{\delta}_S)$ is the time spent by the sensor to collect the data, and ϵ_S is the measurement error of the sensor. The transition Monitor also fires a token to the place Trigger Estimate RUL, which enables the agent DM to estimate the RUL of component- ξ .

DATA MANAGEMENT TEAM (DM)

Under PdM strategies, the agent DM estimates using condition data. We consider RUL^ξ as the remaining time until the moment when the degradation level of component- ξ reaches a predefined level η [35]. Thus, $RUL^\xi = \min\{t' | Z^\xi(t + t') \geq \eta\}$ where t is the current time and $Z^\xi(t + t')$ is the degradation level after time t' . We estimate $Z^\xi(t + t')$ using prognostics algorithms run on the condition data set $\{\tilde{Z}^\xi(t)\}$.

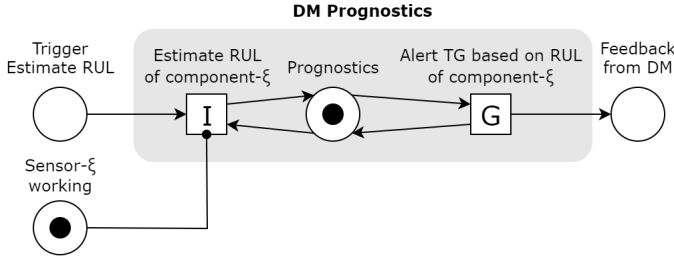


Figure 2.9: LPN: Prognostics of the agent DM.

The LPN model of the DM for RUL prognostics is given in Figure 2.9. The prognostics are triggered by a token in the place Trigger Estimate RUL, which is generated by the transition Monitor of the agent AC (see Figure 2.8). Thus, it is assumed that prognostics are immediately updated each time new data is available. The transition Estimate RUL of component- ξ requires a token colored by S_S on the place Sensor- ξ working. A token in the place Prognostics is colored by S_P defined as:

$$S_P = (RUL^\xi, \{\tilde{Z}^\xi(t)\}) \quad (2.5)$$

The transition Estimate RUL of component- ξ updates RUL^ξ and $\{\tilde{Z}^\xi(t)\}$ based on the given prognostics algorithm. If the estimated RUL^ξ meets a predefined condition, feedback is

generated by the guard transition Alert TG based on RUL of component- ξ . The new token is generated in the place Feedback from DM, which enables the agent TG to generate a new task (see Figure 2.11).

2

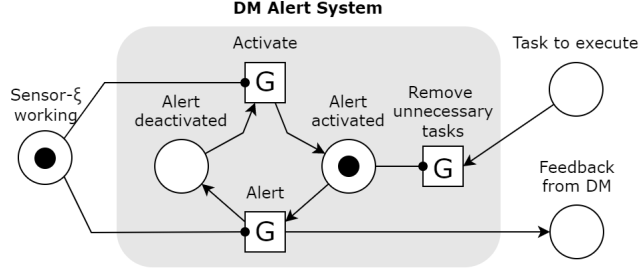


Figure 2.10: LPN: Alert system of the agent DM.

In addition, the agent DM may alert directly based on sensor data. The LPN model of the DM for alert system is given in Figure 2.10. When the alert system is activated, a token is located in the place Alert activated. When $\tilde{Z}(t)$ of the token in the place Sensor- ξ working satisfies the guard function $\mathcal{G}_{\text{Alert}}$, the transition Alert fires a token in the place Feedback from DM. The transition Alert also fires a token from the place Alert activated to the place Alert deactivated, preventing triggering multiple feedback. The guard function $\mathcal{G}_{\text{Alert}}$ of the transition Alert is defined based on the maintenance strategy. For example, the function $\mathcal{G}_{\text{Alert}}(\tilde{Z}(t)) = 1(\tilde{Z}(t) \geq \eta_A)$ is specified for a given strategy, and defines the moment when a new task is generated to prevent degradation.

The transition Activate is fired if its guard function $\mathcal{G}_{\text{Activate}}$ is satisfied. When the place Alert activated has a token, the alert system checks the sensor data. In this case, the transition Remove unnecessary tasks may remove some task tokens that satisfies its guard function, from the place Task to execute. Removal of maintenance tasks is only applicable if the maintenance strategy allows it.

TASK GENERATING TEAM (TG)

The agent Task Generating team (TG) determines the tasks to be performed based on the maintenance strategy and the feedback from other agents. The procedures to consider the feedback are specific to the maintenance strategy. In particular, the maintenance strategy defines the conditions under which a specific type of task needs to be executed, the specific component and the interval to perform this task. In SDCPN formalization, a task is represented as a token colored by \mathcal{S}_T , which is defined as:

$$\mathcal{S}_T = \left(\xi, \nu, \Phi_\nu, d, t^{\text{sch}}, t^{\text{exe}}, i_0 \right) \quad (2.6)$$

where ξ is the target component of the task, ν is the type of the task and Φ_ν is a set of parameters describing the task, d is the interval of the task, t^{sch} and t^{exe} are scheduled time and the actual execution time of the task, and i_0 is the index of the first flight cycle after task execution. The agent TG specifies ξ , ν , Φ_ν , and d , while the other variables will be specified by the agent TP and the agent ME.

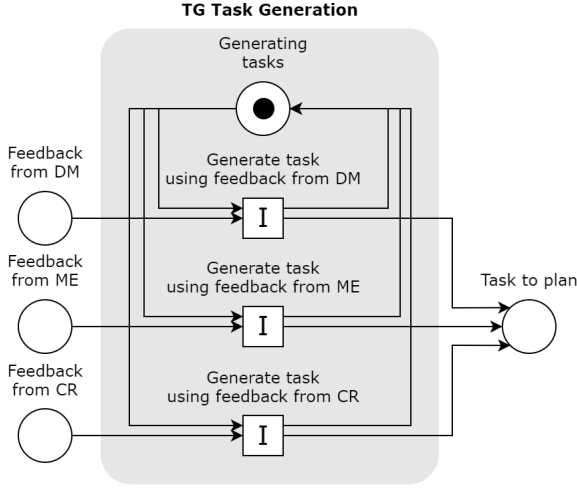


Figure 2.11: LPN: Task generation of the agent TG.

The LPN in Figure 2.11 models how the agent TG generates tasks. A token in the place *Generating tasks* represents that the agent TG is working, and it is required for all transitions in this LPN. The three transitions generate tasks based on RUL prognostics and alerts from the agent DM (see Figures 2.9, 2.10), feedback from the agent ME (see Figure 2.13), and complaints from the agent CR (see Figure 2.14). Firing functions of these transitions determine ξ , ν , Φ_ν , and d of the task token colored by \mathcal{S}_T of Equation (2.6). This new task token is put on the place *Task to plan* and delivered to the agent TP.

TASK PLANNING TEAM (TP)

The agent task planning team (TP) plans the time to execute the tasks. The agent TP takes the input of the agent TG as the time intervals at which tasks must be executed. Another input for the agent TP is the aircraft flight schedule that specifies the ground-time when tasks can be executed. Then, the agent TP finds the latest, feasible time for the tasks to be executed such that the task execution intervals are not exceeded. Formally, this scheduled time t^{sch} is given to the task token colored by \mathcal{S}_T in Equation (2.6).

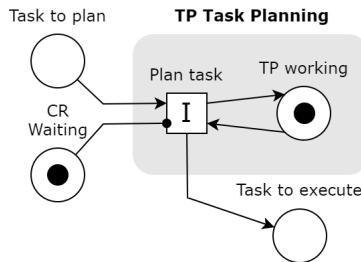


Figure 2.12: LPN: Task planning of the agent TP.

The LPN of the agent TP is shown in Figure 2.12. A token in the place TP working shows that the agent TP is ready to plan a task. The transition Plan task requires a token in the place CR waiting of the agent flight crew. This token has a color representing flight schedules, S_F :

$$S_F = \left(I, \{\tau_i^{\text{dep}}\}_{i \in I}, \{\Delta\tau_i\}_{i \in I} \right) \quad (2.7)$$

where I is a set of index of flights cycles, and $\{\tau_i^{\text{dep}}\}_{i \in I}$ and $\{\Delta\tau_i\}_{i \in I}$ are the set of departure times and the block times of the flight cycles. When a task token colored by S_T is given in the place Task to plan, the transition Plan task fires a task token to the place Task to execute. The firing function of this transition determines t^{sch} the color S_T based on the given task planning algorithm. As a result, the place Task to execute gets a task token with the execution time t^{sch} .

MECHANICS TEAM (ME)

The agent mechanics team (ME) executes the tasks given from the agent TP. When $t \geq t^{\text{sch}}$, the agent ME prepares to execute the task. The agent ME executes a given task when the aircraft is in the state On-ground. Depending on the type of the task, the agent ME inspects, replaces, restores, or lubricates a target component. Especially after the inspection, the agent ME decides whether there an additional unscheduled task is needed. The decision is based the observed degradation level $\hat{Z}^\xi(t)$ and the given maintenance strategy. Such an unscheduled task is executed right away. After completing the task, the agent ME reports to the agent TP.

Figure 2.13 shows the LPN of the agent ME. A token colored by S_T is used in this LPN, representing the task allocated to the agent ME. The token is placed in the place Waiting when there is an available agent ME to execute the given task. This LPN is triggered by the new task token S_T in the place Task to execute, which is generated from the LPN of the agent TP in Figure 2.12. The guard transition Prepare task has the guard function $\mathcal{G}_{\text{Prepare task}} = \mathbb{1}(t \geq t^{\text{sch}})$, which fires the given task token to the place Starting. Depending on the type of the task v , specified in the token color S_T , the relevant task transition fires. For example, if the given task is a replacement, the guard transition Replace fires the token to the place Replacing. Since the aircraft must be available during the task execution, the task-related transitions are enabled by the places On-ground and Condition of component- ξ (see Figure 2.6 and 2.7). When the delay transition Replace fires, meaning that the agent ME completed the task, the task token is fired to the place Completing. Here, the delay $\delta_{\text{rep}} \sim \text{Exp}(\delta_{\text{rep}})$ models the time spent on a replacement. As soon as the delay transition Replace fires, a new token is generated in the place Trigger Replacement. This new token enables the immediate transition Replacement in the LPN of the component in Figure 2.7. The same process is used for the restoration and the lubrication tasks. The transition Report fires the token to the place Waiting, meaning that the agent ME is ready for the next task. If the completed task needs to be repeated later, the transition Report fires the task token to the place Feedback from ME, making the agent TG to generate it again.

For the inspection task, the delay transition Inspect does not fire a token to the trigger places because the inspection does not change the condition of the target component. The color of this token has an additional color variable, i.e., the observed degradation

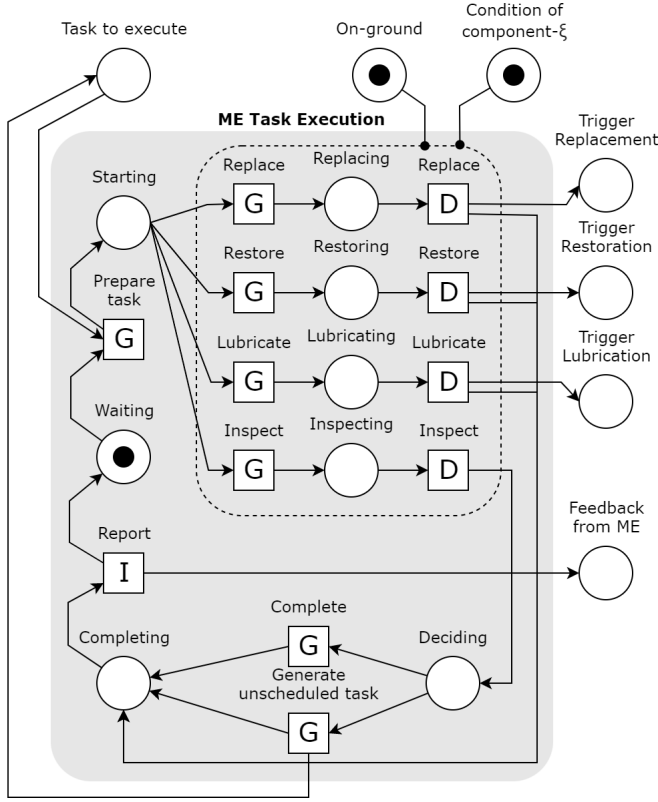


Figure 2.13: LPN: Task execution of the agent ME.

level of the component $\hat{Z}(t)$.

$$\hat{Z}(t + \delta_{\text{ins}}) = Z(t) + \epsilon_{\text{ins}} \quad (2.8)$$

where ϵ_{ins} is the error of the inspection, and $\delta_{\text{ins}} \sim \text{Exp}(\bar{\delta}_{\text{ins}})$ is time spent to inspect the component or the delay of the transition *Inspect*. Then, instead of firing a token to the place *Completing*, the delay transition *Inspect* fires a token in the place *Deciding* which is an intermediate place. Then, the guard transitions *Complete* and *Generate unscheduled task* check $\hat{Z}(t)$. The given maintenance strategy specifies their guard functions. For instance, a restoration can be scheduled if the observed degradation level $\hat{Z}(t)$ is greater than a predefined threshold η_{ins} , i.e., $\mathcal{G}_{\text{Generate unscheduled task}} = \mathbb{1}(\hat{Z}(t) \geq \eta_{\text{ins}})$. Based on the maintenance strategy, the transition *Generate unscheduled task* fires a new task token to the place *Task to execute*. For this task token, $t^{\text{sch}} = t$.

FLIGHT CREW (CR)

The agent flight crew (CR) operates the aircraft based on a flight schedule and conducts operational checks. The agent CR departs at $t \geq \tau_i^{\text{dep}}$, if the aircraft is not under maintenance. After $\Delta\tau_i$ hours of flights, the agent CR arrives at the destination airport. During

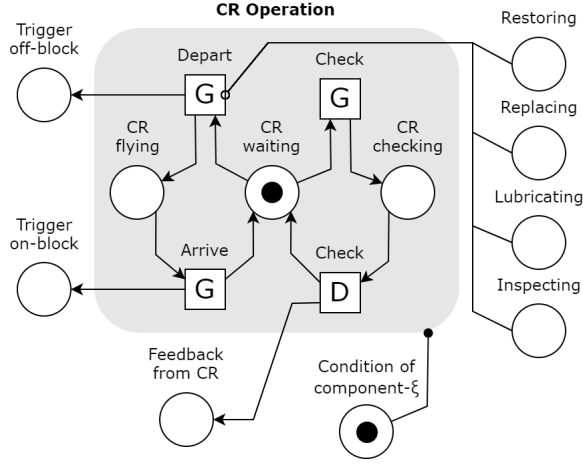


Figure 2.14: LPN: Operation of the agent CR.

the ground-time, they can check the condition of the aircraft components, i.e., operational check. The result is reported to the agent TG, which may generate a new task.

Figure 2.14 shows the LPN of the agent CR. This LPN uses a flight schedule token colored by S_F in Equation (2.7). Initially, the token is placed in the place CR Waiting, meaning that there is an available flight crew. The guard transition Depart has a guard function, $\mathcal{G}_{\text{Depart}} = \mathbb{1}(t \geq \tau_i^{\text{dep}})$ where $i \in I_1$ is the next flight cycle. This is disabled if there is a token in one of the places Restoring, Replacing, Lubricating, and Inspecting, meaning that the departure can be delayed if the agent ME is executing a task at τ_i^{dep} . The transition Depart fires a token in the place Trigger off-block, triggering the transition Off-block of the LPN **AC Operation** in Figure 2.6. It also moves the token from CR Waiting to CR Flying. After completing a flight, the guard transition Arrive fires according to its guard function $\mathcal{G}_{\text{Arrive}} = \mathbb{1}(t \geq \tau_i^{\text{dep}} + \Delta\tau_i)$. A token is fired to the place Trigger on-block by the guard transition Arrive, triggering the transition On-block of the LPN **AC Operation** in Figure 2.6. The transition Arrive also fires a token to CR Waiting.

The agent CR may conduct operational checks for a certain aircraft components depending on the maintenance strategy. As in the case of the inspection of the agent ME, a guard transition Check and a delay transition Check is used for the operational check of the the component condition. The degradation level of the component observed by the agent CR, \hat{Z}_{CR} is updated as below:

$$\hat{Z}_{\text{CR}}(t + \delta_{\text{CR}}) = Z(t) + \epsilon_{\text{CR}}, \quad (2.9)$$

where $\delta_{\text{CR}} \sim \text{Exp}(\bar{\delta}_{\text{CR}})$ is the time spent for the operational check, and ϵ_{CR} is the error in the operational check. The result is reported to the agent TG, when the delay transition Check fires a token to the place Feedback.

2.3.3. ASSESSMENT OF MAINTENANCE STRATEGIES USING SIMULATION OF AGENT-BASED MODEL

With the formalization of the ABM in Section 2.3.2, we assess safety and efficiency indicators of maintenance strategies of interest.

As a first step, we implement the maintenance strategy of interest to the ABM by adjusting the transitions, the initial location of the tokens, and the LPNs. For the delay transitions, the parameters are estimated based on, for instance, maintenance manuals specific to the given maintenance strategy, historical data on the execution of the task, etc. For the guard transitions, the guard functions are also specified based on the given maintenance strategy. For example, for the agent DM, the guard function $\mathcal{G}_{\text{Alert}}$ of the transition Alert, and its parameter η_A are specified based on the given maintenance strategy (see Figure 2.10). Similarly, the firing functions of the transitions of the agent TG also need to be specified based on the maintenance strategy (see Figure 2.11). For instance, if the maintenance strategy requires to replace the component when an alert is triggered by the agent DM, the firing function of the transition Generate task using feedback from DM is set to generate a replacement task token. (see Figure 2.11).

Next, we mark the location of the initial tokens in the LPNs in Section 2.3 (see Figures from 2.7 to 2.14). For the colored tokens, the initial colors are set as follows. The initial degradation level $Z^\xi(0)$ of the component- ξ is represented in the color \mathcal{S}_C of a token in the place Condition of component- ξ (see Figure 2.7). The flight schedule I , $\{\tau_i^{\text{dep}}\}_{i \in I}$ and $\{\Delta \tau_i\}_{i \in I}$ is represented the color \mathcal{S}_F of a token in the place CR Waiting (see Figure 2.14).

Lastly, we can add and/or remove additional LPNs, according to the maintenance strategy. For instance, when we consider a system of multiple aircraft components, we add the LPN in Figure 2.7 to the agent AC. When the given maintenance strategy does not require part of the agents, we remove the unnecessary LPNs. For example, if prognostics are not used under the given maintenance strategy, then we remove the LPN in Figure 2.9 from the agent DM.

Following the adjustment of the ABM according to the given maintenance strategy, we define safety and efficiency indicators to assess this maintenance strategy. Let E be a safety/operations event that we analyze using Monte Carlo simulation. For example, the release of un-airworthy aircraft is considered as a safety event. Similarly, the execution of maintenance task is seen as an operations event. We propose generic safety/efficiency indicators to evaluate the occurrence of the event E as follows. Let $T_E(j)$ be the j^{th} occurrence time of event E . Let $N_E(t)$ be the number of occurrences of event E by time $t > 0$. Then, $P[T_E(j) \leq t]$, and $\mathbb{E}(N_E(t))$ represent the probability to have the event E before time t and the expected number of event E by time t , respectively. These two indicators are estimated by conducting Monte Carlo simulations of the ABM.

2.4. ASSESSMENT OF MAINTENANCE STRATEGIES FOR AIRCRAFT LANDING GEAR BRAKES

In this section, we illustrate the framework proposed in Section 2.2 and Section 2.3 for the maintenance of aircraft landing gear brakes. In Section 2.4.1, we describe the maintenance of aircraft landing gear brakes. In Section 2.4.2, we introduce a degradation model of the brakes. In Section 2.4.3, we describe two TBM strategies derived from prac-

tice and two novel CBM and PdM strategies that we propose. In Section 2.4.4, the safety and efficiency indicators are introduced. In Section 2.4.5, the estimation of the model parameters is discussed. In Section 2.4.6, we present the simulation results. Finally, we discuss the obtained results in Section 2.4.7.

2.4.1. PROBLEM DESCRIPTION

We consider the maintenance of landing gear brakes of a wide-body aircraft (see Figure 2.15). The aircraft is equipped with 8 breaks equally distributed on both sides (see Figure 2.16). Over time, the thickness of a brake disc reduces due to wear [7]. The remaining thickness of a brake disc is measured by a visual inspection (see Figure 2.17a). When the thickness of a brake disc is thinner than a threshold, the brake is replaced (see Figure 2.17b), to ensure aircraft airworthiness.

Currently, the maintenance of the landing gear brakes is performed under time-based maintenance (TBM) strategies [7, 8, 9]. Specifically, two maintenance tasks are used: brake inspections at fixed time intervals and replacements. If, upon an inspection, a certain amount of degradation is observed, a brake replacement is scheduled. In general, the interval of inspection is much shorter than the expected life cycle of the brakes, for safety reasons. As shown in [9], under such a fixed-interval inspection strategy, short intervals reduce the probability to have undesired incidents, but the increased number of inspections leads to additional costs with the maintenance. Also, many of these inspections are redundant as they do not lead to further actions such as replacement. On the other hand, in spite of the frequent brake inspections, the degradation of some brakes can still exceed the desirable threshold. In this chapter, we consider two TBM strategies, with medium and high frequency of inspections.

For a better trade-off between frequent inspections (high costs) and unexpected brake degradation levels, monitoring the condition of the brakes using sensors is considered promising [13, 33, 34, 36]. We propose a condition-based maintenance (CBM) strategy to use sensor measurements in determining the moment of inspections. We also propose a predictive maintenance (PdM) strategy to use brake RUL prognostics to determine the moment of replacement. We compare the CBM and PdM strategies against the medium and high frequency TBM strategies, with respect to safety and efficiency indicators.

For our assessment, we simulate the maintenance process of a wide-body aircraft that is operated according to a flight schedule for a period of 10 years. The flight schedule has n_{FC} flight cycles, where flight cycle i is defined by the moments of departure (τ_i^{dep}) and arrival (τ_i^{arr}), $i \in \{1, \dots, n_{FC}\}$ (see Figure 2.2). We assume that the condition of the aircraft brakes degrades over time according to a stochastic process. We also assume that the maintenance tasks for brakes can be executed in all destination airports. As a safety indicator, we define a brake-related safety incident, and evaluate its frequency. As efficiency indicators, we assess the number of required tasks, associated with these maintenance strategies, and the remained thickness of the brake discs at the moment of replacement.

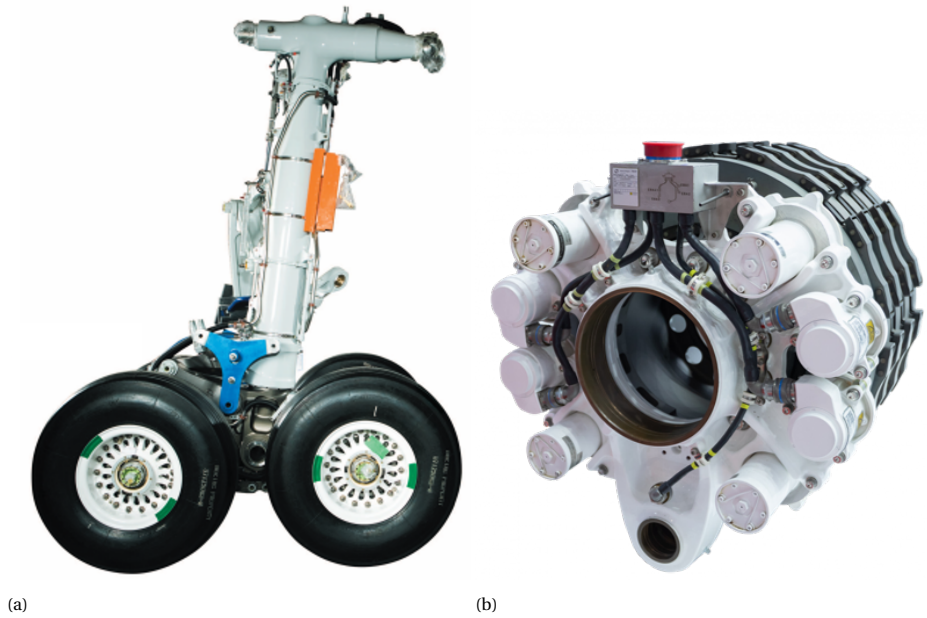


Figure 2.15: (a) A landing gear of Boeing 787 aircraft. (b) A brake of a landing gear.
Image sources: <https://www.safran-group.com/products-services/>

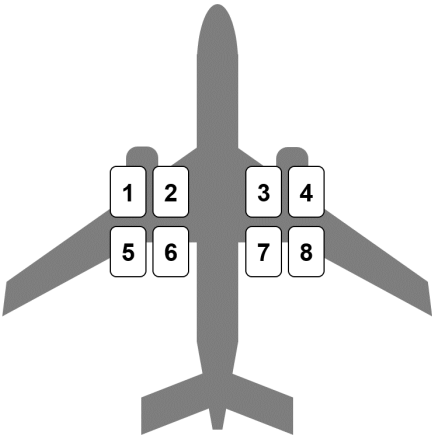


Figure 2.16: Position of the 8 brakes of a wide-body aircraft with their position index.

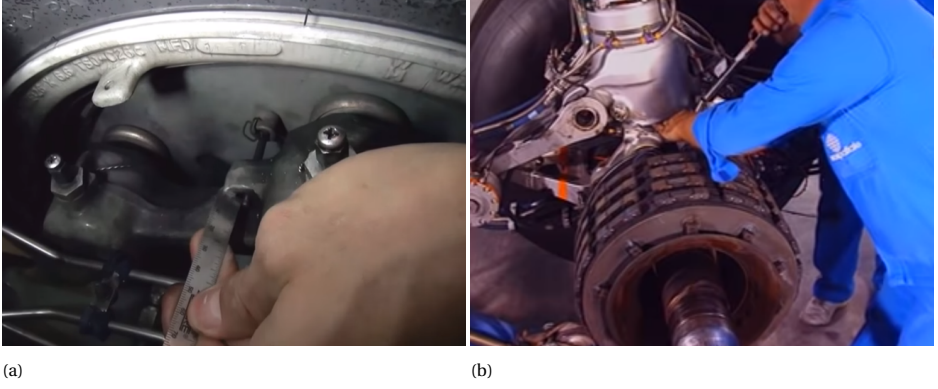


Figure 2.17: (a) Landing gear brake inspection. (b) Landing gear brake replacement.

Image source (a): <https://www.youtube.com/watch?v=ki6BJt--e3k>

Image source (b): <https://www.youtube.com/watch?v=RSrTX10i2hc>

2.4.2. DEGRADATION MODEL OF THE AIRCRAFT LANDING GEAR BRAKES

We model the continuous degradation of an aircraft brake using a Gamma process [9, 15, 37]. During a flight cycle (see Figure 2.2), the brakes are used: after take-off to stop the wheels before retraction, during landing to decelerate, and during taxi to stop or to make turns. These phases are shorter compared to the entire flight-time. Thus, we assume that the brakes are used the same amount of time in each flight cycle, i.e., $u(\tau_i^{\text{arr}} - \tau_i^{\text{dep}}) = 1$ in Equation (2.1).

Let the degradation level of a brake at the beginning and the end of the flight-time of flight cycle i be $Z(\tau_i^{\text{dep}})$ and $Z(\tau_i^{\text{arr}})$, respectively. Then, we model the brake degradation increment during flight-time i as follows (i.e., Equation (2.1) becomes):

$$Z(\tau_i^{\text{arr}}) - Z(\tau_i^{\text{dep}}) \sim \text{Gamma}(\alpha, \beta), \quad (2.10)$$

where $\alpha > 0$ is the shape parameter and $\beta > 0$ is the scale parameter of the Gamma distribution. We also assume that the degradation is negligible during ground-time.

We consider two maintenance tasks: inspection and replacement of the brakes. Following an inspection during the ground-time of flight cycle i , the degradation level remains the same, i.e.,

$$Z(\tau_i^{\text{arr}}) = Z(\tau_{i+1}^{\text{dep}}). \quad (2.11)$$

Following a replacement at the ground-time of flight cycle i ,

$$Z(\tau_{i+1}^{\text{dep}}) = 0, \quad (2.12)$$

which indicates that the brake is new and has no degradation at the beginning of flight cycle $i + 1$.

For simplicity, when no brake replacement occurs during flight cycle i , we denote the degradation level at the end of the flight-time i as:

$$Z_i = Z(\tau_i^{\text{arr}}) = Z(\tau_{i+1}^{\text{dep}}) \quad (2.13)$$

Then, using this in Equation (2.10), $(Z_{i+1} - Z_i) \sim \text{Gamma}(\alpha, \beta)$. Thus, during the time between flight cycles i_1 and i_2 , ($i_2 > i_1$), given that there is no brake replacement, the degradation Z_i follows a Gamma process with the linear shape function $\alpha(i_2 - i_1)$:

$$Z_{i_2} - Z_{i_1} \sim \text{Gamma}(\alpha(i_2 - i_1), \beta). \quad (2.14)$$

If a brake is replaced during the ground-time of flight cycle i_{rep} , then $Z_{i_{\text{rep}}+1} = 0$, and we restart the Gamma process from the flight cycle $i_{\text{rep}} + 1$.

Lastly, we consider a predefined degradation threshold η . Once $Z_i \geq \eta$, the brake is assumed to be inoperative. Without loss of generality, under a proper scaling, we consider $\eta = 1$.

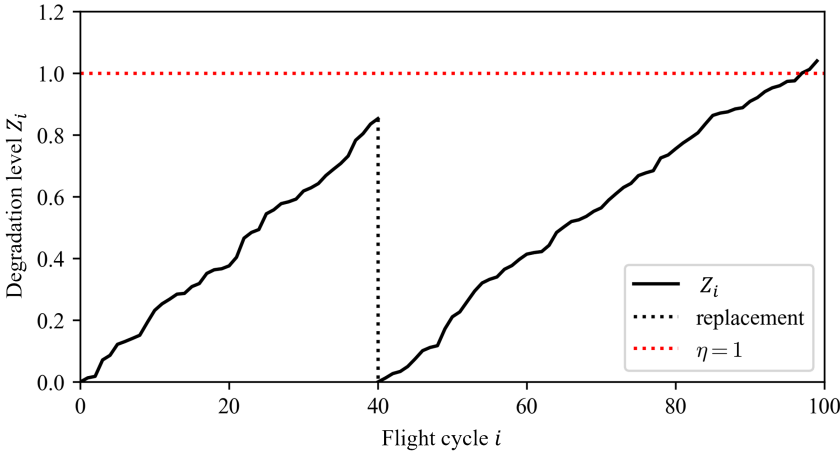


Figure 2.18: A realization of the degradation process following Equation (2.14).

Figure 2.18 shows an example of the degradation process $\{Z_i\}$ following Equation (2.14) where $\alpha = 2$, $\beta = 0.01$. Here, we consider 100 flight cycles and a degradation threshold $\eta = 1$. The degradation level increases until the brake is replaced at the flight cycle $i = 40$. A new degradation process is restarted from flight cycle $i = 41$ with $Z_{41} = 0$. As a result, Equation (2.13) does not hold for $i = 40$, i.e., $Z(\tau_{40}^{\text{arr}}) \neq Z(\tau_{41}^{\text{dep}})$. In the flight cycles $i \geq 98$, $Z_i \geq \eta$, which implies that the aircraft is released with the brake degraded more than the acceptable level.

We construct the LPNs of the 8 brakes in the agent AC as shown in Figure 2.19. Each LPN **AC Brake- ξ** shows the LPN of brake- ξ , where $\xi = \{1, 2, \dots, 8\}$ (see also Figure 2.16). Each LPN is made by taking only two necessary transitions for degradation and replacement of brakes from the LPN of a general aircraft component in Figure 2.7. Each LPN uses tokens whose color is given by the parameters of the degradation process. For example, the token in the LPN **AC Brake- ξ** has color $(\xi, Z_i^\xi, \alpha^\xi, \beta^\xi)$, where Z_i^ξ is the degradation level of the brake- ξ at the end of the flight cycle i , α^ξ and β^ξ are the shape and the scale parameters of the degradation process of brake- ξ , defined in Equation (2.14). Each transition Replacement brake- ξ has the corresponding input places Trigger replacement

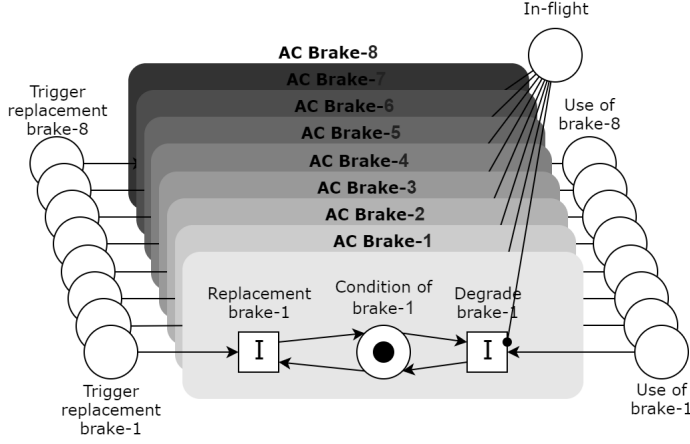


Figure 2.19: LPNs : Brakes of the agent AC.

brake- ξ . Each transition Degrade brake- ξ has its own input place Use of brake- ξ , but has a common enabling place In-flight because the operational state of the aircraft applies to all brakes.

2.4.3. MAINTENANCE STRATEGIES FOR AIRCRAFT LANDING GEAR BRAKES

We consider four aircraft landing gear brake maintenance strategies, which we refer to as TBM-CI, TBM-FI, CBM, and PdM. TBM-CI is a time-based maintenance strategy that uses fixed time intervals (flight cycles) at which visual inspections are conducted by mechanics. Such time-based maintenance strategies are often used in practice. In this chapter, we consider TBM-CI to be a baseline strategy. The TBM-FI strategy is a time-based maintenance strategy that requires more frequent inspections compared to TBM-CI, i.e., it uses shorter inspection interval. The CBM strategy uses sensor data such that inspections are triggered only after the sensor data indicates a high level of degradation. The PdM strategy uses sensor data to estimate the RUL of the brakes. In turn, using the RUL, the moment for brake replacements is decided. Unlike the other three maintenance strategies, PdM does not rely on visual inspections conducted by the mechanics, instead, utilizes the sensor data and RUL estimation. Below we specify these four maintenance strategies

TBM-CI STRATEGY

TBM-CI strategy requires periodic brake inspections at fixed intervals of flight cycles. Under TBM-CI, we assume that the brakes are inspected every 50 FCs, i.e., $d_{\text{ins}}^{\text{TBM-CI}} = 50$ FCs. Upon an inspection, if \hat{Z}_i^ξ the observed degradation level of brake- ξ exceeds a replacement threshold $\eta_{\text{rep}} = 0.97$, but is not larger than $\eta = 1$, i.e., $\eta_{\text{rep}} \leq \hat{Z}_i^\xi \leq \eta$, then a replacement of brake- ξ is scheduled within 20 FCs. We call such a replacement a *scheduled replacement*. If $\hat{Z}_i^\xi \geq \eta$, then brake- ξ is replaced immediately, before the next flight cycle. We call such a replacement an *unscheduled replacement*.

Under the TBM-CI strategy, the agents are modeled as follows. The agent TP generates a task token for scheduled inspections of the eight brakes in every $d_{\text{ins}}^{\text{TBM-CI}}$. The agent ME executes inspections and replacements of the eight brakes. The LPN **Task Execution** of the agent ME is given in Figure 2.20. For simplicity, Figure 2.20 shows two tasks: Replacing brake- ξ and Inspecting brake- ξ , which are applied in the same way for all eight brakes, $\xi \in \{1, 2, \dots, 8\}$. After an inspection, the agent ME has three possible actions, based on the inspected condition of the brake \hat{Z}^ξ (see Figure 2.20) in the form of three guard transitions connected to the place Deciding:

i) If $\hat{Z}_i^\xi < \eta_{\text{rep}}$, the transition Complete moves the token from the place Deciding to the place Completing, without generating new task tokens.

ii) If $\eta_{\text{rep}} \leq \hat{Z}_i^\xi < \eta$, the transition Request scheduled replacement fires a new task token to the place Task to plan, so that the agent TP can schedule a replacement within 20 FCs.

iii) If $\hat{Z}_i^\xi \geq \eta$, the transition Generate unscheduled replacement fires a new task token to the place Task to execute. This unscheduled replacement is executed immediately, before the next departure of aircraft.

For the agent AC, because TBM-CI does not use sensor data, we do not have tokens in LPN **AC Sensor- ξ** (Figure 2.8), and do not consider the agent DM. For the agent TG, TP and CR, the agent models in Section 2.3 are used.

TBM-FI STRATEGY

The TBM-FI strategy is similar to TBM-CI, but now we consider twice as many inspections, i.e., $d_{\text{ins}}^{\text{TBM-FI}} = 25$ FCs.

CBM STRATEGY

The CBM strategy utilizes \tilde{Z}_i^ξ , the sensor data on the condition of the brakes, to decide the moment of brake inspections. As soon as $\tilde{Z}_i^\xi \geq \eta_{\text{ins}}^{\text{CBM}}$ with $\eta_{\text{ins}}^{\text{CBM}} = 0.75$, we schedule inspections every 50 FCs, i.e., $d_{\text{ins}}^{\text{CBM}} = 50$. When considering the CBM strategy, we discard early inspections required under TBM-CI and TBM-FI strategies. In particular, we are interested in discarding early inspections, when the degradation level of the brake is low.

Under the CBM strategy, agent models of AC and DM are modified, in comparison to the TBM-CI strategy, as follows. The agent AC has additional LPNs, representing the sensors. For the 8 brakes, We have 8 sensor LPNs as shown in Figure 2.8. In each LPN of sensor- ξ , transition Monitor has its own input place Trigger sensor- ξ and enabling place Condition of brake- ξ . Also the LPN **Alert System** of the agent DM is adjusted from the general LPN **Alert System** (see Figure 2.10) for the 8 brakes as shown in Figure 2.21. The transitions Alert and Activate have 8 enabling places for 8 sensors, i.e., Sensor- ξ working for $\xi \in \{1, 2, \dots, 8\}$. The transition Alert is fired if $\exists \xi \in \{1, 2, \dots, 8\}$ such that $\tilde{Z}_i^\xi(t) \geq \eta_{\text{ins}}^{\text{CBM}}$. The transition Activate is fired if $\tilde{Z}_i^\xi(t) < \eta_{\text{ins}}^{\text{CBM}}$ for $\forall \xi \in \{1, 2, \dots, 8\}$. Then, the agent TG generates periodic inspection tasks using the token in the place Feedback from DM (see Figure 2.11).



2



2

PdM STRATEGY

Lastly, the PdM strategy schedules brake replacements based on a data-driven estimation of the RUL (prognostic) of the brakes. We define RUL^ξ as the predicted number of remaining flight cycles until the degradation level of brake- ξ becomes unacceptable, $Z_i^\xi \geq \eta$. Under PdM, the sensors monitor the condition of the brakes in every flight cycles and the data on the condition of the brakes, $\{\tilde{Z}_i^\xi\}$, is stored.

Using a linear regression to analyze the sensor data, we estimate the degradation level of the brakes in the upcoming flight cycles. Let $\tilde{Z}_{(i+j)}^\xi$ be the degradation level after j flight cycles when i is the latest completed flight cycle at the moment of RUL estimation. In other words, we have data for $\{\tilde{Z}_0^\xi, \dots, \tilde{Z}_i^\xi\}$ and we estimate the degradation at flight cycle $i + j$.

We consider the following linear regression model:

$$\tilde{Z}_{(i+j)}^\xi = \omega_0^\xi + \omega_1^\xi \cdot j, \quad (2.15)$$

where we estimate the coefficients ω_0^ξ and ω_1^ξ by the ordinary least squares method. Then we have that:

$$RUL^\xi = \min\{j \in \mathbb{Z}^+ \mid \omega_0^\xi + \omega_1^\xi \cdot j \geq \eta\}. \quad (2.16)$$

Using this approach, we estimate RUL^ξ every flight cycles for eight brakes. Now, if $RUL^\xi \leq 30$ FCs, then brake- ξ is replaced after RUL^ξ FCs.

The agent DM under PdM is specified by adding the LPN **Prognostics** in Figure 2.9 and giving a token in the place Prognostics. The agent DM receives the data \tilde{Z}_i^ξ from the token in the enabling places Sensor- ξ working. The transition Estimate RUL of component- ξ stores the data set $\{\tilde{Z}_i^\xi\}$ and estimates RUL^ξ , updating the token S_P in the place Prognostics. The guard transition Alert TG based on RUL of component- ξ fires a token to the place Feedback from DM, if $RUL^\xi \leq 30$ FCs. Then, the agent TG generates a task token to replace brake- ξ , using the LPN **Task Generation** (see Figure 2.11).

2.4.4. SAFETY AND EFFICIENCY INDICATORS OF MAINTENANCE FOR AIRCRAFT LANDING GEAR BRAKES

In this section we define indicators that show the safety and efficiency of the maintenance strategies for aircraft landing gear brakes. To assess the safety of the maintenance strategy, we define a safety incident, which is an undesirable event considering the safety of the aircraft operation. The aircraft is designed to be safe even with some inoperative brakes whose degradation level is greater than threshold, i.e., $Z(t) \geq \eta$. The master minimum equipment list (MMEL) specifies the minimum number of operable brakes to dispatch an aircraft safely. For example, in the case of Airbus A350 and Boeing B787, MMEL specifies that the aircraft can be dispatched if it has more than three operable brakes on each side [5, 6]. In line with the MMEL, if we dispatch an aircraft with more than one inoperative brakes on at least one side, we regard it as a *brake-related safety incident*. A formal definition is as follows:

Definition 1 (brake-related safety incident) We say that there is a brake-related safety incident at flight cycle i , if the incident indicator function $\mathbb{I}(i) = 1$, where $\mathbb{I}(i)$ is defined as

follows:

$$\mathbb{I}(i) = \mathbb{1} \left(\left(\sum_{\xi \in L} [\mathbb{1}(Z_i^\xi \geq \eta)] \geq 2 \right) \vee \left(\sum_{\xi \in R} [\mathbb{1}(Z_i^\xi \geq \eta)] \geq 2 \right) \right), \quad (2.17)$$

where $L = \{1, 2, 5, 6\}$ and $R = \{3, 4, 7, 8\}$ are the sets of position indices of the brakes on the left and right side of the aircraft, respectively. And, $\mathbb{1}(\cdot)$ is an indicator function which is 1 if the given logical expression is true and 0 else.

Based on the brake-related safety incident, we define two safety assessment indicators, $T_{\text{Inc}}(j)$ and $N_{\text{Inc}}(t)$ as below:

Definition 2 We say that if $\mathbb{I}(i) = 1$ for flight cycle i , the brake-related safety incident occurs at the arrival time τ_i^{arr} . $T_{\text{Inc}}(j)$ is the time when the j^{th} brake-related safety incident occurs, which is defined as follows:

$$T_{\text{Inc}}(j) = \min \{ \tau_i^{\text{arr}}, i \in I \mid (\mathbb{I}(i) = 1) \wedge (\tau_i^{\text{dep}} > T_{\text{Inc}}(j-1)) \}, \quad (2.18)$$

with $T_{\text{Inc}}(0) = 0$.

Definition 3 Let $N_{\text{Inc}}(t)$ denote the number of brake incidents that occur by time $t > 0$, which is defined as follows:

$$N_{\text{Inc}}(t) = \sum_{j=1}^{\infty} \mathbb{1}(T_{\text{Inc}}(j) < t). \quad (2.19)$$

Following Definitions 2 and 3, we denote by $T_{\text{Inc}}(1)$ the time the first brake incident occurs and by $N_{\text{Inc}}(t_H)$ the total number of brake incidents occurred by the time horizon of simulation $t_H > 0$. These two indicators are used to understand the safety of brake maintenance strategies. For instance, $P[T_{\text{Inc}}(1) \leq t]$, the probability to have an incident by time t , is used to understand how risk evolves over time under a particular maintenance strategy.

To assess the efficiency of the maintenance strategies, we consider i) the number of maintenance tasks executed in a period t_H , and ii) the degradation level of brakes at the moment of replacement.

Definition 4 We denote by $N_{\text{Task}}(t_H)$ the total number of maintenance tasks, both inspections and replacements, that occur in a period of time t_H . Similarly, $N_{\text{Rep}}(t_H)$, $N_{\text{Sch}}(t_H)$, and $N_{\text{Uns}}(t_H)$ implies the number of all replacements, scheduled replacements, and unscheduled replacements that occur in a period of time t_H .

Definition 5 We denote by $Z_{i_{\text{rep}}}$ the degradation level of brakes at the moment of replacement, given that the brake is replaced after flight cycle i_{rep}

In general, as long as safety is maintained, a low number of maintenance tasks is preferred. This is because a large number of maintenance tasks generally implies higher maintenance cost. The maintenance strategies given in Section 2.4.3 use two types of maintenance tasks, i.e., inspection, and replacement. Inspection is generally less expensive task compared to replacement. There are two types of replacement tasks, i.e.,

scheduled replacement and unscheduled replacement. Unscheduled replacements are not desired because they may cause unexpected ground-time with a high chance, especially when maintenance resources such as spare components, mechanics, or hangars are not available [3].

Also, if $Z_{i_{\text{rep}}} \geq \eta$, the replacement is performed after the brake is degraded beyond the threshold η . On the other hand, $Z_{i_{\text{rep}}} < \eta$ implies that the operable brake is replaced before the threshold η . This may be a waste of resources in the sense that we replace the brake that can be used more. Considering both safety and efficiency of the maintenance, it is desired to have $Z_{i_{\text{rep}}}$ as large as possible, but not exceeding η .

2.4.5. ESTIMATION OF THE MODEL PARAMETERS

First, we estimate the parameters of the brake degradation model in Section 2.4.2. The parameters α and β of the Gamma process in Equation (2.14) are estimated based on the sensor data recording the thickness of the brake discs. This data is collected from a fleet of wide-body aircraft, where aircraft have been in operation for a period of 6 months up to 3 years.

The disc thickness data is scaled such that it indicates the degradation level Z_i of a brake following Equation (2.14). The thickness of a brand new brake disc is scaled to be $Z_i = 0$. The thickness of a brake disc that needs to be replaced is scaled to be $Z_i = 1$, in line with our replacement threshold $\eta = 1$.

Figure 2.22 shows the degradation level data obtained from brake-1, $\{\tilde{Z}_i^1\}$. Each line indicates the recorded degradation data between two consecutive replacements. The x-axis shows the number of the flight cycles since the brake is replaced. Some data sets start from the non-zero degradation level because some degraded brakes are initially installed in practice to avoid the case when multiple brakes become inoperative at the same time.

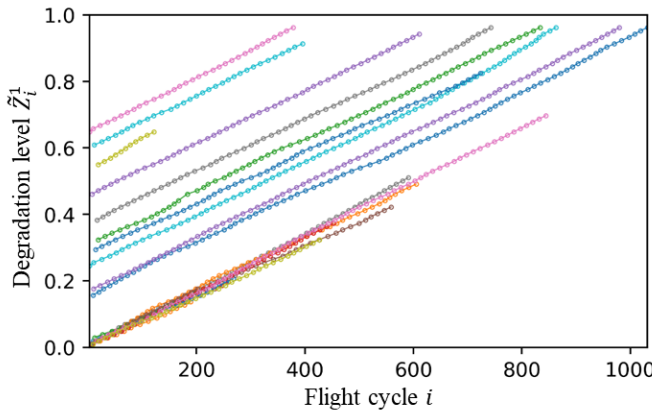


Figure 2.22: Degradation level data of the aircraft brake-1.

We first estimate the parameters α and β of the Gamma process in Equation (2.14) from the recorded degradation level data sets $\{\tilde{Z}_i\}$, using the maximum likelihood es-

timation (MLE) method as follows. Let Δi be the number of flight cycles between two successive data points \tilde{Z}_i and $\tilde{Z}_{i+\Delta i}$. Thus, the increment of the brake degradation level between flight cycle i and $i + \Delta i$ follows a Gamma distribution:

$$\tilde{Z}_{i+\Delta i} - \tilde{Z}_i \sim \text{Gamma}(\alpha \Delta i, \beta) \quad (2.20)$$

We now apply the MLE method to estimate the parameters α and β of the gamma distribution in Equation (2.20) following the method proposed in [38, 39, 40].

Table 2.2: Estimation of the parameters of the aircraft brake degradation model $\text{Gamma}(\alpha, \beta)$. In the first column, L and R indicate the brake is on the left and right side, respectively.

Brake position ξ	Parameters $\hat{\alpha}$ $\hat{\beta}$		KS test rejection rate
1 (L)	3.350	2.063e-4	0.23%
2 (L)	4.146	1.836e-4	3.28%
3 (R)	3.546	2.217e-4	0.40%
4 (R)	3.390	2.171e-4	4.82%
5 (L)	4.667	1.715e-4	1.43%
6 (L)	4.100	1.856e-4	0.11%
7 (R)	3.068	2.329e-4	0.07%
8 (R)	2.583	2.852e-4	0.45%

Table 2.2 shows the estimated parameters $\hat{\alpha}$ and $\hat{\beta}$ for each of the eight brakes. The difference in the parameters among brake positions can be explained by, for instance, the layout of the airport which requires the aircraft to perform a different number of left and right turns while taxiing at an airport.

Next, we conduct a Kolmogorov-Smirnov (KS) test to verify the following null hypothesis:

$H_0: \{Z_i\}$ follows a gamma process with shape parameter $\hat{\alpha}$ and scale parameter $\hat{\beta}$.

Since our Gamma process data points are not equally spaced, i.e., each data $\tilde{Z}_{i+\Delta i} - \tilde{Z}_i$ follows a different Gamma distribution, we cannot directly apply KS test. To address this, based on the original data $\{\tilde{Z}_i\}$, we resample an equally spaced Gamma process data $\{\tilde{Z}_{i'}\}_{i' \in I'}$ such that I' is an equally spaced flight index set [41]. The data $\tilde{Z}_{i'}$ is re-sampled using the interpolation between two consecutive available data points $\tilde{Z}_{i_l} < \tilde{Z}_{i_r}$ and by constructing a Gamma bridge as follows [41]:

$$\frac{\tilde{Z}_{i'} - \tilde{Z}_{i_l}}{\tilde{Z}_{i_r} - \tilde{Z}_{i_l}} \sim \text{Beta}(\alpha(i' - i_l), \alpha(i_r - i')), \quad (2.21)$$

where $\text{Beta}(\alpha(i' - i_l), \alpha(i_r - i'))$ is a Beta distribution with two shape parameters $\alpha(i' - i_l)$ and $\alpha(i_r - i)$.

Figure 2.23 shows a part of the recorded, original brake data $\{\tilde{Z}_i\}$ which has unequal intervals Δi , and the equally spaced data $\{\tilde{Z}_{i'}\}$ that is re-sampled from the original data $\{\tilde{Z}_i\}$, as shown in Equation (2.21).

Because this approach is based on sampling from a Beta distribution, we repeat the KS test with different realization of the resampling and determine the average rejection

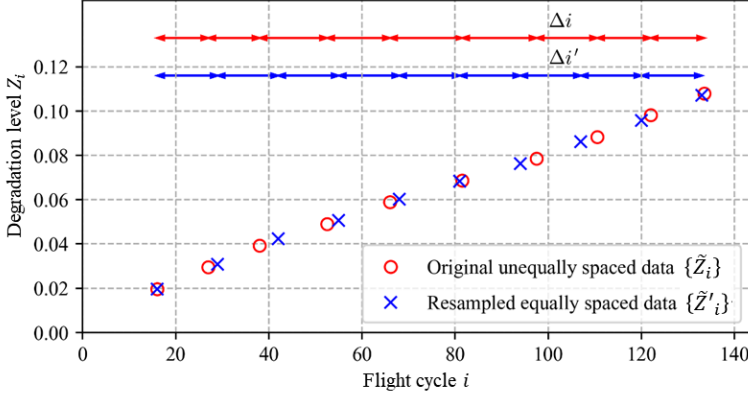


Figure 2.23: Degradation process data example – unequally spaced data points \tilde{Z}_i , and equally spaced data points Z'_i , after resampling.

rate[41]. Table 2.2 shows the rejection rate for 10^4 KS tests with a significance level of 0.05.

Apart from the brake degradation model, we also assume that the inspection error ϵ_{ins} in Equation (2.8), follows a normal distribution, i.e., $\epsilon_{\text{ins}} \sim \mathcal{N}(0, \sigma_{\text{ins}}^2)$. Here, $\sigma_{\text{ins}} = 7.53 \times 10^{-5}$ is assumed based on the minimum scale of the degradation measurement during visual inspection.

Lastly, we estimate the sensor accuracy ϵ_s in Equation (2.4) by comparing the sensor data and the detailed brake inspection reports conducted by the manufacturer of the brakes. Assuming that the detailed inspection is accurate enough, we estimate $\epsilon_s = \tilde{Z}_i - Z_i$, the error between the sensor readings and the detailed inspection results, which has a mean and standard deviation of 0.000327 and 0.0204, respectively. We assume that ϵ_s follows a non-biased Gaussian distribution $\mathcal{N}(0, 0.0204^2)$. We further test our assumption by means of a KS test with the null hypothesis:

$$H_0: \epsilon_s \sim \mathcal{N}(0, 0.0204^2).$$

The p-value of the KS test is 0.4493, and, thus, the null hypothesis is not rejected.

2.4.6. MONTE CARLO SIMULATION RESULTS

We conduct Monte Carlo simulations to evaluate the four maintenance strategies by integrating the brake degradation model (Section 2.4.2), the agent models following the maintenance strategies (Section 2.4.3), the safety and efficiency indicators (Section 2.4.4), and the estimated parameter (Section 2.4.5).

We simulate the maintenance of the landing gear brakes for a period of 10 years, i.e., $t_H = 10$ years. We generate flight cycles based on an actual flight schedule of a aircraft operated during 2015-2019 by an European airline. We initialize the degradation levels of the brakes with the observed degradation level at a random moment in the recorded data sets. This is due to the fact that, in practice, not all the eight brakes are installed as new at the same time, in order to avoid the case when all the eight brakes reach a maximum degradation level at the same time. For each maintenance strategy, 10^4 Monte Carlo

Table 2.3: Number of brake-related safety incidents in $t_H = 10$ years of operations.

	TBM-CI	TBM-FI	CBM	PdM
$\mathbb{E}[N_{\text{Inc}}(t_H)]$	0.8248	0.0470	0.8377	0.0386
95% confidence interval				
Upper bound	0.8804	0.0558	0.8936	0.0460
Lower bound	0.7692	0.0382	0.7818	0.0312

Table 2.4: Probability to have at least one brake-related safety incident by $t_H = 10$ years.

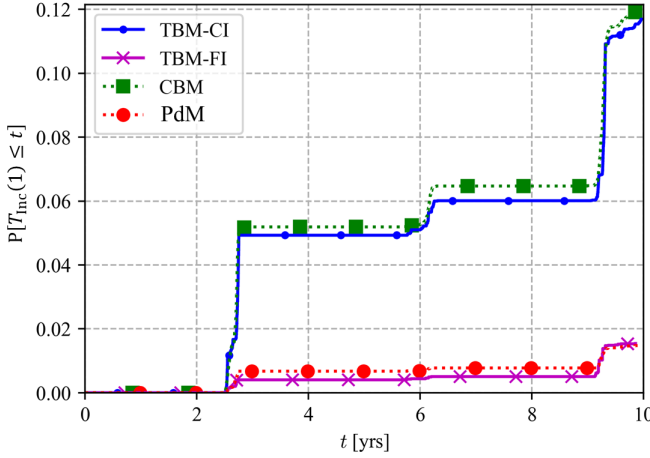
	TBM-CI	TBM-FI	CBM	PdM
$P[T_{\text{Inc}}(1) \leq t_H]$	0.1169	0.0154	0.1215	0.0148

simulation runs are conducted.

Firstly, we consider the number of brake-related safety incidents by t_H , i.e., $N_{\text{Inc}}(t_H)$ defined in Equation (2.19). Table 2.3 shows $\mathbb{E}[N_{\text{Inc}}(t_H)]$, the expected number of incidents by t_H , under the four maintenance strategies and the 95% confidence intervals of $\mathbb{E}[N_{\text{Inc}}(t_H)]$. Under the baseline maintenance strategy, TBM-CI, 0.8248 incidents are expected to occur by t_H . When we inspect the brakes twice often, under TBM-FI, the expected number of incidents decreases to 0.0470. Thus, this significantly improves the safety indicators at the cost of double the number of inspections. Under CBM, if we start the periodic inspections after the sensor indicates 75% wear of the brakes, then the expected number of incident are comparable to the case TBM-CI. Finally, under PdM, if we replace the brakes based on the RUL estimation, it is expected to have 0.0386 incidents. This indicator is significantly smaller than in the case of TBM-CI, and similar to the case of TBM-FI.

Another safety indicator is $T_{\text{Inc}}(1)$, the time when the first brake-related safety incident occurs (see Equation (2.18)). Table 2.4 shows the probability to have at least one brake-related safety incident in 10 years of aircraft operation, i.e., $P[T_{\text{Inc}}(1) \leq t_H]$. Under TBM-CI, $P[T_{\text{Inc}}(1) \leq t_H]$ is 0.1169. Under TBM-FI and PdM, $P[T_{\text{Inc}}(1) \leq t_H]$ is 0.0154 and 0.0148, respectively, which are significantly smaller than the case TBM-CI. Under CBM, $P[T_{\text{Inc}}(1) \leq t_H]$ is slightly higher than TBM-CI. Thus, compared to TBM-CI, Table 2.3 shows that the use of TBM-FI or PdM significantly improves the safety indicators of the brake maintenance in a similar degree, while CBM does not improve the safety indicators significantly.

Figure 2.24 shows the empirical cumulative distribution function (CDF) of $T_{\text{Inc}}(1)$, i.e., $P[T_{\text{Inc}}(1) \leq t]$. The CDF of $T_{\text{Inc}}(1)$ significantly increases, approximately every 3 years. This shows that the brake-related incidents are concentrated in a short interval of time. This is because the degradation of brakes reaches η after 1250-1400 FCs, which is approximately the number of flight cycles made in 3 years. By comparing the different maintenance strategies, we observe that the jumps of $P[T_{\text{Inc}}(1) \leq t]$ occurs at similar t . Thus, Figure 2.24, shows that the moment of brake-related safety incident is less affected by the maintenance strategies.

Figure 2.24: Empirical cumulative distribution function of $T_{\text{inc}}(1)$.Table 2.5: Average number of maintenance tasks executed in $t_H = 10$ years.

	TBM-CI	TBM-FI	CBM	PdM
Inspections	632.0	1272.0	402.8	-
Scheduled replacements	18.5	23.50	18.3	23.2
Unscheduled replacements	4.8	$\leq 10^{-4}$	4.9	$\leq 10^{-4}$

For the analysis of efficiency, Table 2.5 shows the expected number of tasks in $t_H = 10$ years under the four maintenance strategies. Here, we consider three types of tasks: inspections, scheduled replacements, and unscheduled replacements. Under TBM-CI, 632.0 inspections, 18.5 scheduled replacements, and 4.8 unscheduled replacements are expected to be carried out in 10 years of aircraft operation. TBM-FI uses twice as much inspections, 1272.0 but needs almost no unscheduled replacements. In this case, additional costs with inspections are expected, while the costs with the unscheduled replacements are expected to decrease. Thus, TBM-FI is expected to be more cost efficient compared to TBM-CI if 640 additional inspections are cheaper than 4.8 additional unscheduled replacements. Because CBM starts the routine inspections later, it requires only 402.8 inspections, i.e. with 33% less inspections than TBM-CI. The amount of scheduled and unscheduled replacements remain the same as in the case of TBM-CI. Thus, CBM is expected to be more cost efficient than TBM-CI since it requires less inspections. Lastly, PdM requires the least amount of tasks since this strategy does not rely on inspections and unscheduled replacements. The number of scheduled replacements under PdM is almost the same as the total amount of replacements under TBM-CI. This is because we need the same number of replacements for a given period of t_H , as we are using the same brakes for the same flight schedule.

We also analyze the degradation level of the brakes at the moment of replacement,

Table 2.6: Expected degradation level at the moment of brake replacement, $t_H = 10$ years.

	TBM-CI	TBM-FI	CBM	PdM
$\mathbb{E}[Z_{i_{\text{rep}}}^{\text{arr}}]$	1.00096	0.99487	1.00128	0.99887
95% confidence interval				
Upper bound	1.00110	0.99498	1.00144	0.99895
Lower bound	1.00080	0.99475	1.00112	0.99878

i.e., $Z_{i_{\text{rep}}}$. Table 2.6 shows the expected value of $Z_{i_{\text{rep}}}$ under each maintenance strategy. Here, $\mathbb{E}[Z_{i_{\text{rep}}}] > 1$ under TBM-CI and CBM, while $\mathbb{E}[Z_{i_{\text{rep}}}] < 1$ under TBM-FI and PdM. This is in line with the safety indicators of TBM-CI and CBM in Table 2.3. Considering the 95% confidence intervals, $\mathbb{E}[Z_{i_{\text{rep}}}]$ of PdM is higher than that of TBM-FI. This implies that the brakes are used efficiently without exceeding the threshold η under PdM.

SENSITIVITY ANALYSIS

We next analyze the sensitivity of the safety and efficiency indicators with respect to the key parameters of each maintenance strategy. Again, we consider the indicators of TBM-CI as a baseline. In the case of TBM-FI, we consider $d_{\text{ins}}^{\text{TBM-FI}}$, the interval of inspection as a key parameter. Here, TBM-FI with $d_{\text{ins}}^{\text{TBM-FI}} = 50$ FCs is identical to TBM-CI. In the case of CBM, we vary $\eta_{\text{ins}}^{\text{CBM}}$, i.e., the routine inspection is started at different degradation level. Lastly, in the case of PdM, we consider ϵ_S , the sensor error as a key parameter since the sensitivity associated with the sensor accuracy is one of the major concerns in the CBM strategies.

Figures 2.25-2.26 show the expected number of incidents $\mathbb{E}[N_{\text{inc}}(t_H)]$ and the total number of tasks under TBM-FI for $20 \leq d_{\text{ins}}^{\text{TBM-FI}} \leq 80$ FCs. Figure 2.25 shows that the expected number of incidents increases as the interval of inspection ($d_{\text{ins}}^{\text{TBM-FI}}$) increases because of the higher chance of missing a critical degradation level. At the same time, Figure 2.26 shows that the total number of tasks decreases due to the reduced number of inspections as $d_{\text{ins}}^{\text{TBM-FI}}$ increases.

In the case of CBM, Figure 2.27 shows that the expected number of incident ($\mathbb{E}[N_{\text{inc}}(t_H)]$) remains in the same level even when the routine inspections are triggered later, i.e., higher $\eta_{\text{ins}}^{\text{CBM}}$. In particular, $\mathbb{E}[N_{\text{inc}}(t_H)]$ stabilizes around 0.8, which is similar to the case of TBM-CI. Figure 2.28 shows a linear decrease in the total number of tasks. However, in Figure 2.29, the number of scheduled and unscheduled replacements does not change significantly as $\eta_{\text{ins}}^{\text{CBM}}$ increases. This implies that the decrements are mainly attributed to the reduced number of inspections. The sensitivity analysis results obtained for CBM show that we can decrease the number of tasks while keeping the same level of safety indicators if we substitute early routine inspections for the condition monitoring.

Lastly, under PdM, although the expected number of incident $\mathbb{E}[N_{\text{inc}}(t_H)]$ increases slightly as ϵ_S increases, the increase is limited in comparison to TBM-CI. Figure 2.30 shows that, even when ϵ_S is 0.08, only 0.21 safety incidents are expected, which is significantly less than the indicator under TBM-CI. Overall, Figure 2.30 shows that PdM significantly reduces the probability of having brake-related safety incidents when the

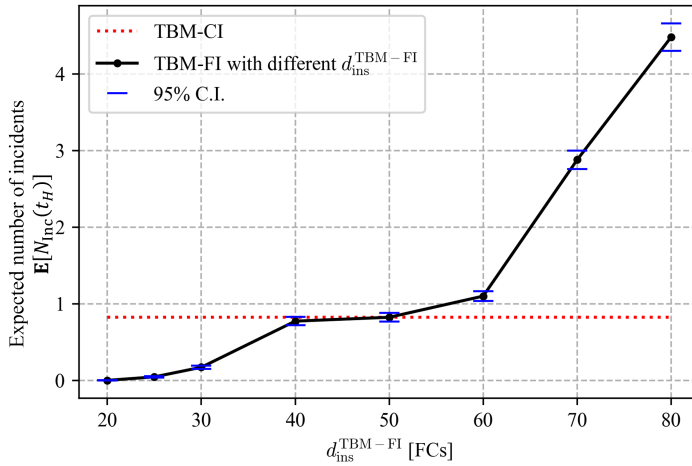


Figure 2.25: Expected number of the brake-related safety incidents under TBM-FI with different d_{ins}^{TBM-FI} .

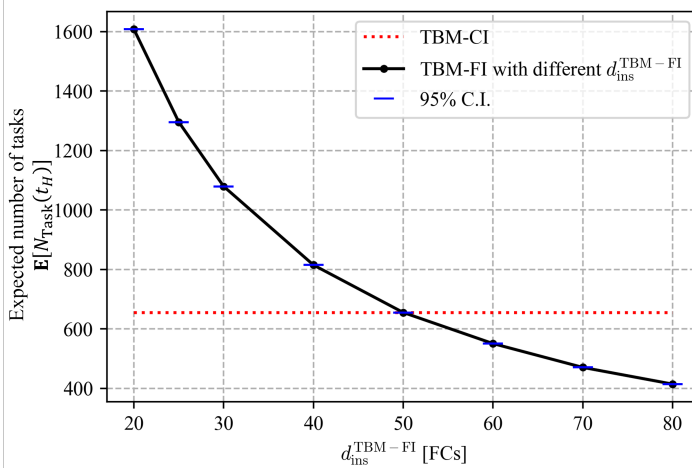


Figure 2.26: Expected total number of tasks under TBM-FI with different d_{ins}^{TBM-FI} .

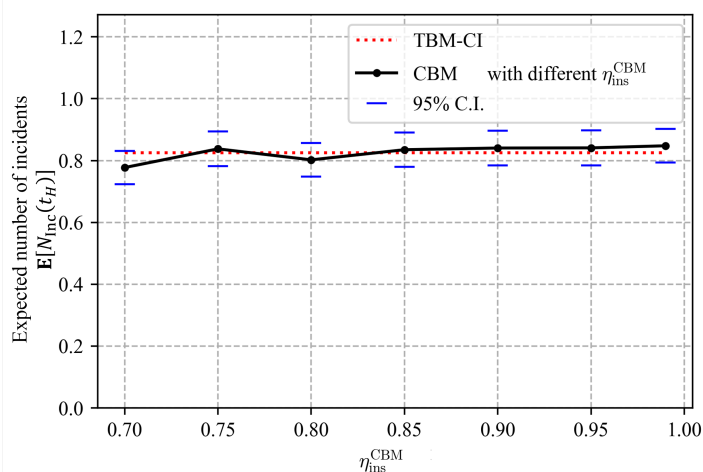


Figure 2.27: Expected number of the brake-related safety incidents under CBM with different $\eta_{\text{ins}}^{\text{CBM}}$.

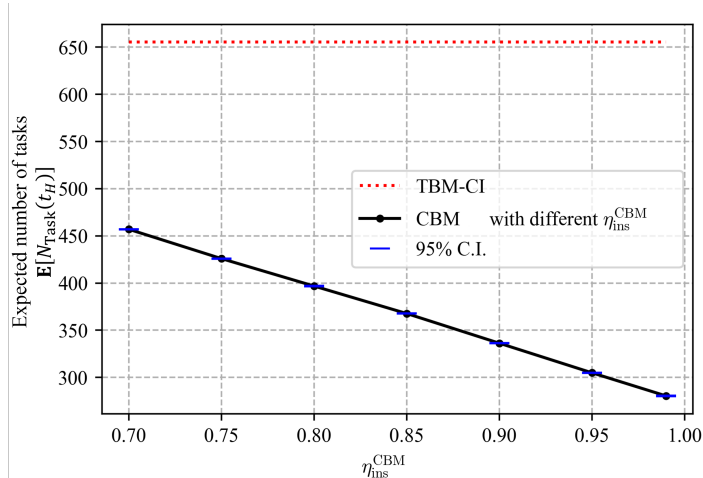


Figure 2.28: Expected total number of tasks under CBM with different $\eta_{\text{ins}}^{\text{CBM}}$.

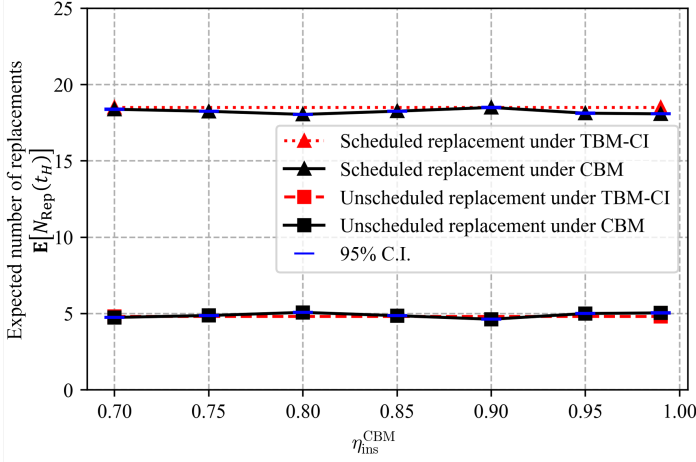


Figure 2.29: Expected number of scheduled and unscheduled replacements under CBM with different η_{ins}^{CBM}

sensor accuracy $\epsilon_s \leq 0.8$. In the case of the number of tasks, PdM relies only on scheduled replacements, and thus the number of tasks of PdM is incompatible to the other strategies that also make use of routine inspections and unscheduled replacements. For this reason, Figure 2.31 only shows the number of scheduled replacements, which is independent of the sensor error.

2.4.7. DISCUSSION

In this chapter, we propose two novel maintenance strategies, CBM and PdM, and assess them against two TBM strategies, TBM-CI and TBM-FI.

TBM-FI, which uses frequent inspections, has better safety indicators when compared with the baseline TBM-CI. However, it is not cost-efficient if the increment of the number of inspections is more expensive than the reduced number of unscheduled replacements.

For CBM, which replaces early inspections with sensor data analysis, the safety indicators are similar to those for TBM-CI, but the number of tasks, especially inspections, is reduced significantly.

The PdM strategy has similar safety indicators as TBM-FI, but significantly reduces the required number of maintenance tasks, since it uses the RUL estimation to schedule replacements. Moreover, the efficiency indicators of PdM show an improvement relative to all other strategies. In fact, PdM makes the most use of the breaks. This is shown by the fact that, under PdM, the brakes are used until their degradation level is very close to a predefined degradation threshold.

Lastly, following our sensitivity analysis, the safety indicators under CBM are not affected when the start of periodic inspections is delayed. Also, our numerical results show that the safety indicators under PdM are better than in the case of TBM-CI, even when considering larger sensor errors.

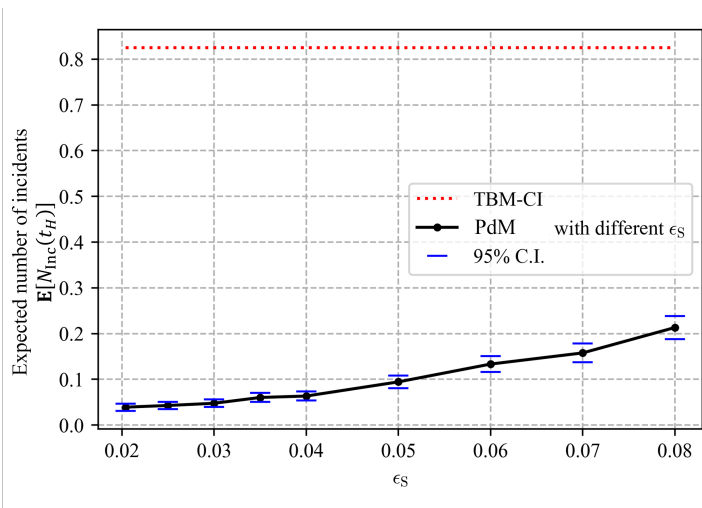


Figure 2.30: Expected number of the brake-related safety incidents under PdM with different ϵ_S .

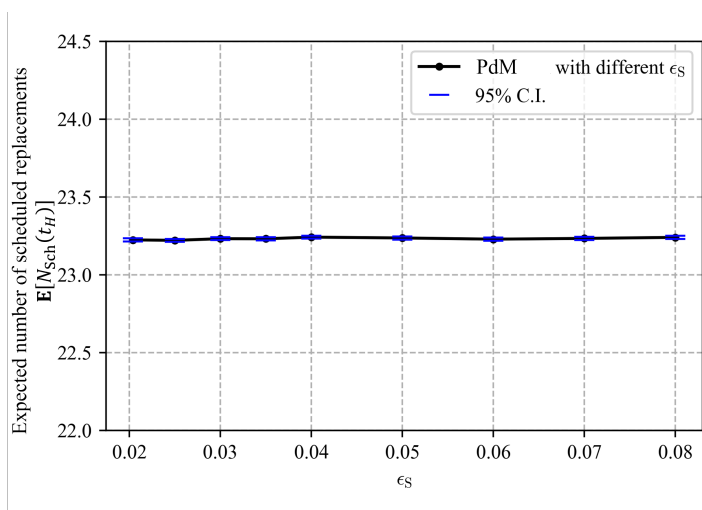


Figure 2.31: Expected total number of tasks under PdM with different ϵ_S .

From a methodology point of view, the basic model proposed in Section 2.3 can be readily adapted to other aircraft components, maintenance strategies, as well as additional agents and characteristics of the maintenance process.

2.5. CONCLUSIONS

We propose a framework to assess safety and efficiency of the aircraft maintenance process. We develop an agent-based model (ABM) of the end-to-end aircraft maintenance. The agent models are formalized by means of stochastically and dynamically colored Petri nets (SDCPNs). Next, we specify the agent models for several aircraft maintenance strategies. Using a Monte Carlo simulation of the ABM, we assess safety and efficiency indicators for the considered maintenance strategies.

We illustrate our framework for the maintenance of the aircraft landing gear brakes. We propose a condition-based maintenance (CBM) strategy and a predictive maintenance (PdM) strategy, and compare them to time-based maintenance (TBM) strategies. Then the safety and efficiency of these strategies are assessed. Our results show a trade-off between safety and efficiency. In particular, the strategy based on data-driven prognostics shows improved safety and efficiency indicators compared to TBM strategies.

In conclusion, this framework supports the assessment of novel aircraft maintenance strategies, ahead of their implementation in practice. As the ABM is designed to be generic, different aircraft components, or different strategies can be readily analyzed. Moreover, this framework is expected to be the basis of follow-up research for the design and analysis of predictive aircraft maintenance considering more realistic interactions of agents in practice.

As future work, we plan to extend our ABM for aircraft maintenance by considering heterogeneous components, limited availability of spare parts and hangars, aircraft fleet, human behaviors, and various error models.

REFERENCES

- [1] J. Hessburg. *Air Carrier MRO Handbook*. New York: McGraw-Hill Professional, 2001. ISBN: 978-0071361330.
- [2] IATA's Maintenance Cost Task Force. *Airline Maintenance Cost Executive Commentary (FY2016 data)*. Tech. rep. IATA's Maintenance Cost Task Force, 2017.
- [3] X. Zhou, L. Xi, and J. Lee. "Reliability-centered predictive maintenance scheduling for a continuously monitored system subject to degradation". In: *Reliability Engineering and System Safety* 92.4 (2007), pp. 530–534. ISSN: 09518320. DOI: [10.1016/j.ress.2006.01.006](https://doi.org/10.1016/j.ress.2006.01.006).
- [4] P. Chemweno, L. Pintelon, P. N. Muchiri, and A. Van Horenbeek. "Risk assessment methodologies in maintenance decision making: A review of dependability modelling approaches". In: *Reliability Engineering and System Safety* 173. January (2018), pp. 64–77. ISSN: 09518320. DOI: [10.1016/j.ress.2018.01.011](https://doi.org/10.1016/j.ress.2018.01.011).
- [5] FAA Flight Operations Evaluation Board (FOEB). *Master Minimum Equipment List (MMEL) Airbus A350-900 Series, All Models*. 2017.

- [6] FAA Flight Operations Evaluation Board (FOEB). *Master Minimum Equipment List BOEING 787*. 2015.
- [7] A. A. Ghobbar. “Aircraft Maintenance Engineering”. In: *Encyclopedia of Aerospace Engineering* (2010), pp. 1–14. DOI: [10.1002/9780470686652.eae552](https://doi.org/10.1002/9780470686652.eae552).
- [8] H. Wang. “A survey of maintenance policies of deteriorating systems”. In: *European Journal of Operational Research* 139.3 (2002), pp. 469–489. DOI: [10.1016/S0377-2217\(01\)00197-7](https://doi.org/10.1016/S0377-2217(01)00197-7).
- [9] M. J. Kallen and J. M. van Noortwijk. “Optimal maintenance decisions under imperfect inspection”. In: *Reliability Engineering and System Safety* 90.2-3 (2005), pp. 177–185. ISSN: 09518320. DOI: [10.1016/j.ress.2004.10.004](https://doi.org/10.1016/j.ress.2004.10.004).
- [10] S. Taghipour, D. Banjevic, and A. K. Jardine. “Periodic inspection optimization model for a complex repairable system”. In: *Reliability Engineering and System Safety* 95.9 (2010), pp. 944–952. ISSN: 09518320. DOI: [10.1016/j.ress.2010.04.003](https://doi.org/10.1016/j.ress.2010.04.003).
- [11] J. Shen, L. Cui, and Y. Ma. “Availability and optimal maintenance policy for systems degrading in dynamic environments”. In: *European Journal of Operational Research* 276.1 (2019), pp. 133–143. ISSN: 03772217. DOI: [10.1016/j.ejor.2018.12.029](https://doi.org/10.1016/j.ejor.2018.12.029).
- [12] A. K. Jardine, D. Lin, and D. Banjevic. “A review on machinery diagnostics and prognostics implementing condition-based maintenance”. In: *Mechanical Systems & Signal Processing* 20 (2006), pp. 1483–1510. ISSN: 08883270. DOI: [10.1016/j.ymssp.2005.09.012](https://doi.org/10.1016/j.ymssp.2005.09.012).
- [13] S. Alaswad and Y. Xiang. “A review on condition-based maintenance optimization models for stochastically deteriorating system”. In: *Reliability Engineering & System Safety* 157 (2017), pp. 54–63. ISSN: 09518320. DOI: [10.1016/j.ress.2016.08.009](https://doi.org/10.1016/j.ress.2016.08.009).
- [14] A. Grall, C. Bérenguer, and L. Dieulle. “A condition-based maintenance policy for stochastically deteriorating systems”. In: *Reliability Engineering and System Safety* 76.2 (2002), pp. 167–180. ISSN: 09518320. DOI: [10.1016/S0951-8320\(01\)00148-X](https://doi.org/10.1016/S0951-8320(01)00148-X).
- [15] M. Crowder and J. Lawless. “On a scheme for predictive maintenance”. In: *European Journal of Operational Research* 176.3 (2007), pp. 1713–1722. ISSN: 03772217. DOI: [10.1016/j.ejor.2005.10.051](https://doi.org/10.1016/j.ejor.2005.10.051).
- [16] H. Liao, E. A. Elsayed, and L. Y. Chan. “Maintenance of continuously monitored degrading systems”. In: *European Journal of Operational Research* 175.2 (2006), pp. 821–835. ISSN: 03772217. DOI: [10.1016/j.ejor.2005.05.017](https://doi.org/10.1016/j.ejor.2005.05.017).
- [17] Y. Gao, Y. Feng, Z. Zhang, and J. Tan. “An optimal dynamic interval preventive maintenance scheduling for series systems”. In: *Reliability Engineering and System Safety* 142 (2015), pp. 19–30. ISSN: 09518320. DOI: [10.1016/j.ress.2015.03.032](https://doi.org/10.1016/j.ress.2015.03.032).

- [18] J. Andrews, D. Prescott, and F. De Rozières. “A stochastic model for railway track asset management”. In: *Reliability Engineering and System Safety* 130 (2014), pp. 76–84. ISSN: 09518320. DOI: [10.1016/j.ress.2014.04.021](https://doi.org/10.1016/j.ress.2014.04.021).
- [19] D. Zhang, H. Hu, and C. Roberts. “Rail maintenance analysis using Petri nets”. In: *Structure and Infrastructure Engineering* 13.6 (2017), pp. 783–793. ISSN: 17448980. DOI: [10.1080/15732479.2016.1190767](https://doi.org/10.1080/15732479.2016.1190767).
- [20] B. Le, J. Andrews, and C. Fecarotti. “A Petri net model for railway bridge maintenance”. In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 231.3 (2017), pp. 306–323. ISSN: 17480078. DOI: [10.1177/1748006X17701667](https://doi.org/10.1177/1748006X17701667).
- [21] F. Santos, Â. P. Teixeira, and C. G. Soares. “Modelling and simulation of the operation and maintenance of offshore wind turbines”. In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 229.5 (2015), pp. 385–393. ISSN: 17480078. DOI: [10.1177/1748006X15589209](https://doi.org/10.1177/1748006X15589209).
- [22] J. M. Leigh and S. J. Dunnett. “Use of Petri Nets to Model the Maintenance of Wind Turbines”. In: *Quality and Reliability Engineering International* 32.1 (2016), pp. 167–180. ISSN: 10991638. DOI: [10.1002/qre.1737](https://doi.org/10.1002/qre.1737).
- [23] J. Sheng and D. Prescott. “A coloured Petri net framework for modelling aircraft fleet maintenance”. In: *Reliability Engineering and System Safety*. Vol. 189. Elsevier Ltd, 2019, pp. 67–88. DOI: [10.1016/j.ress.2019.04.004](https://doi.org/10.1016/j.ress.2019.04.004).
- [24] M. Kaegi, R. Mock, and W. Kröger. “Analyzing maintenance strategies by agent-based simulations: A feasibility study”. In: *Reliability Engineering and System Safety* 94.9 (2009), pp. 1416–1421. ISSN: 09518320. DOI: [10.1016/j.ress.2009.02.002](https://doi.org/10.1016/j.ress.2009.02.002).
- [25] M. H. Everdij, M. B. Klompstra, H. A. Blom, and B. Klein Obbink. “Compositional Specification of a Multi-agent System by Stochastically and Dynamically Coloured Petri Nets”. In: *Stochastic Hybrid Systems*. 2006, pp. 325–350. DOI: [10.1007/11587392_10](https://doi.org/10.1007/11587392_10).
- [26] M. Everdij, H. Blom, S. Stroeve, and B. Kirwan. *Agent-based Dynamic Risk Modelling for ATM*. Tech. rep. Eurocontrol, 2014.
- [27] C. M. Macal and M. J. North. “Tutorial on agent-based modelling and simulation”. In: *Proceedings of the Winter Simulation Conference*. IEEE, 2005, p. 14. DOI: [10.1057/jos.2010.3](https://doi.org/10.1057/jos.2010.3).
- [28] C. M. Macal. “Tutorial on agent-based modeling and simulation: ABM design for the zombie apocalypse”. In: *Proceedings of the 2018 Winter Simulation Conference*. 2018, pp. 207–221. ISBN: 9781538665725. DOI: [10.1109/WSC.2018.8632240](https://doi.org/10.1109/WSC.2018.8632240).
- [29] V. Panteleev, V. Kamaev, and A. Kizim. “Developing a model of equipment maintenance and repair process at service repair company using agent-based approach”. In: *Procedia Technology* 16 (2014), pp. 1072–1079. ISSN: 22120173. DOI: [10.1016/j.protcy.2014.10.121](https://doi.org/10.1016/j.protcy.2014.10.121).

- [30] H. A. Blom, J. Krystul, M. B. K. G.J. Bakker, and B. K. Obbink. “Free Flight Collision Risk Estimation by Sequential MC Simulation”. In: *Stochastic Hybrid Systems*. 2007, pp. 249–281. ISBN: 1420008544.
- [31] M. Mitici and H. A. P. Blom. “Mathematical Models for Air Traffic Conflict and Collision Probability Estimation”. In: *IEEE Transactions on Intelligent Transportation Systems* 20.3 (2019), pp. 1052–1068. ISSN: 15249050. DOI: [10.1109/TITS.2018.2839344](https://doi.org/10.1109/TITS.2018.2839344).
- [32] Air Transportation Association. *ATA MSG-3, Operator/Manufacturer Scheduled Maintenance Volume 1 - Fixed Wing Aircraft*. Vol. 1 - Fixed. Air Transport Association of America, 2013.
- [33] J. Hale. “Boeing 787 from the Ground Up”. In: *Aero* (2006), pp. 17–23.
- [34] L. Wenk and C. Bockenheimer. “Structural Health Monitoring: A real-time on-board ‘stethoscope’ for Condition-Based Maintenance”. In: *Airbus technical magazine, Flight Airworthiness Support Technology* (2014), pp. 22–28.
- [35] X. S. Si, W. Wang, C. H. Hu, and D. H. Zhou. “Remaining useful life estimation - A review on the statistical data driven approaches”. In: *European Journal of Operational Research* 213.1 (2011), pp. 1–14. ISSN: 03772217. DOI: [10.1016/j.ejor.2010.11.018](https://doi.org/10.1016/j.ejor.2010.11.018).
- [36] P. Seyte and C. P. Garcia. “Health Monitoring and Prognostics”. In: *Airbus technical magazine, Flight Airworthiness Support Technology* (2016), pp. 36–41.
- [37] J. M. van Noortwijk. “A survey of the application of gamma processes in maintenance”. In: *Reliability Engineering and System Safety* 94 (2009), pp. 2–21. ISSN: 09518320. DOI: [10.1016/j.ress.2007.03.019](https://doi.org/10.1016/j.ress.2007.03.019).
- [38] R. Edirisinghe, S. Setunge, and G. Zhang. “Application of Gamma Process for Deterioration Prediction of Buildings from Discrete Condition Data”. In: *Sri Lankan Journal of Applied Statistics* 12.1 (2012), pp. 13–25. ISSN: 1391-4987. DOI: [10.4038/sljastats.v12i0.4965](https://doi.org/10.4038/sljastats.v12i0.4965).
- [39] X. Huang and J. Chen. “Time-dependent reliability model of deteriorating structures based on stochastic processes and bayesian inference methods”. In: *Journal of Engineering Mechanics* 141.3 (2015), p. 04014123. ISSN: 07339399. DOI: [10.1061/\(ASCE\)EM.1943-7889.0000845](https://doi.org/10.1061/(ASCE)EM.1943-7889.0000845).
- [40] M. Mahmoodian and A. Alani. “Modeling Deterioration in Concrete Pipes as a Stochastic Gamma Process for Time-Dependent Reliability Analysis”. In: *Journal of Pipeline Systems Engineering and Practice* 5.1 (2014), pp. 1–10. ISSN: 1949-1190. DOI: [10.1061/\(ASCE\)PS.1949-1204.0000145](https://doi.org/10.1061/(ASCE)PS.1949-1204.0000145).
- [41] E. Grall-Maës. “Use of the Kolmogorov-Smirnov test for gamma process”. In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 226.6 (2012), pp. 624–634. ISSN: 1748006X. DOI: [10.1177/1748006X12462522](https://doi.org/10.1177/1748006X12462522).

3

MULTI-OBJECTIVE ANALYSIS OF PREDICTIVE AIRCRAFT MAINTENANCE

This chapter presents a multi-objective analysis of predictive aircraft maintenance (PdAM). We define objectives representing various key performance indicators (KPIs) of aircraft maintenance, such as aircraft reliability, and flight delay. These objectives are evaluated using Monte Carlo simulation of the model of PdAM developed in Chapter 2. We identify two groups of objectives representing the reliability and cost-efficiency of aircraft maintenance. In general, improving reliability objectives leads to trade-offs of cost-efficiency objectives. A case study shows that predictive maintenance dominates traditional maintenance strategies when considering Pareto optimality. The identified objectives are used for the optimization of PdAM in the following chapters.

Parts of this chapter have been published in the following research article:

J. Lee and M. Mitici, "Multi-objective analysis of condition-based aircraft maintenance strategies using discrete event simulation," in *2021 Annual Reliability and Maintainability Symposium (RAMS)*, pp. 1–6, Orlando, FL, USA, May 24–27, 2021.

This paper has been awarded *Thomas L. Fagan, Jr., RAMS Student Paper Award 1st Place, Reliability and Maintainability Symposium* in 2021.

3.1. INTRODUCTION

With the increasing use of condition monitoring systems, the maintenance of aircraft is undergoing a paradigm shift where data analysis is central [1, 2]. Traditionally, aircraft maintenance tasks are executed at fixed time intervals. These strategies are referred to as time-based maintenance (TBM) [3]. Nowadays, TBM is gradually replaced by condition-based maintenance (CBM), where sensor data are used to specify when and which maintenance tasks to execute. An example of CBM is the case when a maintenance task is executed as soon as sensor data indicate degradation above accepted levels [4]. Furthermore, predictive maintenance (PdM) analyzes sensor data to estimate the remaining-useful-life (RUL) of components, and schedules a maintenance task based on the estimated RUL [4, 5]. This estimated RUL is further used to schedule maintenance tasks, in anticipation of failures.

Transitioning from TBM to PdM requires the consideration of multiple objectives. One main objective of PdM for aircraft is the reduction of maintenance costs [6, 7]. Additionally, aircraft maintenance aims to comply with aircraft operational regulations [3, 8], to limit the need for unscheduled maintenance tasks [9], to reduce aircraft delays due to maintenance [10], and to utilize the aircraft as much as possible [11]. Given these multiple objectives, it is of interest to understand how they are impacted by maintenance strategies, how they are related to each other, and what are the trade-offs between them.

In this chapter, a methodology based on discrete-event simulation is proposed to analyze the relation between multiple objectives of aircraft maintenance, and to identify the trade-offs between them. Specifically, a general aircraft maintenance model is proposed for which a discrete-event simulation is conducted. The aircraft maintenance model considers the operation of the aircraft, systems of multiple, redundant aircraft components, and a stochastic degradation model for aircraft components. With this framework, multiple objectives are analyzed for a sensor-based CBM, a RUL-based PdM, and, for comparison reasons, a traditional TBM strategy. Then, conflicting objectives are identified and Pareto fronts are obtained. The resulting Pareto fronts show that the PdM strategy is located in the attractive Pareto knee region where conflicting objectives are balanced.

3.2. METHODOLOGY

Multiple objectives of the maintenance of multi-component aircraft systems are analyzed by means of a discrete event simulation. Below we introduce the aircraft maintenance model that is being simulated. This aircraft maintenance model is based on our study in [4].

3.2.1. MULTI-COMPONENT AIRCRAFT MAINTENANCE MODEL

We model the maintenance of multi-component aircraft systems considering the following events: aircraft operation (aircraft arrival and departure), degradation of components, maintenance tasks (component replacement, component inspection, and sensor monitoring), and degradation incidents when the components reach such a high level of degradation that the system becomes inoperable.

The aircraft is operated based on a sequence of flight cycles, each cycle i being de-

finied by a departure and an arrival time (see Figure 3.1). The aircraft departs from the airport at time τ_i^{dep} and arrives at the arrival airport after a flight-time $\Delta\tau_i$, where $\Delta\tau_i \sim \mathcal{N}(\overline{\Delta\tau_i}, \sigma_i^2)$. If an arrival time is $\tau_i^{\text{arr}} = \tau_i^{\text{dep}} + \Delta\tau_i$, then the time interval between this arrival and the successive departure is referred to as ground-time. Maintenance tasks are performed during ground-time. If a task is not completed until the next departure time τ_{i+1}^{dep} , the departure is delayed.

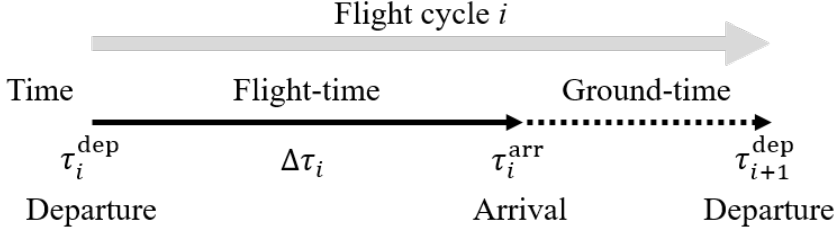


Figure 3.1: Flight cycle, where maintenance tasks can be executed during ground-time.

The aircraft consists of components that degrade during flight-time. Let the degradation level of a component at time t be $Z(t)$. A new component without degradation has $Z(t) = 0$. We say that the component is *inoperable*, considering a safety margin of degradation, if

$$Z(t) \geq 1. \quad (3.1)$$

We consider components that degrade monotonically and gradually over time, as is the case of bearings that wear out over time or brake pads that erode over time. For such components, a Gamma process is shown to model well the degradation [12]. Similarly, we assume that the degradation increment resulting from flight cycle- i follows a Gamma distribution [12]:

$$Z(\tau_i^{\text{arr}}) - Z(\tau_i^{\text{dep}}) \sim \text{Gamma}(\alpha, \beta), \quad (3.2)$$

where, α is the shape parameter, and β is the scale parameter of the Gamma process. It is assumed that the degradation is negligible during ground-time, i.e.,

$$Z(\tau_i^{\text{dep}}) - Z(\tau_{i+1}^{\text{arr}}) = 0. \quad (3.3)$$

Following Equations (3.2) and (3.3), $Z(t)$ becomes a piece-wise Gamma process.

Over time, the components undergo maintenance. As for the maintenance tasks, we consider component replacement, component inspection, and sensor monitoring.

Component replacement: When a component is replaced with a new one at time t , the degradation process is reset to be $Z(t) = 0$. The time Δt_{Rep} spent for the replacement of this component is modeled as an exponential time, i.e., $\Delta t_{\text{Rep}} \sim \text{Exp}(\bar{\delta}_{\text{Rep}})$.

Component inspection: When a component is inspected, the degradation level is known with an error. Let $\hat{Z}(t)$ be the degradation level obtained following an inspection. Then,

$$\hat{Z}(t) = Z(t) + \epsilon_{\text{Ins}}, \quad (3.4)$$

where $\epsilon_{\text{Ins}} \sim \mathcal{N}(0, \sigma_{\text{Ins}}^2)$. The inspection time Δt_{Ins} is assumed to follow an exponential distribution, i.e., $\Delta t_{\text{Ins}} \sim \text{Exp}(\bar{\delta}_{\text{Ins}})$.

Sensor monitoring: For modern aircraft equipped with condition-monitoring systems, sensors are used to automatically monitor the degradation level of the component. Let $\tilde{Z}(t)$ be the degradation level of the component obtained from sensor monitoring. Then,

$$\tilde{Z}(t) = Z(t) + \epsilon_{\text{Sen}}, \quad (3.5)$$

where $\epsilon_{\text{Sen}} \sim \mathcal{N}(0, \sigma_{\text{Sen}}^2)$. We assume that the sensor error is larger than the inspection error, i.e., $\sigma_{\text{Ins}}^2 \leq \sigma_{\text{Sen}}^2$. Compared to other tasks that require an execution time, we assume that sensor monitoring is instantaneous, i.e., $\Delta t_{\text{Sen}} = 0$.

Figure 3.2 shows an example of the degradation of a component following Equations (3.2) and (3.3). The gray regions represent flight-times, while the hatched regions represent ground-times. $Z(t)$ jumps after each flight-time following Equation (3.2). During the 5th ground time, the component is replaced, and after time $\Delta t_{\text{Rep}} = 2.5$, the degradation level of this component is reset to zero. In this example, this component is replaced before its degradation level exceeds a level of inoperability $\eta = 1$.

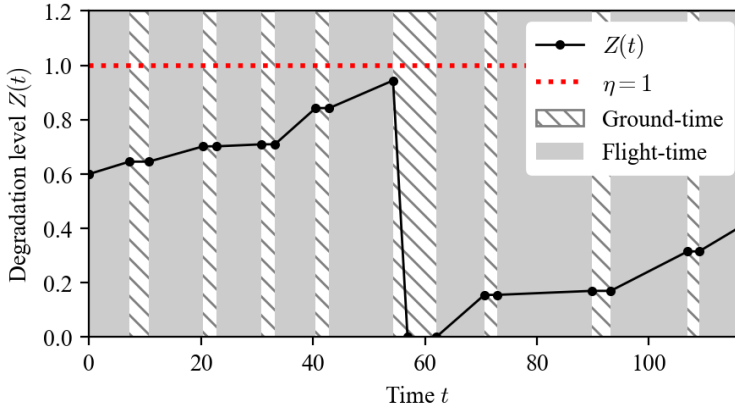


Figure 3.2: Example of component degradation over time.

For redundancy, an aircraft system often consists of multiple components. Here, we say that a multi-component system has k -out-of- n redundancy if the system consists of n components and needs to have at least k operable components, ($0 < k \leq n$). As soon as more than $(n - k)$ components become inoperable in a system with k -out-of- n redundancy, we say that a *degradation incident* occurs. The main objective of aircraft maintenance is to avoid degradation incidents, and to keep the aircraft systems operable.

We consider two aircraft systems, each of which consists of 4 components with 3-out-of-4 redundancy, i.e., a total of 8 components. We assume that the components follow the same Gamma process with parameters α and β as in Equation (3.2), and that the degradation of one component is independent of the degradation of the other components.

3.2.2. AIRCRAFT MAINTENANCE STRATEGIES AND PARAMETERS

Maintenance strategies determine the execution of maintenance tasks, i.e., which types of tasks should be executed, and when should these tasks be executed. In this study, we consider a TBM strategy named fixed-interval inspection (FII) [13], and a CBM strategy named sensor-based replacement (SBR), and a PdM strategy named RUL-based replacement (RBR). These strategies are discussed in detail in [4].

FIXED-INTERVAL INSPECTION (FII) STRATEGY

The fixed-interval inspection (FII) strategy is a TBM strategy that schedules component replacements based on periodic inspections performed by mechanics, without sensor monitoring [13]. Under the FII strategy, all components are inspected every d_{Ins} flight cycles. If upon inspection it is observed that the degradation of a component exceeds a threshold ($\hat{Z}(t) \geq \eta_{\text{Rep}}$), then the replacement of this component is scheduled within d_{Rep} flight cycles. The FII strategy has been widely implemented in traditional aircraft maintenance [4, 13].

SENSOR-BASED REPLACEMENT (SBR) STRATEGY

The Sensor-based replacement (SBR) strategy is a CBM strategy that utilizes sensor monitoring, instead of inspections performed by mechanics [4]. Under the SBR strategy, sensors measure the degradation level $\tilde{Z}(t)$ of components and report this after each flight-time. If $\tilde{Z}(t) \geq \eta_{\text{Rep}}$, where η_{Rep} is a degradation threshold, then the component is replaced within d_{Rep} flight cycles. Unlike the component inspections in the FII strategy, sensor monitoring does not cause any delays.

RUL-BASED REPLACEMENT (RBR) STRATEGY

The RUL-based replacement (RBR) strategy is a PdM strategy which uses the sensor data indicating the level of degradation to estimate the remaining-useful-life of the component, *RUL* [4]. Here, *RUL* is estimated based on the last sensor monitoring data $\{\tilde{Z}(t') | 0 < t' \leq t_o\}$, where t_o is the current time. We consider the following linear model to estimate the degradation level of a component at time $t_o + t$:

$$\tilde{Z}(t_o + t) = \omega_0 + \omega_1 t. \quad (3.6)$$

The coefficients ω_0 and ω_1 are estimated after every flight cycle based on the most recent sensor data using the ordinary least square method. Then, after each flight cycle we predict *RUL* as follows [4]:

$$RUL = \min \{t | \omega_0 + \omega_1 t \geq 1\}. \quad (3.7)$$

Finally, if *RUL* is below a threshold RUL_{min} , a component replacement is scheduled within d_{Rep} flight cycles.

Each of the three maintenance strategies has its own parameters. For instance, the FII strategy has the parameters d_{Ins} , and η_{Rep} ; the SBR strategy has the parameter η_{Rep} ; and the RBR strategy has the parameter RUL_{min} . We consider the parameter values given in Table 3.1. Each parameter has its range, and its value is selected from evenly distributed 1-levels following factorial design [14]. For example, for the values of RUL_{min} , we consider the range $20 \leq RUL_{\text{min}} \leq 60$ with steps of 1, which leads to a 41-level FD.

Table 3.1: Maintenance strategies and their parameters .

Strategy	Parameter	Range	Step	Level
FII	d_{Ins}	[20, 80]	10	7
	η_{Rep}	[0.95, 1.00]	0.002	26
SBR	η_{Rep}	[0.95, 1.00]	0.001	51
RBR	RUL_{min}	[20, 60]	1	41

3

3.2.3. MULTIPLE OBJECTIVES OF AIRCRAFT MAINTENANCE

In general, aircraft maintenance has multiple objectives, i.e., keeping the aircraft systems operational while minimizing maintenance costs and maximizing the quality of service. We introduce the following objectives [4, 6, 9, 10].

- N_{Inc} : The number of degradation incidents. This directly represents the reliability of a maintenance strategy from the perspective of keeping the aircraft systems operable [4]. A low N_{Inc} implies that it is less likely to have inoperable systems considering k -out-of- n redundancy.
- N_{Rep} : The number of component replacements. Since maintenance tasks require new components, manpower, and other resources, the number of component replacements gives a direct indication of the maintenance cost [6]. A small N_{Rep} is preferred as long as the aircraft systems are kept operational.
- N_{Uns} : The number of unscheduled component replacements. Component replacements are scheduled in advance (before d_{Rep} flight cycles) in order to have time to prepare the necessary resources. When a component replacement is necessary but there is not enough preparation time because the failure was unexpected, we call this an unscheduled component replacement. Because unscheduled replacements involve higher costs and delays [9], it is desired to minimize N_{Uns} .
- T_{D} : Aircraft delay caused by maintenance tasks. Among many causes of aircraft delay, maintenance is the second most likely cause of delays longer than one hour [10]. Thus, it is of interest to complete the maintenance tasks before a next departure time τ_{i+1}^{dep} . This is achieved by scheduling maintenance tasks only when enough ground-time is available.
- $MCTR$: The mean number of flight cycles to component replacement. This measures the exploitation time of the components. A high $MCTR$ implies that the maintenance strategy utilizes the component efficiently and does not waste the useful life of the component [4]. Thus, it is desired that $MCTR$ is maximized.

The goal is to minimize (or maximize) all these objectives by selecting a maintenance strategy with proper parameter values. However, some objectives may conflict with others. Therefore, their relation and trade-offs are analyzed next.

3.2.4. DISCRETE EVENT SIMULATION OF AIRCRAFT MAINTENANCE

Based on the aircraft maintenance model in Section 2.1, we conduct a discrete event simulation of 10 years of aircraft operations, to estimate the objective values of different maintenance strategies and parameters. A maintenance strategy and specific parameter values are referred to as a case. Specifically, a case is defined as a tuple of (*strategy, parameter*), e.g., (the RBR strategy, $RUL_{\min} = 30$). Considering the ranges and levels of the parameters in Table 3.1, we simulate 7×26 cases for the FII strategy, 51 cases for the SBR strategy, and 41 cases for the RBR strategy, which results in a total of 274 cases. For each case, we run the discrete event simulation 10^4 times and estimate the objectives using Monte Carlo methods.

3.3. SIMULATION RESULTS: MULTI-OBJECTIVE ANALYSIS

Using simulation, the objective values of the 274 cases are obtained. Again, each case corresponds to a maintenance strategy and its specific parameter values. Below we present the results obtained.

3.3.1. RELATION BETWEEN MULTIPLE OBJECTIVES

Figure 3.3 shows $\binom{5}{2}$ pairs of objective. Circle, triangle, and square markers denote the objective values of cases with the FII, SBR, and RBR strategies, respectively. Except for MCTR, all objectives are considered for minimization.

Each plot in Figure 3.3 shows the relation of a pair of objectives, where some pairs of objectives are conflicting (plots (2) – (7)), while other pairs of objectives are improved together (plot (1), plots (8)–(10)). Some relations (whether conflicting or not) can be expected before simulations. For example, it is expected that as $MCTR$ increases, N_{Rep} decreases (see plot (1)). However, since their trend and trade-off are not trivial, the simulation results can be analyzed further to obtain an in-depth understanding of the characteristics of these objectives and the maintenance strategies considered. For example, in plot (1), the relation between $MCTR$ and N_{Rep} is neither linear nor inversely proportional. Rather, when $N_{\text{Rep}} = 2.4$, $MCTR$ suddenly drops from 1,200 to 1,250. This is because $MCTR$ can be significantly different depending on the moment when components are replaced, even if the same number of component replacements are performed.

More interesting relations between conflicting objectives are shown in Figure 3.3, plots (2) – (7). For example, in plot (6) it is shown that fewer degradation incidents occur (low N_{Inc}) when components are replaced often (high N_{Rep}), i.e., there is a trade-off between N_{Inc} and N_{Rep} . However, the trade-off is unclear under the SBR strategy where there are nearly zero degradation incidents but different numbers of replacements. This shows that in some cases of the SBR strategy, the components are replaced unnecessarily often. A similar trade-off is shown between N_{Uns} and N_{Rep} in plot (5). The similarity between plots (5) and (6) is because N_{Inc} and N_{Uns} are positively correlated, as shown in plot (8).

The analysis based on Figure 3.3 is reinforced by the analysis of the correlation between the considered objectives using the Pearson correlation coefficient (see Table 3.2). The Pearson correlation coefficient quantifies the linear correlation between two objectives. A positive coefficient between two objectives implies that they are likely to be im-

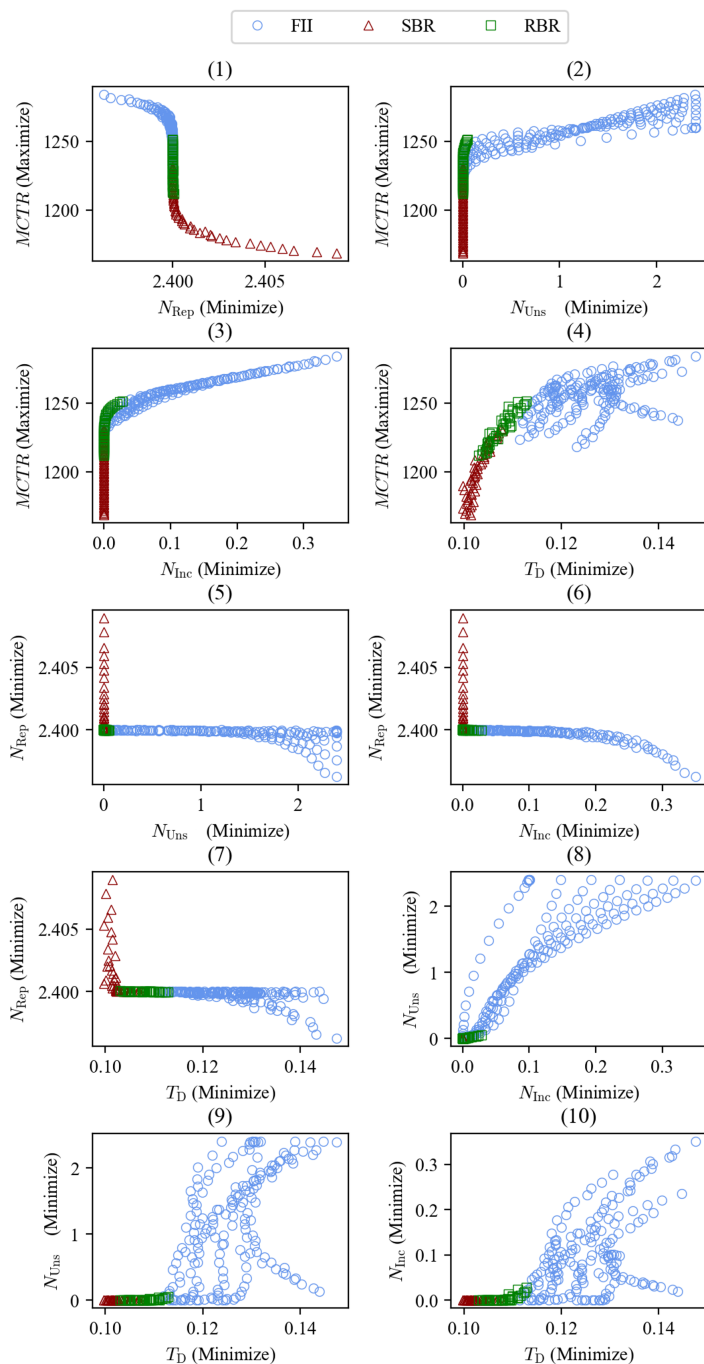


Figure 3.3: Pairwise objectives of the aircraft maintenance for the FII, SBR and RBR strategies.

Table 3.2: The Pearson correlation coefficient of objectives for 274 cases of maintenance strategies and parameters.

		Group 1		Group 2		
		$MCTR$	N_{Rep}	N_{Uns}	N_{Inc}	T_D
Group 1	$MCTR$	-	0.69	-0.77	-0.76	-0.80
	N_{Rep}	0.69	-	-0.40	-0.47	-0.50
Group 2	N_{Uns}	-0.77	-0.40	-	0.90	0.73
	N_{Inc}	-0.76	-0.47	0.90	-	0.70
	T_D	-0.80	-0.50	0.73	0.70	-

proved together, while a negative coefficient implies that they conflict with each other. For example, the Pearson coefficient between T_D and $MCTR$ is -0.80 , which represents the trade-off shown in plot (4) of Figure 3.3.

Based on the Pearson correlation coefficient values, we categorize the objectives into two groups such that the objective pairs within the same group have positive coefficients (see Table 3.2). Group 1 is $\{MCTR, N_{Rep}\}$, and Group 2 is $\{N_{Uns}, N_{Inc}, T_D\}$.

$MCTR$ and N_{Rep} in Group 1 are positively correlated as both of them measure the exploitation time of a component. Since the exploitation of components is directly related to the cost of maintenance, these objectives imply an economic benefit of the maintenance. On the other hand, N_{Uns} , N_{Inc} , and T_D in Group 2 measure the number of undesired events, i.e., unscheduled maintenance, degradation incidents, and aircraft delay due to maintenance. In other words, these objectives represent the reliability of the maintenance. The conflict between Group 1 (Cost) and Group 2 (Reliability) shows the general trade-off between the reliability and the cost of aircraft maintenance.

Although aircraft maintenance has various objectives, it is useful to analyze the maintenance strategies based on a small number of representative objectives [15]. To consider both reliability and economic aspects, we analyze the aircraft maintenance based on two objectives, one chosen from Group-1 and one from Group 2. In particular, we choose $MCTR$ from Group 1 since it better represents the economic value because the variance of N_{Rep} is very small compared to that of $MCTR$ (see the scales of N_{Rep} and $MCTR$ in plot (1) of Figure 3.3). For the objective representing reliability (Group 2), N_{Inc} or T_D are chosen. N_{Uns} is not chosen because it is strongly correlated with $MCTR$ (coefficient 0.9 in Table 3.2), and therefore N_{Uns} is improved together with $MCTR$.

3.3.2. TRADE-OFF BETWEEN AIRCRAFT MAINTENANCE OBJECTIVES

Following the analysis in Section 3.1, in this subsection we analyze the trade-offs between Group 1 objective $MCTR$ and Group 2 objectives N_{Inc} and T_D . Pareto fronts are generated for $\{MCTR, N_{Inc}\}$ and $\{MCTR, T_D\}$ by collecting non-dominated cases from the total 274 cases (the FII, SBR, or RBR strategy with their parameter values, see Table 3.1).

Figure 3.4 shows the Pareto front for the objectives $MCTR$ and N_{Inc} . Since these two objectives are conflicting, no single solution achieves a maximum $MCTR$ and a minimum N_{Inc} simultaneously. Rather, we should trade-off $MCTR$ for N_{Inc} . For instance, the number of degradation incidents can be minimized ($N_{Inc} \leq 10^{-4}$) if we accept a small

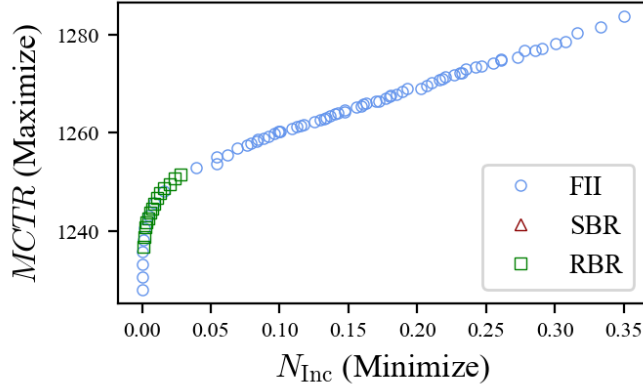


Figure 3.4: Pareto front considering $MCTR$ and N_{Inc} .

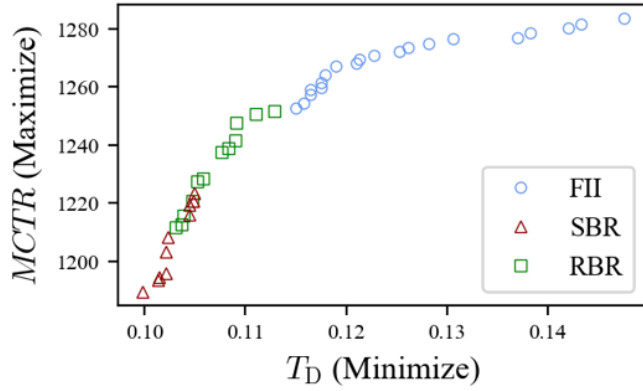


Figure 3.5: Pareto front considering $MCTR$ and T_D .

$MCTR \leq 1,235$. Or, if we want to extend $MCTR \geq 1,250$, then N_{Inc} is increased to 0.02.

The Pareto front in Figure 3.4 also provides insight into the maintenance strategies considered. In all cases, the SBR strategy is dominated by the FII or RBR strategies, thus not shown in the Pareto front in Figure 3.4. This means that the RBR or FII strategies are preferred when considering $MCTR$ and N_{Inc} . More interestingly, the cases considering the RBR strategy are located in the middle, or in the extruded region of the Pareto front, which is called the *knee region* [16]. The non-dominated solutions in the knee region are generally preferred because they provide a balanced solution, i.e., both objectives are moderately optimized. Outside of the knee region, an objective is significantly deteriorated to achieve a slight improvement in the other objective, which is less preferred for aircraft maintenance [16, 17]. By comparing plot (3) of Figure 3.3 and Figure 3.4, it can be seen that the FII strategy cases in this knee region are dominated by the RBR strategy cases. In Figure 3.4, the non-dominated FII strategy cases cause either a large number of

degradation incidents $N_{\text{Inc}} \geq 0.04$ or a low $MCTR \leq 1,240$, but the non-dominated RBR strategy cases achieve a small $N_{\text{Inc}} \leq 0.04$ and a moderate $MCTR \geq 1,240$. This indicates that PdM using RUL prognostics (the RBR strategy) is beneficial when we aim to improve both $MCTR$ and N_{Inc} .

Figure 3.5 shows the Pareto front between $MCTR$ and the delay T_D . Unlike the Pareto front in Figure 3.4, the SBR strategy is visible in the lower-left corner of the Pareto front in Figure 3.5. These non-dominated SBR strategy cases have a low delay ($T_D \leq 0.1$) but they are not cost-effective ($MCTR \leq 1,230$). The RBR strategy, on the other hand, is located in the middle of the Pareto front, where $0.105 \leq T_D \leq 0.115$ and $1,230 \leq MCTR \leq 1,250$. In this region, many FII strategy cases are dominated by the RBR strategy cases (compare plot (4) of Figure 3.3 and Figure 6). Thus, when both objectives are considered with similar importance (knee region), the introduction of the RBR strategy improves both objectives.

Overall, these results show that PdM using RUL prognostics (the RBR strategy) has a benefit in improving both the reliability (N_{Inc} , T_D) and cost ($MCTR$) of aircraft maintenance.

3.4. CONCLUSION

We have conducted a multi-objective analysis of aircraft condition-based maintenance strategies, using discrete event simulation. Our aircraft maintenance model covers the general features of the maintenance of multi-component aircraft systems, such as aircraft operations, stochastic degradation of aircraft components, redundancy of aircraft systems, and maintenance strategies.

We have considered as objectives the minimization of the mean number of flight cycles to component replacement ($MCTR$), the number of replacements (N_{Rep}), the number of degradation incidents (N_{Inc}), the number of unscheduled replacements (N_{Uns}), and the delay due to maintenance (T_D). Based on their correlation and trade-off, we chose two pairs of conflicting objectives to represent the reliability and cost of aircraft maintenance. We constructed Pareto fronts between these conflicting objectives under condition-based maintenance strategies (the SBR and RBR strategies), and a traditional time-based maintenance strategy (the FII strategy). The results show that the advanced PdM strategy (the RBR strategy) dominates the other strategies in the knee region of the Pareto fronts. This suggests that the introduction of PdM in aircraft maintenance achieves a balance between the reliability and the cost of maintenance.

REFERENCES

- [1] J. Hale. "Boeing 787 from the Ground Up". In: *Aero* (2006), pp. 17–23.
- [2] L. Wenk and C. Bockenhimer. "Structural Health Monitoring: A real-time on-board 'stethoscope' for Condition-Based Maintenance". In: *Airbus technical magazine, Flight Airworthiness Support Technology* (2014), pp. 22–28.
- [3] H. Wang. "A survey of maintenance policies of deteriorating systems". In: *European Journal of Operational Research* 139.3 (2002), pp. 469–489. DOI: [10.1016/S0377-2217\(01\)00197-7](https://doi.org/10.1016/S0377-2217(01)00197-7).

- [4] J. Lee and M. Mitici. “An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets”. In: *Reliability Engineering and System Safety* 202 (2020), p. 107052. ISSN: 0951-8320. DOI: [10.1016/j.ress.2020.107052](https://doi.org/10.1016/j.ress.2020.107052).
- [5] I. de Pater and M. Mitici. “Predictive maintenance for multi-component systems of repairables with Remaining-Useful-Life prognostics and a limited stock of spare components”. In: *Reliability Engineering and System Safety* 214 (2021), p. 107761. ISSN: 0951-8320. DOI: [10.1016/j.ress.2021.107761](https://doi.org/10.1016/j.ress.2021.107761).
- [6] M. J. Kallen and J. M. van Noortwijk. “Optimal maintenance decisions under imperfect inspection”. In: *Reliability Engineering and System Safety* 90.2-3 (2005), pp. 177–185. ISSN: 09518320. DOI: [10.1016/j.ress.2004.10.004](https://doi.org/10.1016/j.ress.2004.10.004).
- [7] A. Grall, L. Dieulle, C. Bérenguer, and M. Roussignol. “Continuous-time predictive-maintenance scheduling for a deteriorating system”. In: *IEEE Transactions on Reliability* 51.2 (2002), pp. 141–150. ISSN: 00189529. DOI: [10.1109/TR.2002.1011518](https://doi.org/10.1109/TR.2002.1011518).
- [8] M. Mitici and H. A. P. Blom. “Mathematical Models for Air Traffic Conflict and Collision Probability Estimation”. In: *IEEE Transactions on Intelligent Transportation Systems* 20.3 (2019), pp. 1052–1068. ISSN: 15249050. DOI: [10.1109/TITS.2018.2839344](https://doi.org/10.1109/TITS.2018.2839344).
- [9] J. Sheng and D. Prescott. “A coloured Petri net framework for modelling aircraft fleet maintenance”. In: *Reliability Engineering and System Safety*. Vol. 189. Elsevier Ltd, 2019, pp. 67–88. DOI: [10.1016/j.ress.2019.04.004](https://doi.org/10.1016/j.ress.2019.04.004).
- [10] M. Zámková, M. Prokop, and R. Stolín. “Factors influencing flight delays of a European airline”. In: *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis* 65.5 (2017), pp. 1799–1807. ISSN: 12118516. DOI: [10.11118/actaun201765051799](https://doi.org/10.11118/actaun201765051799).
- [11] R. J. Ferreira, A. T. de Almeida, and C. A. Cavalcante. “A multi-criteria decision model to determine inspection intervals of condition monitoring based on delay time analysis”. In: *Reliability Engineering and System Safety* 94.5 (2009), pp. 905–912. ISSN: 09518320. DOI: [10.1016/j.ress.2008.10.001](https://doi.org/10.1016/j.ress.2008.10.001).
- [12] J. M. van Noortwijk. “A survey of the application of gamma processes in maintenance”. In: *Reliability Engineering and System Safety* 94 (2009), pp. 2–21. ISSN: 09518320. DOI: [10.1016/j.ress.2007.03.019](https://doi.org/10.1016/j.ress.2007.03.019).
- [13] K. T. Huynh, A. Barros, C. Bérenguer, and I. T. Castro. “A periodic inspection and replacement policy for systems subject to competing failure modes due to degradation and traumatic events”. In: *Reliability Engineering and System Safety* 96.4 (2011), pp. 497–508. ISSN: 09518320. DOI: [10.1016/j.ress.2010.12.018](https://doi.org/10.1016/j.ress.2010.12.018).
- [14] J. P. Kleijnen. “Design Of Experiments: Overview”. In: *Proceedings of the 2008 Winter Simulation Conference*. IEEE, 2008, pp. 479–488. ISBN: 9781424427086. DOI: [10.1109/WSC.2008.4736103](https://doi.org/10.1109/WSC.2008.4736103).
- [15] S. Greco, M. Ehrgott, and J. R. Figueira. *Multiple Criteria Decision Analysis*. Springer New York, 2016. ISBN: 978-1-4939-3093-7. DOI: [10.1007/978-1-4939-3094-4](https://doi.org/10.1007/978-1-4939-3094-4).

- [16] I. Das. “On characterizing the “knee” of the Pareto curve based on Normal-Boundary Intersection”. In: *Structural Optimization* 18.2 (1999), pp. 107–115.
DOI: [10.1007/BF01195985](https://doi.org/10.1007/BF01195985).
- [17] J. Branke. “MCDA and multiobjective evolutionary algorithms.pdf”. In: *Multiple Criteria Decision Analysis: State of the Art Surveys*. 2016, pp. 977–1008.
DOI: [10.1007/978-1-4939-3094-4_23](https://doi.org/10.1007/978-1-4939-3094-4_23).

4

PREDICTIVE AIRCRAFT MAINTENANCE AT COMPONENT LEVEL USING RUL PROGNOSTICS AND DEEP REINFORCEMENT LEARNING

In this chapter, we propose a framework to optimize predictive aircraft maintenance (PdAM) at the component level. The framework consists of two parts: the development of probabilistic Remaining-Useful-Life (RUL) prognostics, and predictive maintenance planning. The probabilistic RUL prognostics estimate the probability distribution of RUL using a convolutional neural network with Monte Carlo dropout. The estimated RUL distribution is used to plan component replacements using a deep reinforcement learning (DRL) approach. DRL approach minimizes the long-term cost with predictive maintenance planning, balancing the risk of component failure and the wasted life of the component. This framework is illustrated for the maintenance of a turbofan engine.

Parts of this chapter have been published in the following research article:

J. Lee and M. Mitici, "Deep reinforcement learning for predictive aircraft maintenance using probabilistic Remaining-Useful-Life prognostics," *Reliability Engineering and System Safety*, 2022.

4.1. INTRODUCTION

Modern aircraft are equipped with multiple sensors that generate large volumes of health monitoring measurements for aircraft systems and components. For example, for a Boeing 787, approximately 1,000 parameters are continuously monitored for the engine, amounting to a total of 20 terabytes of data per flight hour [1]. Such data are the basis for Remaining-Useful-Life (RUL) estimations [2] and predictive aircraft maintenance planning [3].

Many studies have focused in the last years on developing RUL prognostics for aircraft components and systems [4]. For example, RUL prognostics for aircraft landing gear brakes are developed using stochastic regression models [5]. The RUL of aircraft cooling units are estimated using particle filtering [6]. RUL prognostics for electro - mechanical actuators are obtained using a Gaussian process regression [7]. Several RUL prognostics for turbofan engines have been developed using convolutional neural networks (CNNs) [8], deep convolutional neural network (DCNN) [9], multi-scale DCNN [10], and CNN with pooling [11, 12]. All these studies, however, predict RUL as a point estimate, i.e., a single value of RUL. Yet, quantifying the uncertainty associated with the estimated RUL is seen as a pre-requisite for predictive maintenance [13]. In this line, we develop probabilistic RUL prognostics, where the distribution of the RUL is estimated.

Most existing studies develop either RUL prognostics only, or propose advanced maintenance planning models but make simple assumptions about the degradation of systems/components. For instance, the degradation of systems are often assumed to follow stochastic processes such as Gamma processes [5, 14], Wiener processes [15], non-homogeneous Poisson processes [16], or Markov process [17, 18]. Very few studies integrate data-driven RUL prognostics into maintenance planning. In [19], for example, the replacement of aircraft brakes is scheduled taking into account data-driven RUL prognostics. In [8], RUL prognostics for aircraft engines are obtained. Based on these prognostics, alarms are triggered and maintenance actions are specified. In [20], component inspections are scheduled based on the epistemic uncertainty of the estimated RUL. In [21], the crack size of airframe panels is estimated using extended Kalman filter. These panels are further replaced based on these estimates. Similarly, we propose an integrated framework for maintenance planning where data-driven RUL prognostics for turbofan engines are used to specify the moment of engine replacement. However, these studies above integrate RUL prognostics into maintenance planning using fixed thresholds, i.e., one fixed threshold is used to trigger maintenance of all same-type components, irrespective of the estimated degradation of each individual component. For instance, in [8] all engines are replaced as soon as their RUL is estimated to be 44 days or less. In [21], the replacement of airframe panels is triggered by a fixed threshold of 47.4 mm crack size. In contrast, we propose an adaptive approach that schedules maintenance taking into account the trends of the RUL prognostics, without using any fixed threshold.

In this chapter, we propose a deep reinforcement learning (DRL) approach for predictive maintenance that adaptively schedules maintenance considering probabilistic RUL prognostics. The overview of the proposed framework is shown in Figure 4.1. Based on sensor measurements, the probability distribution of RUL is estimated using CNNs with Monte Carlo dropout. These RUL estimates are updated periodically over time, as more sensor measurements become available. We further develop a DRL approach that

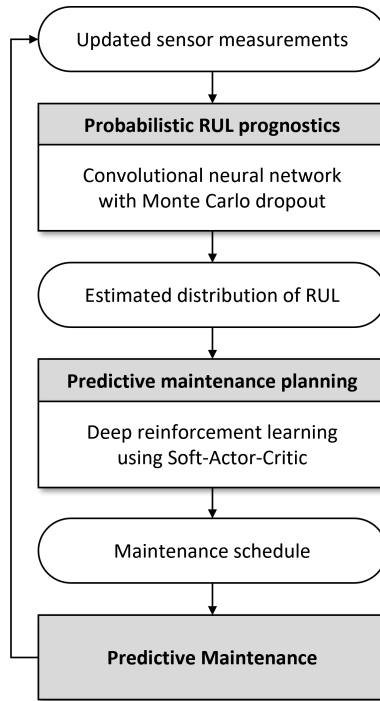


Figure 4.1: Overview of proposed predictive maintenance framework using probabilistic RUL prognostics and DRL.

uses these probabilistic RUL prognostics to plan maintenance. The probabilistic RUL prognostics directly specify the states of the DRL. Using DRL, maintenance actions are triggered adaptively, without relying on fixed thresholds. We illustrate our approach for maintenance planning of aircraft turbofan engines.

The main contributions of this chapter are as follows:

- An integrated framework for predictive maintenance is proposed, where probabilistic Remaining-Useful-Life (RUL) prognostics and deep reinforcement learning (DRL) are used to plan the maintenance of aircraft engines. Here, probabilistic RUL prognostics (the estimated RUL distribution) are directly used to construct the states of the DRL.
- Probabilistic RUL prognostics are obtained using Convolutional Neural Networks (CNNs) and Monte Carlo dropout. Using probabilistic RUL prognostics, we show that the number of unscheduled maintenance is lower than when using point-RUL estimates. This shows the benefit of quantifying the uncertainty of RUL estimates for maintenance planning.
- We pose the problem of predictive maintenance planning as a DRL problem. This approach adaptively proposes maintenance actions based on the trends of the estimated RUL prognostics.

The remainder of this chapter is organized as follows. In Section 4.2, we propose probabilistic RUL prognostics using CNN with Monte Carlo dropout. In Section 4.3, we formulate a DRL problem for predictive maintenance planning taking into account the probabilistic RUL prognostics. In Section 4.4, we illustrate our DRL approach for the maintenance of turbofan engines. In Section 4.5, we compare our DRL approach against other maintenance strategies. Finally, we provide conclusions in Section 4.6.

4.2. ESTIMATING THE DISTRIBUTION OF RUL USING CNN WITH MONTE CARLO DROPOUT

In this section, we obtain probabilistic prognostics of RUL of aircraft engines using multi-channel convolutional neural networks (CNNs) and Monte Carlo dropout. These RUL prognostics are updated after every flight cycle as new degradation data become available.

4.2.1. DATA DESCRIPTION AND PRE-PROCESSING

We consider the degradation data of aircraft turbofan engines obtained by NASA using the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) [22]. Figure 4.2 shows the simplified diagram of the turbofan engine simulated in C-MAPSS [23]. This data set consists of data subsets FD001, FD002, FD003, and FD004, each considering a specific number of fault modes and operating conditions (see Table 4.1) [24]. The training instances of the subsets have run-to-failure data of sensor measurements, while the testing instances have sensor measurements up to some moment prior to failure. Each instance consists of time-series data of 21 sensor measurements per flight cycle. Following [8, 9], we select for our analysis 14 non-constant sensor measurements. We discard the remaining 7 sensor measurements since these are constant across all flight cycles.

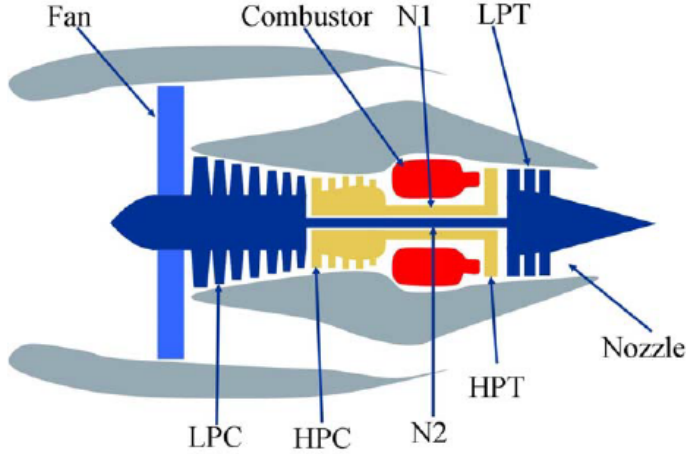


Figure 4.2: Simplified diagram of the engine simulated in C-MAPSS [23].

Table 4.1: C-MAPSS data sets for turbofan engines [24].

	FD001	FD002	FD003	FD004
Training instances	100	260	100	249
Testing instances	100	259	100	248
Operating conditions	1	6	1	6
Fault mode	1	1	2	2

We pre-process the raw data as follows. First, using the clustering of operational settings proposed by [25], 6 operating conditions are identified. Let o_k denote the operating condition of an engine during k^{th} flight cycle, $o_k \in \{1, \dots, 6\}$.

We also consider the history of operating conditions. Let $h_{o,k}$ denote the number of cycles that an engine has been operated under operating condition o , up to k^{th} flight cycle.

Next, the measurements of sensor $s \in \{1, \dots, 14\}$ are normalized with respect to operating condition o as follows [8, 11]:

$$m_{s,k} = \frac{2(m_{s,k}^o - m_{s,\min}^o)}{m_{s,\max}^o - m_{s,\min}^o} - 1, \quad (4.1)$$

where $m_{s,k}$ is the normalized measurement of sensor s at k^{th} flight cycle; $m_{s,k}^o$ is the raw measurement of sensor s at k^{th} flight cycle that is performed under operating condition o ; and $m_{s,\min}^o$ and $m_{s,\max}^o$ are the minimum and maximum measurement of sensor s under operating condition o , respectively. In total, $n_F = 21$ features are considered (the current operating conditions o_k , the history of the 6 operation conditions $h_{o,k}$, and 14 types of sensor measurements $m_{s,k}$).

Finally, n_F features for a time window of n_W flight cycles are considered as the input x of the CNN, i.e.,

$$x = \begin{bmatrix} o_1 & h_{1,1} & \dots & h_{6,1} & m_{1,1} & \dots & m_{14,1} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ o_{n_W} & h_{1,n_W} & \dots & h_{6,n_W} & m_{1,n_W} & \dots & m_{14,n_W} \end{bmatrix}. \quad (4.2)$$

Here, n_W is selected based on the number of cycles available for the shortest test instance in each data subset [8]. We use $n_W = 30$ cycles for FD001 and FD003, $n_W = 21$ for FD002, and $n_W = 19$ for FD004.

4.2.2. ARCHITECTURE OF THE MULTI-CHANNEL CNN WITH MONTE CARLO DROPOUT

To obtain probabilistic RUL prognostics, we propose a neural network architecture combining multi-channel convolutional layers, linear layers, and Monte Carlo dropout (see Figure 4.3 and Table 4.2).

In contrast to [8], where a common 1D kernel is applied for all features, we apply one 1D kernel per each time-series of a feature, i.e., each column of the input x is

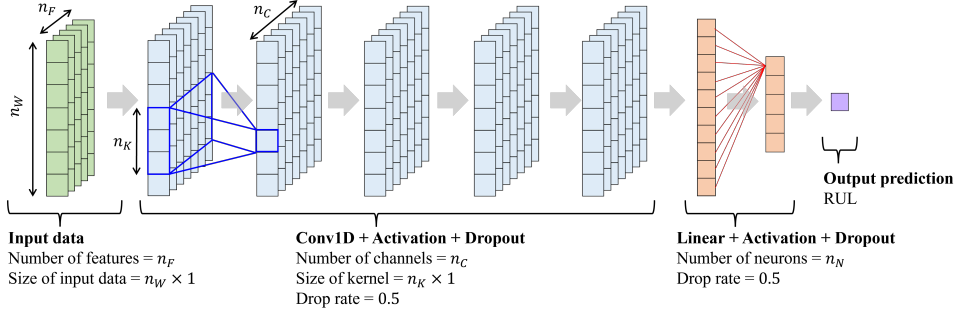


Figure 4.3: Proposed multi-channel CNN architecture. The blue lines visualize how multiple channels are convoluted into the next layer (see Equation (4.3)). The red lines visualize a forward pass of a linear layer (see Equation (4.4)).

Table 4.2: Architecture of the proposed CNN, where n_C and n_K are the number of output channels and the length of the kernel of Conv1D layers, respectively, and n_N is the number of output neurons of Linear layers. A dropout rate $\rho = 0.5$ is used for all layers.

Layer	Type	Layer parameters
1	Conv1D	$n_C = 128$, $n_K = 10$
2	Conv1D	$n_C = 64$, $n_K = 10$
3	Conv1D	$n_C = 32$, $n_K = 10$
4	Conv1D	$n_C = 16$, $n_K = 5$
5	Conv1D	$n_C = 8$, $n_K = 5$
6	Linear	$n_N = 256$
7	Linear	$n_N = 128$
8	Linear	$n_N = 1$

convoluted with different 1D kernels. Such multi-channel 1D convolutional layers are shown to be effective for multi-variate time-series data [26], which is also the case of the C-MAPSS data set. Since an independent kernel is used for time-series of each feature, convolutional layers are able to learn patterns of each feature.

A multi-channel 1D convolutional layer is defined by the size (length) n_K of the kernel, and the number of output channels n_C . Let the l^{th} convolutional layer get input $x^{(l-1)}$ from $(l-1)^{\text{th}}$ layer, where $x^{(l-1)}$ has $n_C^{(l-1)}$ channels. Then, the output of channel c of l^{th} convolutional layer is obtained as follows:

$$x_c^l = g^l \left(b_c^l + \sum_{c'=1}^{n_C^{(l-1)}} \kappa_{c,c'}^l * x_{c'}^{(l-1)} \right) \text{ for } c \in 1, \dots, n_C^l, \quad (4.3)$$

where $*$ is the convolutional operator, $\kappa_{c,c'}^l$ is the kernel for input channel c' and output channel c , b_c^l is the bias of output channel c , and $g^l(\cdot)$ is the activation function of the convolutional layer. Here we use the rectified linear unit (ReLU) activation function.

We use 5 convolutional layers. For the first convolutional layer, the input x^0 is x defined in Equation (4.2), and the number of input channel n_C^0 is equal to the number of features $n_F = 21$. Table 4.2 shows the number of output channels (n_C) and the size of the kernels (n_K) of all convolutional layers. For all convolutional layers we use zero padding to ensure the same size of the outputs.

After the convolutional layers, we apply two intermediate linear layers, and one output linear layer with a single neuron without activation (see Figure 4.3). Let the l^{th} linear layer get (flattened) input $x^{(l-1)}$. Then, its output is obtained as follows:

$$x^l = g^l \left(b^l + w^l x^{(l-1)} \right), \quad (4.4)$$

where w^l is the weight matrix, b^l is the bias, and $g(\cdot)$ is the ReLU activation function. We denote the number of output neurons of the linear layers as n_N (see Table 4.2).

Using the Adam optimizer [27], we optimize the kernels $\kappa_{c,c'}^l$ and the bias b_c^l of the convolutional layers, as well as the weights w^l and the bias b^l of the linear layers. The loss function considered here is the mean-squared-error. We train the network using a fixed learning rate of 0.001, a mini-batch of 256 samples, and a maximum of 10^3 training epochs.

MONTE CARLO DROPOUT

Typically, Monte Carlo dropout is used only during training to prevent overfitting [28]. We use Monte Carlo dropout i) during training to prevent overfitting of the model, and ii) during testing to obtain the probability distribution of the RUL [29]. We apply Monte Carlo dropout after each layer, using a dropping rate of 0.5.

4.2.3. RESULTS: PROBABILISTIC RUL PROGNOSTICS FOR TURBOFAN ENGINES

We first compare the accuracy of our results against other RUL prognostics models [8, 9, 10, 11, 12]. These models estimate RUL only as a point estimate, while our results are the estimated distribution of RUL. To be able to compare our results with these studies, we consider the Root Mean Squared Error (RMSE) between the *mean* of the estimated distribution of RUL and the true RUL. Table 4.3 shows the RMSE obtained for the testing instances of each data subsets. In general, the RMSE of subset FD002 and FD004 are higher than that of FD001. This is due to the multiple operating conditions considered in FD002 and FD004 (see Table 4.1). Also, FD002 and FD004 have the shortest time window of the input data compared to FD001 and FD003 ($n_W = 21$ for FD002 and $n_W = 19$ for FD004).

Table 4.3 shows that our multi-channel CNN with Monte Carlo dropout outperforms several other studies that employ CNNs for RUL prognostics. In fact, we obtained the lowest RMSE for subsets FD002 and FD004. For subsets FD001 and FD003, only MS-DCNN achieves a slightly smaller RMSE compared to our approach [10]. In general, the accuracy of our prognostics is higher or comparable to other existing studies.

Table 4.3: RMSE of the RUL predictions of the C-MAPSS data subsets using the proposed architecture of multi-channel CNN with Monte Carlo dropout and the other studies.

	FD001	FD002	FD003	FD004
Multi-channel CNN with MC dropout, proposed in this chapter	11.70	14.24	11.87	17.78
Single-channel CNN with MC dropout [8]	12.22	15.07	12.47	18.10
DCNN [9]	12.61	22.36	12.64	23.31
MS-DCNN [10]	11.44	19.35	11.67	22.22
CNN with pooling [11]	18.45	30.29	19.82	29.16
CNN with pyramid pooling [12]	12.64	25.92	12.39	26.84

4

ESTIMATING THE DISTRIBUTION OF RUL

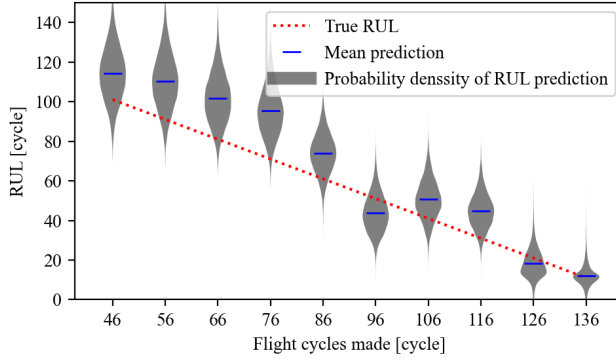
We are interested in quantifying the distribution of the engines' RUL, with the ultimate goal of informing maintenance decisions. Using Monte Carlo dropout, we generate the distribution of the RUL. We update the RUL distribution after every cycle, as new degradation data become available.

We illustrate the probabilistic RUL prognostics for three turbofan engines in testing instances of subset FD002. Figure 4.4 shows the evolution of the estimated RUL distribution over time, as more sensor measurements become available. Figure 4.5 shows the estimated RUL distribution of the three engines after they are operated for 136, 96, and 107 flight cycles, respectively. For Engine 148 (Figure 4.4a), the distribution of RUL is tightened across a sequence of flight cycles. After 136 flight cycles (Figure 4.5a), the error between the mean-estimated RUL and the true RUL is small (1.00 cycles), and the RUL distribution is concentrated around the true RUL (the standard deviation is 6.31 cycles).

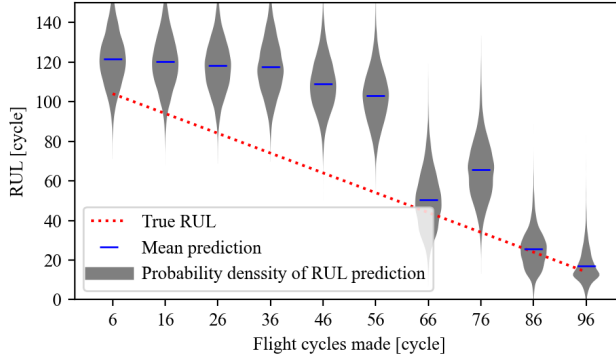
For Engine 173 (Figure 4.5b), the RUL distribution is right-skewed and the error between the mean-estimated RUL and the true RUL is small (2.87 cycles). Although the error of the estimated *point* (mean) of RUL is small, having the distribution of RUL provides additional support in maintenance decisions. Should we consider for Engine 173 only the mean prediction of RUL (16.87 cycles) to schedule a replacement, then we would be inclined to schedule a replacement close to 16 cycles. However, this maintenance decision would lead to an engine failure since the true RUL of Engine 173 is 14 cycles. Should we consider the estimated *distribution* of RUL for Engine 173 (Figure 4.5b), then we would observe the high probability (more than 45%) that Engine 173 fails in less than 14 cycles. In fact, the probability of Engine 173 failing at 12th cycle is highest (8.0%). Observing the RUL distribution we would be inclined to replace the engine close to 12 cycles, and we can avoid an engine failure.

For Engine 021 (Figure 4.5c), the error between the mean RUL prediction and the true RUL is large (26.93 cycles), and the RUL distribution is wide (standard deviation of 11.44). This is also informative for maintenance decision making, i.e., the accuracy of the RUL prognostic is low. In fact, the RUL distribution is wide across a sequence of flight cycles (see Figure 4.4c), leaving maintenance decision making conservative about the moment of engine replacement.

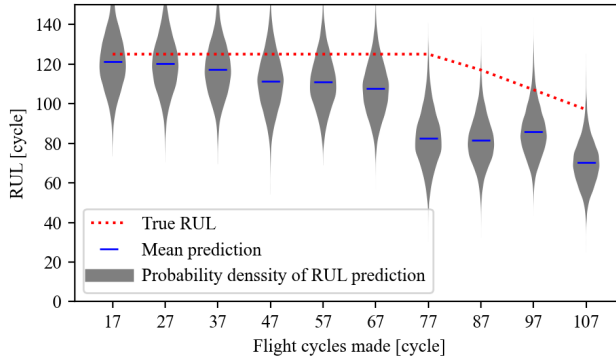
As shown in Figures 4.5 and 4.4, the distribution of RUL prognostics provides valu-



(a) FD002 Testing Engine 148

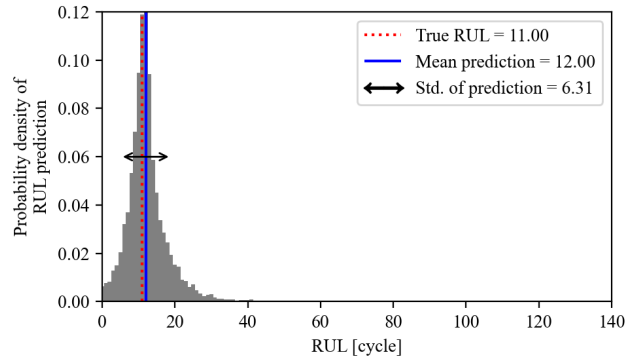


(b) FD002 Testing Engine 173

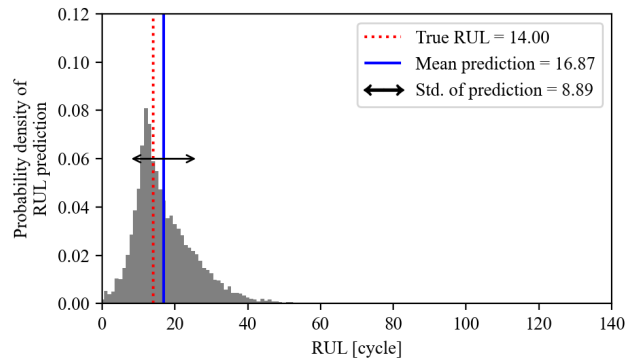


(c) FD002 Testing Engine 021

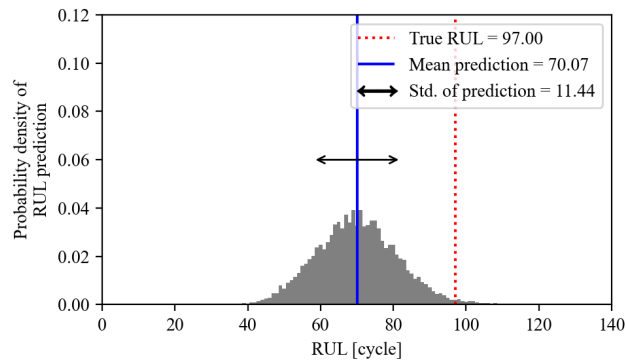
Figure 4.4: Evolution of the RUL distribution over time. (a) The mean-estimated RUL gets closer to the true RUL, and the variance decreases. (b) The mean-estimated RUL gets closer to the true RUL, and the variance decreases though it is skewed. (c) Neither the error of the mean-estimated RUL nor the variance decrease.



(a) FD002 Testing Engine 148



(b) FD002 Testing Engine 173



(c) FD002 Testing Engine 021

Figure 4.5: Probabilistic RUL prognostics. (a) The error between the mean-estimated RUL and the true RUL is small, and the RUL distribution is narrow and centered around the true RUL. (b) The mean-estimated RUL is slightly larger than the true RUL, and the RUL distribution is right-skewed. (c) The error between the mean-estimated RUL and the true RUL is large, and the RUL distribution is wide.

able information that can lead to more efficient maintenance decisions. In the next section, we propose a deep reinforcement learning approach to specify the moment of engine replacement based on the estimated distribution of RUL.

4.3. PLANNING PREDICTIVE AIRCRAFT MAINTENANCE USING DEEP REINFORCEMENT LEARNING AND PROBABILISTIC RUL PROGNOSTICS

In this section, we propose a deep reinforcement learning (DRL) approach for predictive maintenance of turbofan engines taking into account probabilistic RUL prognostics (estimated RUL distribution). These probabilistic RUL prognostics are updated periodically, as more measurements become available.

4.3.1. SCHEDULING ENGINE REPLACEMENTS CONSIDERING UPDATED PROBABILISTIC RUL PROGNOSTICS

The maintenance schedule of aircraft engines is updated every D flight cycles. In other words, every D cycles, we need to decide whether to replace an engine during the next D cycles (a *decision step*). Some existing studies assume that maintenance schedules can be updated every 1 cycle/day ($D = 1$) [17]. However, this assumption would be unrealistic for the maintenance of aircraft engines because it needs to be scheduled several cycles/days in advance to prepare required equipment [8]. Thus, we assume $D > 1$ and make predictive maintenance plan for the next D cycles.

Our aim is to minimize the total maintenance cost while avoiding engine failures and minimizing the waste of useful life of engines. If an engine is replaced too late and as a result this engine fails before the *scheduled replacement*, then we have to perform a very costly *unscheduled replacement* [19]. On the other hand, if we schedule a replacement too early, we waste the useful life of this engine. The long-run maintenance cost also increases when engines are replaced often. Our goal is to propose an approach to optimally schedule engine replacement taking into account probabilistic RUL prognostics (estimates of the RUL distribution).

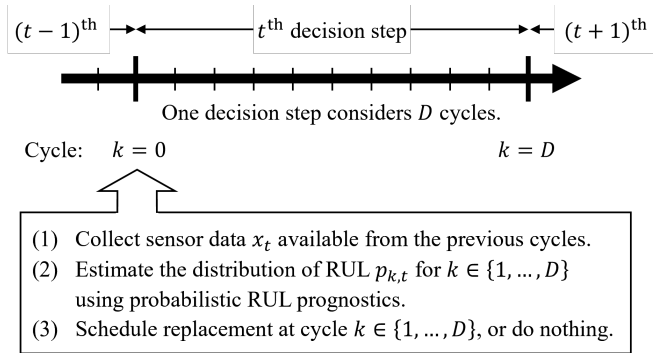


Figure 4.6: Maintenance planning at t^{th} decision step.

Figure 4.6 illustrates the maintenance planning at a decision step t . At the start of decision step t , we make use of sensor measurements x_t available up to this decision step t . Using x_t , we estimate the distribution of the engine RUL using a CNN with Monte Carlo dropout (see Section 4.2). Let $p_{k,t}$ denote the estimated cumulative probability that the RUL of the engine is less than k cycles, given the available sensor measurements x_t . Formally,

$$p_{k,t} = P(R_t \leq k | x_t), \text{ for } k \in \{1, \dots, D\} \quad (4.5)$$

where R_t is the hidden, true RUL of the engine at the start of decision step t . Based on $p_{k,t}$, at decision step t we decide whether to schedule an engine replacement after k cycles ($k \in \{1, \dots, D\}$), or do nothing within the next D cycles. If we do not schedule any replacement in the next D cycles, then we only collect further sensor measurements during these cycles. At the beginning of the $(t+1)^{\text{th}}$ decision step, these measurements are used, together with a CNN, to update the estimated distribution of RUL, $p_{k,(t+1)}$.

For example, at the start of decision step t , we need to decide when to schedule replacement based on $p_{k,t}$ given in Figure 4.7. Figure 4.7 is the estimated RUL distribution for FD002 Testing Engine 148 of the the C-MAPSS data set. This cumulative probability $p_{k,t}$ is estimated after 136 cycles of usage. Our prognostics model predicts that the probabilities that this engine will fail within 10 and 15 cycles are 37% and 82%, respectively. In fact, the true RUL of this engine at this moment is 11 cycles, and our prognostics model predicts that this engine will fail within 11 cycles with probability 48%. The distribution in Figure 4.7 contains information of uncertainty useful for optimal maintenance planning, but it is not straightforward to be used by a human without enough experience and knowledge. Therefore, we propose a deep reinforcement learning approach to make an optimal replacement schedule using the distribution of RUL.

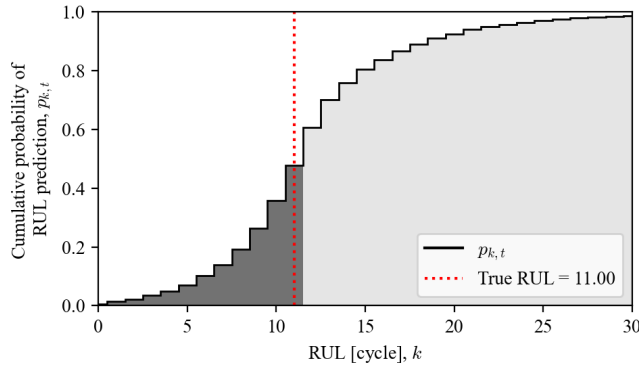


Figure 4.7: Estimated cumulative probability $p_{k,t}$ for FD002 Testing Engine 148 after it is operated for 136 cycles.

4.3.2. FORMULATING PREDICTIVE MAINTENANCE PLANNING AS A DEEP REINFORCEMENT LEARNING PROBLEM

We formulate the predictive maintenance planning for an engine as a deep reinforcement learning (DRL) problem (see Figure 4.8). The hidden state of the engine is R_t , the true RUL of the engine at decision step t , but this is not observable. Instead, the observed state s_t of the engine is the RUL distribution estimated by the sensor measurements and the prognostics using CNN. Given this observation s_t , an agent (decision-maker) takes an action $a_t \in \mathcal{A}$ based on the *policy*. Then, a reward r_t is given based on the hidden state R_t and the action a_t . Finally, the system transits from state s_t to s_{t+1} at the next decision step ($t + 1$). We formalize our DRL problem as follows.

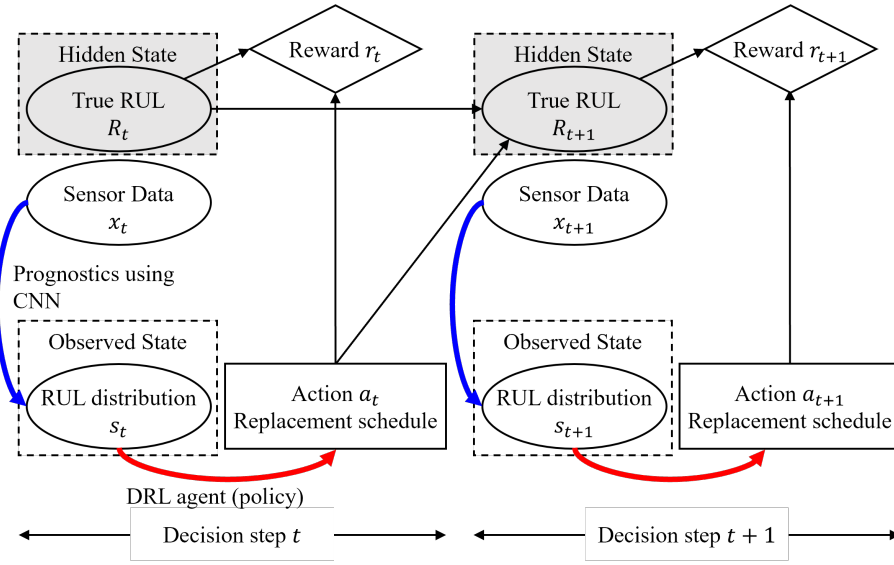


Figure 4.8: Illustration of states, actions, rewards and transitions of the DRL problem for predictive maintenance planning.

The observed state s_t is the estimated distribution of the RUL $p_{k,t}$ for the next D cycles, i.e., $k \in \{1, \dots, D\}$. Formally,

$$s_t = [p_{1,t}, \dots, p_{D,t}], \quad (4.6)$$

where $p_{k,t}$ is the probability that the RUL of the engine is less than k cycles (see Equation (4.5)).

Given the state s_t , the agent choose an action a_t : either schedule a replacement of the engine at cycle k ($k \in \{1, \dots, D\}$), or Do nothing. Formally,

$$a_t = \begin{cases} k, & 0 < k \leq D \quad \text{Schedule replacement at cycle } k, \\ M, & M > D \quad \text{Do nothing} \end{cases}, \quad (4.7)$$

where $a_t = M > D$ implies that we do not schedule replacement in the next D cycles and postpone the engine replacement to the next decision step ($t + 1$).

The reward r_t obtained at decision step t is defined for 4 cases considering the action a_t and the hidden state R_t : (1) a replacement is scheduled earlier than the engine failure; (2) a replacement is scheduled later than the engine failure; (3) we decide to do nothing in the next D cycles, but the engine fails within the next D cycles; and (4) we decide to do nothing, and the engine does not fail in the next D cycles. Formally,

$$r_t = \begin{cases} -c_{\text{sch}}(k) & \text{if } (k-1) < a_t \leq k \text{ and } R_t > k \\ -c_{\text{uns}} & \text{if } (k-1) < a_t \leq k \text{ and } R_t \leq k \\ -c_{\text{uns}} & \text{if } a_t > D \text{ and } R_t \leq D \\ 0 & \text{if } a_t > D \text{ and } R_t > D \end{cases} \quad (4.8)$$

Here, $c_{\text{sch}}(k)$ denotes the cost of a scheduled replacement at cycle k ($k \in \{1, \dots, D\}$), which is defined as follows:

$$c_{\text{sch}}(k) = c_0 - c_1 k, \quad (4.9)$$

where c_0 is a fixed cost of replacement and c_1 is the cost that depends on the scheduled cycle k . We assume that the replacement scheduled earlier is expensive because we need less time to prepare the required equipment, i.e., $c_1 > 0$ [19]. Also, $c_0 - c_1 D > 0$. In Equation (4.8), c_{uns} denotes the cost of unscheduled replacement, where we assume $c_{\text{uns}} > c_0$ because unscheduled replacement is generally more expensive [19]. Figure 4.9 plots the costs of scheduled and unscheduled replacements when $c_0 = 1$, $c_1 = 0.01$, and $c_{\text{uns}} = 2$.

The goal of the DRL agent is to choose an optimal moment to schedule a replacement such that the expected rewards are maximized (or the expected costs are minimized). When scheduling an engine replacement, the agent takes into account the estimated probability of having an engine failure in the next D cycles (probabilistic RUL prognostics).

In the example in Figure 4.9, if we schedule an engine replacement at cycle 25 and this engine does not fail until then, then the engine replacement cost is 0.75. However,

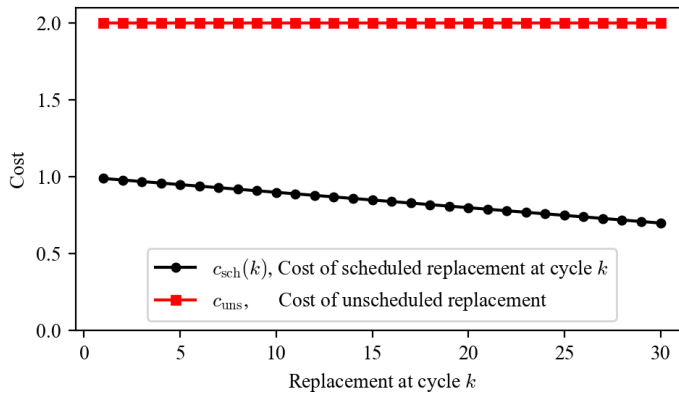


Figure 4.9: Cost model of scheduled and unscheduled replacements. We assume $c_0 = 1$, $c_1 = 0.01$, $c_{\text{uns}} = 2$, and $D = 30$.

there is a risk of having this engine fail before 25 cycles, which leads to unscheduled replacement at a higher cost of 2.0, instead of 0.75. In the case of Engine 148, the probability that this engine fails before 25 cycles is more than 97% (see Figure 4.7). On the other hand, if we schedule a replacement at cycle 5, the cost will be 0.95. Although this cost is higher than a replacement scheduled at cycle 25, this maintenance action reduces the risk of an expensive engine failure. In the case of Engine 148, the estimated failure probability is less than 7% at cycle 5. In general, it is not trivial to choose an optimal moment to replace an engine, given the estimated RUL distribution (Figure 4.7) and the cost model (Figure 4.9).

Once the DRL agent chooses an action, the hidden state (true RUL) and the observed state (RUL distribution, s_{t+1}) are updated accordingly. If the engine is replaced at decision step t , then the next decision step considers a new engine from the C-MAPSS dataset, with its initial sensor measurements and initial true RUL. Otherwise, we further obtain sensor measurements during the next D cycles and update the distribution of the RUL (the next state s_{t+1}) by generating new RUL prognostics using a CNN with Monte Carlo dropout.

4.3.3. TRAINING DRL AGENT FOR PREDICTIVE MAINTENANCE PLANNING

The DRL agent chooses action a_t (maintenance decision) for a given state s_t (estimated distribution of RUL) based on a *policy* $\pi(a_t|s_t) : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$, which is the probability to choose action a_t for a given state s_t . The optimal policy π^* is defined as a policy that maximizes the expected reward defined as follows:

$$J(\pi) = \sum_t \mathbb{E}_{(s_t, a_t) \sim \rho_\pi} [\gamma^t r_t(s_t, a_t)], \quad (4.10)$$

where γ is a discount factor, and $\rho_\pi(s_t, a_t)$ is the state-action trajectory distribution induced by a policy π [30].

SOFT-ACTOR-CRITIC ALGORITHM TO TRAIN THE DRL AGENT FOR PREDICTIVE MAINTENANCE PLANNING

We train the DRL agent using a Soft-Actor-Critic (SAC) algorithm [30]. The SAC algorithm is an actor-critic algorithm where a policy (actor) is trained to choose actions that maximizes the estimated state-action value (critic). Compared to traditional actor-critic algorithms, the SAC uses a stochastic policy and maximizes a soft objective to explore new policies.

We consider a stochastic policy $\pi_\phi(a_t|s_t)$ to determine the mean $f_\phi^\mu(s_t)$ and the standard deviation $f_\phi^\sigma(s_t)$ of action for a given state s_t , where ϕ is the trainable parameters of f_ϕ^μ and f_ϕ^σ . Then, an action a_t is chosen as follows:

$$a_t = f_\phi^\mu(s_t) + \epsilon_t \cdot f_\phi^\sigma(s_t), \quad (4.11)$$

where ϵ_t is sampled from a spherical Gaussian.

The considered soft objective includes the expected entropy of the policy π_ϕ . Formally,

$$J(\pi) = \sum_t \mathbb{E}_{(s_t, a_t) \sim \rho_\pi} \gamma^t [r_t(s_t, a_t) + \alpha \mathcal{H}(\pi(\cdot|s_t))], \quad (4.12)$$

where α is the temperature parameter determining the relative importance between the entropy term and the reward term. Thus, the SAC algorithm simultaneously maximizes the expected reward and the entropy of the policy, allowing the exploration of new policies.

Considering the soft objective in Equation (4.12), the state-action value (Q function) is modified as the soft Q function $Q : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$. This soft Q function is then obtained by iteratively applying the following modified Bellman backup operator \mathcal{T}^π [30] :

$$\mathcal{T}^\pi Q(s_t, a_t) = r_t(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1} \sim p} [V(s_{t+1})], \quad (4.13)$$

where p is the distribution of s_{t+1} , given s_t and a_t , and $V(s_t)$ is the soft state value function $V : \mathcal{S} \rightarrow \mathbb{R}$ defined as follows:

$$V(s_t) = \mathbb{E}_{a_t \sim \pi} [Q(s_t, a_t) - \alpha \log \pi(a_t | s_t)]. \quad (4.14)$$

In the SAC algorithm, we train three functions: the policy (π), the soft Q function (Q), and the soft value function (V). We model these functions by means of three deep neural networks, π_ϕ , Q_θ , and V_ψ , where ϕ , θ , and ψ are the trainable parameters of each neural network. During training, we collect the replay buffer $\mathcal{D} = \{(s_t, a_t, r_t, s_{t+1})\}$ based on the current policy π_ϕ . Then, we update the trainable parameters to minimize the following loss functions.

The policy net π_ϕ is updated using the Kullback-Leibler (KL) divergence, which guarantees the improvement of the policy in terms of its soft value [30]. We minimize the expected KL divergence as follows:

$$\begin{aligned} J_\pi(\phi) &= \mathbb{E}_{s_t \sim \mathcal{D}} \left[D_{KL} \left(\pi_\phi(\cdot | s_t) \left\| \frac{\exp(\frac{1}{\alpha} Q_\theta(s_t, \cdot))}{Z_\theta(s_t)} \right\| \right) \right] \\ &= \mathbb{E}_{s_t \sim \mathcal{D}, a_t \sim \pi_\phi} \left[\log \pi_\phi(a_t | s_t) - \frac{1}{\alpha} Q_\theta(s_t, a_t) + \log Z_\theta(s_t) \right], \end{aligned} \quad (4.15)$$

where $Z_\theta(s_t)$ is the partition function that does not contribute to the gradient with respect to ϕ , and a_t is sampled from the current policy π_ϕ using Equation (4.11).

For the value net V_ψ , we minimize the residual of the value function calculated based on the critic net Q_θ :

$$J_V(\psi) = \mathbb{E}_{s_t \sim \mathcal{D}} \left[\frac{1}{2} (V_\psi(s_t) - \hat{V}(s_t))^2 \right], \quad (4.16)$$

with

$$\hat{V}(s_t) = \mathbb{E}_{a_t \sim \pi_\phi} [Q_\theta(s_t, a_t) - \alpha \log \pi_\phi(a_t | s_t)]. \quad (4.17)$$

For the critic net Q_θ , we minimize the modified Bellman residual:

$$J_Q(\theta) = \mathbb{E}_{(s_t, a_t) \sim \mathcal{D}} \left[\frac{1}{2} (Q_\theta(s_t, a_t) - \hat{Q}(s_t, a_t))^2 \right], \quad (4.18)$$

with

$$\hat{Q}(s_t, a_t) = r_t(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1} \sim p} [V_{\hat{\psi}}(s_{t+1})]. \quad (4.19)$$

Algorithm 1 Soft-Actor-Critic algorithm for predictive maintenance planning.

```

1: Initialize parameters  $\phi, \psi, \tilde{\psi}, \theta_1, \theta_2$ .
2: for each episode do
3:   Initialize observation  $s_0$ .
4:   for each decision step  $t$  do
5:     Choose action  $a_t$  (Equation (4.11))
6:     Get reward  $r_t$  (Equation (4.8))
7:     Get next state  $s_{t+1}$  (Equation (4.6))
8:     Update replay buffer  $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r_t, s_{t+1})\}$ 
9:     for each learning step do
10:      Sample mini-batch  $\tilde{\mathcal{D}}$  from replay buffer  $\mathcal{D}$ 
11:       $\psi \leftarrow \psi - \lambda_V \nabla_{\psi} J_V(\psi)$ 
12:       $\theta_q \leftarrow \theta_q - \lambda_Q \nabla_{\theta_q} J_Q(\theta_q)$  for  $q \in \{1, 2\}$ 
13:       $\phi \leftarrow \phi - \lambda_{\pi} \nabla_{\phi} J_{\pi}(\phi)$ 
14:       $\tilde{\psi} \leftarrow \tau \psi + (1 - \tau) \tilde{\psi}$ 
15:    end for
16:    Update  $s_t$ 
17:  end for
18: end for

```

Here, we use the target value net $V_{\tilde{\psi}}$, where $\tilde{\psi}$ is an exponentially moving average of the value net parameters [31]. Also, we adopt the double Q-learning approach: we simultaneously train two critic nets (Q_{θ_1} and Q_{θ_2}), and we use $Q_{\theta}(s_t, a_t) = \min\{Q_{\theta_1}(s_t, a_t), Q_{\theta_2}(s_t, a_t)\}$ [32]. Both the target value net and double Q-learning approach are known to stabilize the training process [31, 32]

The gradients of the loss functions in Equations (4.15), (4.16), and (4.18) are obtained by backward propagation. Given the gradients of the corresponding objectives, the parameters ϕ , ψ , and θ are updated using the Adam optimizer with learning rates λ_{π} , λ_V , and λ_Q , respectively.

We train the DRL agent for predictive maintenance for aircraft engines using the SAC algorithm (see Algorithm 1). We first initialize the parameters of the neural network models, ϕ , ψ , $\tilde{\psi}$, θ , θ_1 , and θ_2 (line 1). We train these networks for n_E episodes (line 2). An episode is initialized with observation s_0 , which is the initial distribution of RUL of an engine sampled from a training data set of the DRL agent (line 3). Here, the RUL distribution is estimated using the CNN that is already trained based on an independent training data set. The episode continues for n_T decision steps (line 4). At each decision step t , for the observed state s_t , we sample the action a_t using the policy net $\pi_{\phi}(a_t|s_t)$ (line 5). Based on this action, we obtain a reward r_t and the next state s_{t+1} (line 6-7). We add (s_t, a_t, r_t, s_{t+1}) into the replay buffer \mathcal{D} (line 8). Then, for each learning step, we sample mini-batch $\tilde{\mathcal{D}}$ from replay buffer \mathcal{D} (line 9-10), and use this to calculate the loss functions in Equations (4.15), (4.16), and (4.18). We next update the policy net, the value net, and the critic nets such that the corresponding objectives are minimized (line 11-13). Here λ_{π} , λ_V , and λ_Q are the learning rates of each network. Also, the parameters of the target value net $\tilde{\psi}$ is updated with an exponential moving average of ψ , where τ is a

smoothing factor (line 14). For the next decision step, we update the current state (line 16).

DESIGNING THE ARCHITECTURE OF THE NEURAL NETWORKS

We design the neural network architecture for the policy net π_ϕ , the value net V_ψ , and the critic net Q_θ as shown in Figure 4.10 and Table 4.4.

The policy net π_ϕ has input s_t , which is a vector of size D , and returns two scalar values corresponding to the mean of action $f_\phi^\mu(s_t)$ and the standard deviation of action

Table 4.4: Architecture of the deep neural network models for policy net π_ϕ , value net V_ψ , and critic net Q_θ . (see also Figure 4.10)

Policy net π_ϕ		
Layer	Type	Number of neurons n_N
Input (State s_t)	-	D
Shared hidden layer 1	Linear	$n_N = 256$
Shared hidden layer 2	Linear	$n_N = 128$
Hidden layer 1 for μ	Linear	$n_N = 64$
Hidden layer 2 for μ	Linear	$n_N = 32$
Output $\mu (f_\phi^\mu(s_t))$	Linear	1
Hidden layer 1 for σ	Linear	$n_N = 64$
Hidden layer 2 for σ	Linear	$n_N = 32$
Output $\sigma (f_\phi^\sigma(s_t))$	Linear	1

Value net V_ψ		
Layer	Type	Number of neurons n_N
Input (State s_t)	-	D
Hidden layer 1	Linear	$n_N = 256$
Hidden layer 2	Linear	$n_N = 128$
Output (State value $V(s_t)$)	Linear	1

Critic net Q_θ		
Layer	Type	Number of neurons n_N
Input (State s_t , action a_t)	-	$D + 1$
Hidden layer 1	Linear	$n_N = 256$
Hidden layer 2	Linear	$n_N = 128$
Output (Q value $Q(s_t, a_t)$)	Linear	1

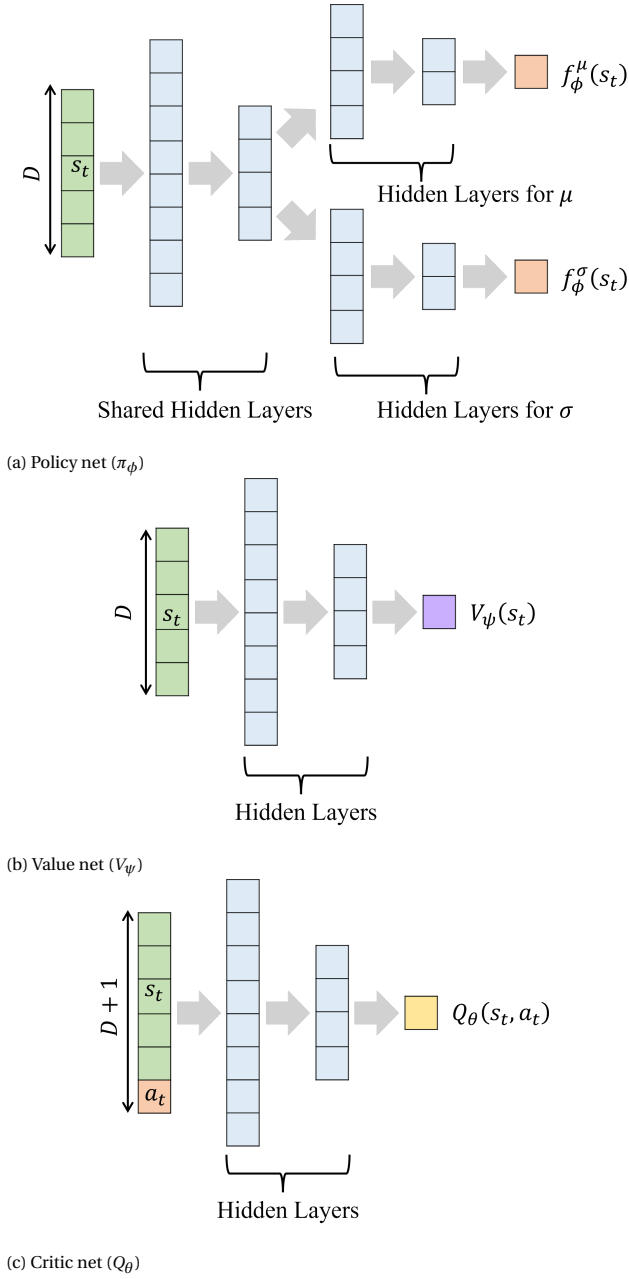


Figure 4.10: Architecture of neural network models for DRL. (a) Policy net (π_ϕ). (b) Value net (V_ψ). (c) Critic net (Q_θ). (see also Table 4.4)

$f_{\phi}^{\sigma}(s_t)$. These two outputs $f_{\phi}^{\mu}(s_t)$ and $f_{\phi}^{\sigma}(s_t)$ are used to sample action a_t for the given state s_t (see Equation (4.11)). We consider hidden layers shared by $f_{\phi}^{\mu}(s_t)$ and $f_{\phi}^{\sigma}(s_t)$ to facilitate learning from the shared features (see Figure 4.10a). Following these shared hidden layers, we consider separated hidden layers for each of $f_{\phi}^{\mu}(s_t)$ and $f_{\phi}^{\sigma}(s_t)$.

The value net V_{ψ} has input s_t and returns a scalar value $V_{\psi}(s_t)$. We consider two hidden, fully-connected layers. The same architecture is used also for the target value net $V_{\bar{\psi}}$.

The input of critic net Q_{θ} is a vector of size $(D+1)$, which is the augmentation of state s_t and action a_t . Its output is a scalar value $Q_{\theta}(s_t, a_t)$. We consider two hidden, fully connected layers (see Table 4.4). As we use a double Q-learning approach, we consider two critic networks Q_{θ_1} and Q_{θ_2} having the same architecture but different parameters θ_1 and θ_2 .

4

4.4. RESULTS: DRL FOR PREDICTIVE AIRCRAFT MAINTENANCE WITH PROBABILISTIC RUL PROGNOSTICS

4.4.1. TRAINING DRL AGENT

We consider the maintenance of a turbofan engine whose degradation measurements are given in subset FD002 of the C-MAPSS data set. From the 260 training instances of FD002, we randomly sample 130 engines as *training set for RUL prognostics* (see Section 4.2 for our CNN with Monte Carlo dropout to estimate the RUL distribution of engines). The remaining 130 engines are used as *training set for the DRL agent* (see Section 4.3 for our DRL approach that considers RUL distributions to plan engine replacements).

We train the DRL agent for $n_E = 5000$ episodes, where each episode consists of maximum $n_T = 100$ decision steps. Each decision step considers $D = 30$ flight cycles (see Figure 4.6 for the definition of a decision step). As a reward (cost) model, we assume $c_{\text{uns}} = 2$, $c_0 = 1$, and $c_1 = 0.01$ for the cost parameters defined in Equations (4.8) - (4.9) (see also Figure 4.9 for the cost model). The hyper-parameters of the SAC algorithms are as follows: discount factor $\gamma = 0.9$, temperature parameter $\alpha = 0.01$, learning rates $\lambda_{\phi} = 10^{-5}$, $\lambda_{\psi} = 10^{-4}$, $\lambda_{\theta} = 10^{-4}$, smoothing factor of target value net $\tau = 10^{-3}$, the maximum size of replay buffer $|\mathcal{D}| = 10^6$, and the size of mini-batch $|\hat{\mathcal{D}}| = 4096$.

Figure 4.11 shows the learning curve of the DRL approach, illustrating the total reward per episode. The total reward rapidly increases during the first 500 episodes and converges to around -18 after 1000 episodes. After 1000 episodes, the total reward of each episode varies because the considered DRL problem is stochastic, i.e., the initial value of true RUL is random and the probability distribution of RUL is estimated by Monte Carlo dropout of the CNN model. However, the moving average of the total reward stabilizes during 1000-5000 episodes. Moreover, 5 independent training curves show the same trends. Thus, the training is stopped after 5000 episodes.

4.4.2. PREDICTIVE MAINTENANCE USING DRL

Following training, we evaluate the trained DRL agent for 1000 episodes. During evaluation, the DRL agent chooses an action a_t for a given state s_t from the mean action f_{ϕ}^{μ} of

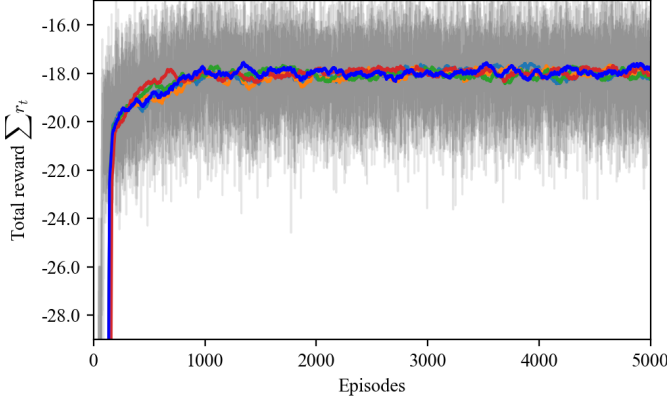


Figure 4.11: Learning curve of the DRL approach during 5000 episodes. The thin grey lines are 5 learning curves, and the solid lines are the moving average of 100 episodes of each learning curve.

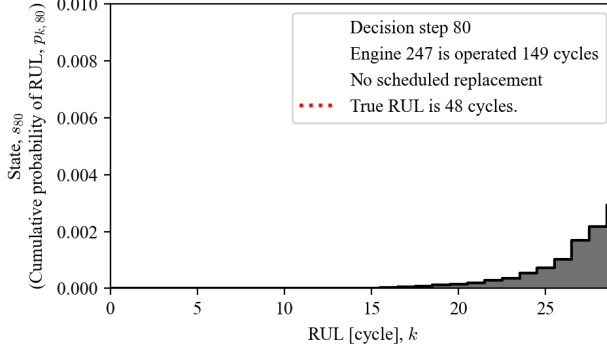
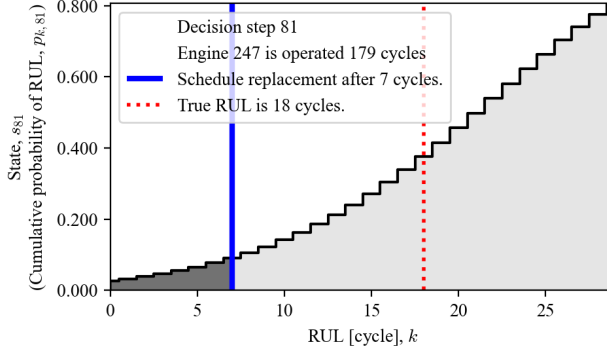
the trained policy π_ϕ . Formally,

$$a_t = f_\phi^\mu(s_t). \quad (4.20)$$

Below we discuss the benefits of our DRL approach for predictive maintenance, by presenting some decision steps (the estimated RUL distributions and the associated maintenance actions made by the DRL agent).

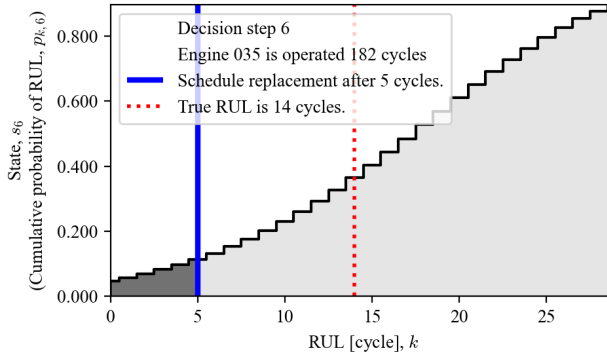
MAINTENANCE DECISION BASED ON UPDATED RUL DISTRIBUTION

The estimated RUL distributions are updated at every decision step t (D cycles), as more sensor measurements become available. This ensures that the maintenance decision is always based on the most recent RUL prognostics. Figure 4.12 shows 2 consecutive decisions steps ($t = 80$ and 81) for the maintenance of Engine 247, FD002 of C-MAPSS data set. At decision step $t = 80$, the engine has been operated for 149 cycles. Our CNN model predicts the probability that the engine will fail within 30 cycles to be $p_{30,80} = 0.005$. The entire distribution $p_{k,80}$ for $k \in \{0, \dots, 30\}$ is given in Figure 4.12a. Such a distribution quantifies the uncertainty of the RUL prognostics, and provides basis for maintenance decisions of the DRL agent. The DRL agent observes the RUL distribution and decides to do nothing, i.e., do not schedule replacement in this decision step $t = 80$. Since no replacement is scheduled for the next $D = 30$ cycles, the engine is operated continuously until the next decision step $t = 81$, and more sensor measurements are collected. Based on the new sensor measurement, we update the distribution of the RUL again using the CNN with Monte Carlo dropout (see Figure 4.12b). At decision step $t = 81$, the probability that the engine will fail within 30 cycles is estimated to be $p_{30,81} = 0.807$. Given the updated distribution of the RUL, the DRL agent schedule a replacement after 7 cycles (see the blue vertical line in Figure 4.12b). The probability that the engine will fail within 7 cycles is $p_{7,80} = 0.091$. In fact, the (hidden) true RUL is 18 cycles at decision step $t = 81$, i.e., the DRL agent schedules a replacement 11 cycles before the engine fails.

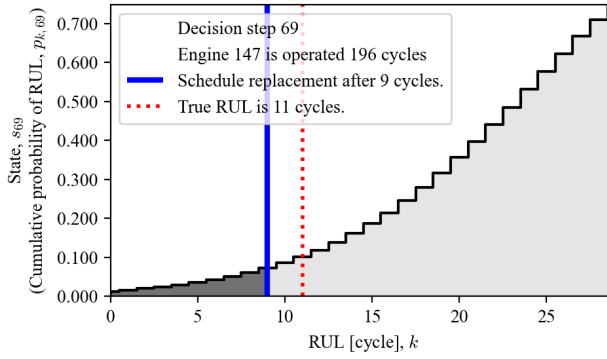
(a) Decision step $t = 80$, replacement is not scheduled.(b) Decision step $t = 81$, replacement is scheduled after 7 cycles.Figure 4.12: Estimated RUL distributions and replacement schedules for Engine 247 during 2 consecutive decisions steps ($t = 80$ and 81).

ADAPTIVE MAINTENANCE DECISION USING DEEP NEURAL NETWORK

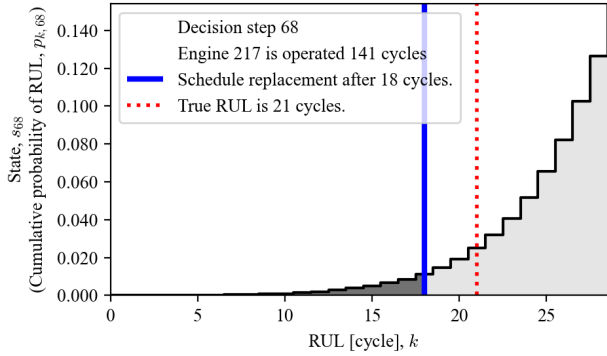
Using a deep neural network model (policy net), our DRL agent adaptively considers the updated RUL distribution of individual engine, instead of relying on one fixed threshold for all engines. As a result, our DRL agent can identify optimal moment of engine replacement taking into account different trends of RUL distributions. For example, Figure 4.13 shows distinctive RUL distributions of three engines, estimated during different episodes. In Figure 4.13a, there is a very high chance that the engine will fail within the next 30 cycles ($p_{30,t} = 0.896$). In this case, the DRL agent schedules a replacement after 5 cycles when the probability that the engine will fail within 5 cycles is estimated to be $p_{5,t} = 0.113$. In Figure 4.13b, the estimated $p_{k,t}$ increases from $p_{1,t} = 0.011$ to $p_{30,t} = 0.748$, and the DRL agent schedules a replacement after 9 cycles when $p_{9,t} = 0.073$. In the last case in Figure 4.13c, the probability that the engine will fail within 30 cycles is smaller compared to the two previous cases ($p_{30,t} < 0.15$), but the trend rapidly increases. In this case, the DRL agent schedules a replacement after 18 cycles when $p_{18,t} = 0.011$, effec-



(a) Replacement is scheduled after 5 cycles when $p_{5,6} = 0.113$



(b) Replacement is scheduled after 9 cycles when $p_{9,69} = 0.073$



(c) Replacement is scheduled after 18 cycles when $p_{18,68} = 0.011$

Figure 4.13: Three different RUL distributions and adaptive maintenance decisions of the DRL agent.

tively preventing the engine failure.

In contrast to our DRL approach, existing predictive maintenance approaches often consider fixed thresholds for all same-type of components to trigger maintenance. For example, in [8], an alarm is triggered when the estimated RULs of engines are below a threshold (44 days). Similarly, in [21], airframe panels are replaced when the predicted crack size is larger than a threshold (47.4mm). Since a fixed threshold value is applied for all components, differences between individual RUL prognostics results may not be considered.

The benefit of our adaptive maintenance planning using deep neural network is evident when trying, unsuccessfully, to find one fixed threshold that is optimal for all three cases in Figure 4.13. Let us assume that we use a fixed threshold 0.11 and always schedule an engine replacement after k cycles if $p_{k,t} > 0.11$, irrespective of the RUL distribution. Using such a fixed threshold of 0.11 will effectively prevent the failure for the first case (Figure 4.13a). Using the same threshold for Figures 4.13b and 4.13c, engine replacements are scheduled after 12 and 28 cycles, respectively. However, in both cases, the engine replacements are later than the true RUL (11 and 21 cycles, respectively), leading to unscheduled replacements at higher cost. In the same line, let us assume that we set a much lower fixed threshold of 0.01, i.e., we always schedule an engine replacement after k cycles if $p_{k,t} > 0.01$. Using this threshold will avoid an unscheduled engine replacements in the last case (Figure 4.13c). However, with such a low threshold, replacements are scheduled too early for the other two cases in Figures 4.13a and 4.13b, wasting the useful life of the engines. This example shows that finding one fixed threshold that is optimal for all cases is challenging. In contrast, our DRL agent adaptively consider the different trends of RUL distributions, without using fixed thresholds. As a result, our approach leads to less unscheduled maintenance.

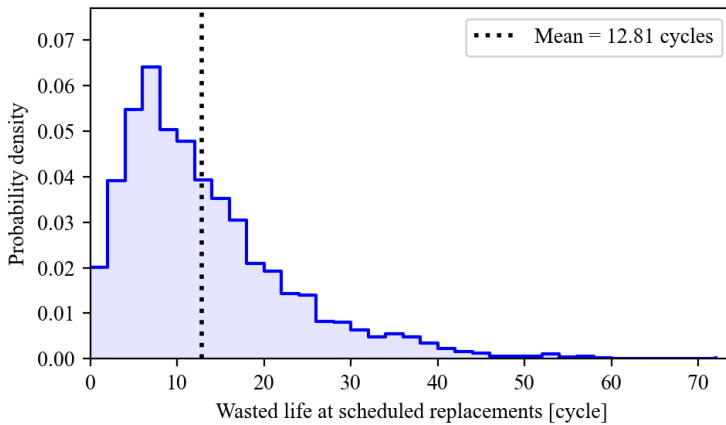


Figure 4.14: Wasted life of engines at the moment of replacements under our DRL approach.

SCHEDULING REPLACEMENTS WITH SMALL WASTED LIFE OF ENGINES

Using updated RUL distributions and adaptive maintenance decision, our DRL agent schedules engine replacements without wasting useful lives of engines. Figure 4.14 shows the distribution of the wasted life of engines at the moment of replacement, when using our DRL approach. On average, replacements are scheduled when the true RUL of an engine is 12.81 cycles. This is only 6% of the average life of the engines in subset FD002. Also, more than 82% of the engines are replaced when their wasted life is less than 20 cycles.

4.5. PREDICTIVE AIRCRAFT MAINTENANCE USING DRL VS. OTHER MAINTENANCE STRATEGIES

In this section, we compare the performance of our DRL approach for predictive maintenance against three other traditional maintenance strategies:

4

PREDICTIVE MAINTENANCE AT MEAN-ESTIMATED RUL

This strategy schedules engine replacements at the mean RUL predicted by the CNN model in Section 4.2. This strategy uses a point estimation of the RUL, while our DRL approach uses a distribution of the RUL. Considering this strategy, we aim to evaluate the impact of using the distribution of RUL for maintenance planning, rather than just a point estimation of the RUL.

CORRECTIVE MAINTENANCE

This strategy replaces engines as soon as they fail. Under this strategy, we always perform unscheduled replacements, which is the most undesirable case.

IDEAL MAINTENANCE AT TRUE RUL

This strategy assumes that the true RUL is known in advance by an Oracle, and engines replacements are scheduled exactly at this true RUL. Under this strategy, there are no unscheduled maintenance tasks and the wasted lives of engines are always zero, i.e., an ideal maintenance strategy.

Table 4.5 shows the performance of these traditional maintenance strategies vs. our DRL approach using three following performance indicators:

i) The total cost: this is the cost of both scheduled and unscheduled replacements during 3000 cycles of engine operations (i.e., 100 decision steps). The cost (reward) model is given in Equations (4.8)-(4.9).

ii) The number of unscheduled replacements: this is a direct metric for maintenance reliability. We aim to minimize the number of unscheduled engine replacements.

iii) The total number of replacements: this is the number of both scheduled and unscheduled replacements during 3000 cycles of engine operations. Since we consider a fixed period of cycles, a lower number of total replacements implies that we utilize the engines for a longer duration.

Table 4.5 shows that our DRL approach using RUL distributions outperforms the other maintenance strategies, especially in terms of the total maintenance cost and the

Table 4.5: Comparison of the proposed DRL approach using RUL distribution, and other maintenance strategies. Percentage in parenthesis indicates the relative ratio to the corrective maintenance.

	Total cost	Number of unscheduled replacements	Total number of replacements
DRL approach using distribution of RUL	17.84 (-36.3%)	0.62 (-95.6%)	14.89 (+6.4%)
Predictive maintenance at mean-estimated RUL	25.23 (-9.8%)	10.87 (-22.3%)	14.00 (0.0%)
Corrective maintenance	27.99 (0.0%)	13.99 (0.0%)	13.99 (0.0%)
Ideal maintenance at true RUL	16.10 (-42.5%)	0.0 (-100.0%)	13.95 (-0.3%)

4

number of unscheduled replacements. Our DRL approach saves 36.3% of the total costs compared to corrective maintenance. Moreover, it also achieves a more reliable maintenance planning by preventing 95.6% of unscheduled replacements. Also, the total number of replacements (both scheduled and unscheduled) is slightly (6.4%) larger for our DRL approach since engines are replaced earlier than their end-of-life to prevent unscheduled replacements. However, this slight increase in the total number of engine replacements is balanced out by a large economic efficiency (large cost savings) and maintenance reliability (lower number of unscheduled replacements) that our DRL approach achieves.

The benefit of using probabilistic RUL prognostics instead of point-RUL estimates is evident when comparing our DRL approach against predictive maintenance at mean-estimated RUL (see Table 4.5). Both strategies make use of RUL prognostics obtained using a CNN (see Section 4.2). But our DRL approach uses probabilistic RUL prognostics (estimated RUL distribution) to plan engine maintenance. As a result, predictive maintenance based on the mean-estimated RUL reduces only 9.8% of total costs and 22.3% of unscheduled replacements, while our DRL approach further reduces the total cost (36.3%) and unscheduled replacements (95.6%).

The cost savings obtained by our DRL approach are further explained in Figure 4.15. Since we assume 2 times higher costs for unscheduled replacements (see the cost model in Figure 4.9), even a small number of unscheduled replacements can take a large portion of the total cost. Due to this reason, all maintenance strategies performed a similar number of total replacements, but the total maintenance costs are significantly different. In the case of predictive maintenance at the mean-estimated RUL, 85% of the total cost is associated with unscheduled replacements. In contrast, for our DRL approach, only 7% of the total cost is associated with unscheduled replacements.

4.6. CONCLUSIONS

In this chapter, we propose a deep reinforcement learning (DRL) approach to plan predictive maintenance for aircraft engines. This maintenance planning takes into account the estimated distribution of the engines' Remaining-Useful-Life (RUL).

We first estimate the RUL distribution of engines using Convolutional Neural Net-

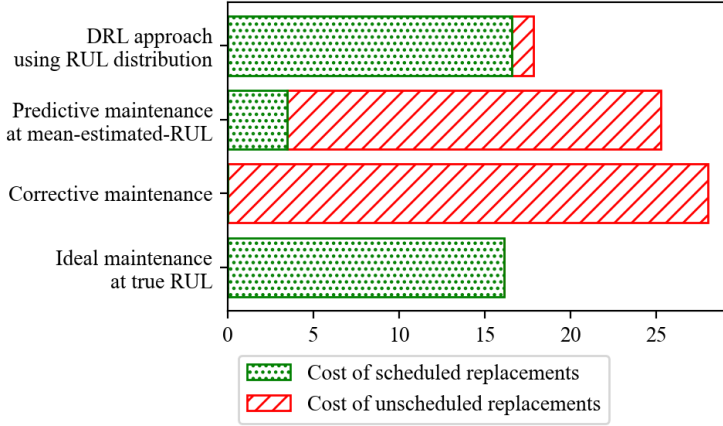


Figure 4.15: Cost of scheduled/unscheduled replacements of the proposed DRL approach and other maintenance strategies.

works with Monte Carlo dropout. These estimates are periodically updated, as more sensor measurements become available. Such estimates of the RUL distribution provide useful information about the uncertainty associated with the RUL prognostics and enables more effective maintenance planning.

With the estimated RUL distribution, we schedule maintenance for turbofan engines using DRL. Maintenance actions are specified adaptively, based on the trends of the RUL prognostics. In contrast to existing studies, we do not use fixed thresholds to trigger maintenance actions. Thus, our DRL approach enables adaptive and flexible, threshold-free maintenance planning.

The results show that our DRL approach with probabilistic RUL prognostics leads to lower maintenance costs and fewer unscheduled maintenance events, when compared to several other maintenance strategies. Compared to maintenance planning at mean-estimated RUL, our DRL approach reduces the total maintenance cost by 29.3%. Moreover, it prevents 95.6% of unscheduled engine replacements. The engines are replaced just before their end-of-life, with an average wasted lives of only 12.8 cycles. Overall, our DRL approach outperforms the several other traditional maintenance strategies in terms of costs and number of unscheduled maintenance events.

Overall, this chapter proposes a generic framework to integrate data-driven, probabilistic RUL prognostics into predictive maintenance. This framework is readily applicable for other aircraft components whose health is continuously monitored.

As future works, we plan to expand the proposed DRL approach for predictive maintenance of multiple components. In addition, we consider more realistic inputs and constraints of aircraft maintenance such as limited space of hangar, logistics of spare parts, and dynamic flight conditions.

ACKNOWLEDGEMENT

This work was in collaboration with MSc student Arthur Reijns and PhD candidate Ingeborg de Pater, supervised by Dr. Mihaela Mitici. I would like to thank them for sharing the initial code and data for the RUL prognostics.

REFERENCES

- [1] V. E. Badea, A. Zamfiroiu, and R. Boncea. “Big Data in the Aerospace Industry”. In: *Informatica Economica* 22.1/2018 (2018), pp. 17–24. ISSN: 14531305. DOI: [10.12948/issn14531305/22.1.2018.02](https://doi.org/10.12948/issn14531305/22.1.2018.02).
- [2] S. Ochella, M. Shafiee, and F. Dinmohammadi. “Artificial intelligence in prognostics and health management of engineering systems”. In: *Engineering Applications of Artificial Intelligence* 108.May 2019 (2022), p. 104552. ISSN: 09521976. DOI: [10.1016/j.engappai.2021.104552](https://doi.org/10.1016/j.engappai.2021.104552).
- [3] O. Fink, Q. Wang, M. Svensén, P. Dersin, W. J. Lee, and M. Ducoffe. “Potential, challenges and future directions for deep learning in prognostics and health management applications”. In: *Engineering Applications of Artificial Intelligence* 92.January (2020), p. 103678. ISSN: 09521976. DOI: [10.1016/j.engappai.2020.103678](https://doi.org/10.1016/j.engappai.2020.103678). arXiv: [2005.02144](https://arxiv.org/abs/2005.02144).
- [4] J. P. Sprong, X. Jiang, and H. Polinder. “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*. Vol. 11. 1. 2019, pp. 1–12. ISBN: 9781936263059. DOI: [10.36001/phmconf.2019.v11i1.776](https://doi.org/10.36001/phmconf.2019.v11i1.776).
- [5] J. Lee and M. Mitici. “An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets”. In: *Reliability Engineering and System Safety* 202 (2020), p. 107052. ISSN: 0951-8320. DOI: [10.1016/j.ress.2020.107052](https://doi.org/10.1016/j.ress.2020.107052).
- [6] M. Mitici and I. de Pater. “Online model-based remaining-useful-life prognostics for aircraft cooling units using time-warping degradation clustering”. In: *Aerospace* 8.6 (2021). ISSN: 22264310. DOI: [10.3390/aerospace8060168](https://doi.org/10.3390/aerospace8060168).
- [7] E. Balaban, A. Saxena, S. Narasimhan, I. Roychoudhury, M. Koopmans, C. Ott, and K. Goebel. “Prognostic health-management system development for electromechanical actuators”. In: *Journal of Aerospace Information Systems* 12.3 (2015), pp. 329–344. ISSN: 23273097. DOI: [10.2514/1.I010171](https://doi.org/10.2514/1.I010171).
- [8] I. de Pater, A. Reijns, and M. Mitici. “Alarm-based predictive maintenance scheduling for aircraft engines with imperfect Remaining Useful Life prognostics”. In: *Reliability Engineering & System Safety* 221 (2022), p. 108341. ISSN: 09518320. DOI: [10.1016/j.ress.2022.108341](https://doi.org/10.1016/j.ress.2022.108341).
- [9] X. Li, Q. Ding, and J. Q. Sun. “Remaining useful life estimation in prognostics using deep convolution neural networks”. In: *Reliability Engineering and System Safety* 172.June 2017 (2018), pp. 1–11. ISSN: 09518320. DOI: [10.1016/j.ress.2017.11.021](https://doi.org/10.1016/j.ress.2017.11.021).

- [10] H. Li, W. Zhao, Y. Zhang, and E. Zio. “Remaining useful life prediction using multi-scale deep convolutional neural network”. In: *Applied Soft Computing Journal* 89 (2020), p. 106113. ISSN: 15684946. DOI: [10.1016/j.asoc.2020.106113](https://doi.org/10.1016/j.asoc.2020.106113).
- [11] G. S. Babu, P. Zhao, and X.-L. Li. “Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life”. In: *International conference on database systems for advanced applications*. Vol. 9642. Cham, 2016, pp. 214–228. ISBN: 978-3-319-32024-3. DOI: [10.1007/978-3-319-32025-0_14](https://doi.org/10.1007/978-3-319-32025-0_14).
- [12] Y. Song, L. Blik, T. Xia, and Y. Zhang. “A Temporal Pyramid Pooling-Based Convolutional Neural Network for Remaining Useful Life Prediction”. In: *Proceedings of the 31st European Safety and Reliability Conference*. 2021, pp. 810–817. ISBN: 978-981-18-2016-8. DOI: [10.3850/978-981-18-2016-8_478-cd](https://doi.org/10.3850/978-981-18-2016-8_478-cd).
- [13] O. Fink. “Data-Driven Intelligent Predictive Maintenance of Industrial Assets”. In: *Women in Industrial and Systems Engineering*. 2020, pp. 589–605. ISBN: 9783030118662. DOI: [10.1007/978-3-030-11866-2_25](https://doi.org/10.1007/978-3-030-11866-2_25).
- [14] P. Do, R. Assaf, P. Scarf, and B. Iung. “Modelling and application of condition-based maintenance for a two-component system with stochastic and economic dependencies”. In: *Reliability Engineering and System Safety* 182.October 2018 (2019), pp. 86–97. ISSN: 09518320. DOI: [10.1016/j.ress.2018.10.007](https://doi.org/10.1016/j.ress.2018.10.007).
- [15] Z. Zhang, X. Si, C. Hu, and Y. Lei. “Degradation data analysis and remaining useful life estimation: A review on Wiener-process-based methods”. In: *European Journal of Operational Research* 271.3 (2018), pp. 775–796. ISSN: 03772217. DOI: [10.1016/j.ejor.2018.02.033](https://doi.org/10.1016/j.ejor.2018.02.033).
- [16] N. C. Caballé, I. T. Castro, C. J. Pérez, and J. M. Lanza-Gutiérrez. “A condition-based maintenance of a dependent degradation-threshold-shock model in a system with multiple degradation processes”. In: *Reliability Engineering and System Safety* 134 (2015), pp. 98–109. DOI: [10.1016/j.ress.2014.09.024](https://doi.org/10.1016/j.ress.2014.09.024).
- [17] Y. Zhao and C. Smidts. “Reinforcement learning for adaptive maintenance policy optimization under imperfect knowledge of the system degradation model and partial observability of system states”. In: *Reliability Engineering and System Safety* (2022), p. 108541. ISSN: 0951-8320. DOI: [10.1016/j.ress.2022.108541](https://doi.org/10.1016/j.ress.2022.108541).
- [18] H. Yang, W. Li, and B. Wang. “Joint optimization of preventive maintenance and production scheduling for multi-state production systems based on reinforcement learning”. In: *Reliability Engineering and System Safety* 214.February (2021), p. 107713. ISSN: 09518320. DOI: [10.1016/j.ress.2021.107713](https://doi.org/10.1016/j.ress.2021.107713).
- [19] J. Lee and M. Mitici. “Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling”. In: *Reliability Engineering and System Safety* 218 (2022), p. 108123. ISSN: 09518320. DOI: [10.1016/j.ress.2021.108123](https://doi.org/10.1016/j.ress.2021.108123).
- [20] S. Kim, J. H. Choi, and N. H. Kim. “Inspection schedule for prognostics with uncertainty management”. In: *Reliability Engineering and System Safety* 222.February (2022), p. 108391. ISSN: 09518320. DOI: [10.1016/j.ress.2022.108391](https://doi.org/10.1016/j.ress.2022.108391).

- [21] Y. Wang, C. Gogu, N. Binaud, C. Bes, R. T. Haftka, and N. H. Kim. “Predictive airframe maintenance strategies using model-based prognostics”. In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 232.6 (2018), pp. 690–709. ISSN: 17480078. DOI: [10.1177/1748006X18757084](https://doi.org/10.1177/1748006X18757084).
- [22] A. Saxena and K. Goebel. *Turbofan Engine Degradation Simulation Data Set*. Moffett Field, CA, 2008.
- [23] A. Saxena, K. Goebel, D. Simon, and N. Eklund. “Damage Propagation Modeling for Aircraft Engine Prognostics”. In: *2008 International Conference on Prognostics and Health Management*. 2008. DOI: [10.1109/PHM.2008.4711414](https://doi.org/10.1109/PHM.2008.4711414).
- [24] Y. Liu, D. K. Frederick, J. A. Decastro, J. S. Litt, and W. W. Chan. *User's Guide for the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS)*. Tech. rep. March. 2012, pp. 1–40.
- [25] L. Peel. “Data Driven Prognostics using a Kalman Filter Ensemble of Neural Network Models”. In: *International Conference on Prognostics and Health Management*. 2008, pp. 1–6. DOI: [10.1109/PHM.2008.4711423](https://doi.org/10.1109/PHM.2008.4711423).
- [26] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao. “Time series classification using multi-channels deep convolutional neural networks”. In: *International Conference on Web-Age Information Management*. 2014, pp. 298–310. ISBN: 9783319080093. DOI: [10.1007/978-3-319-08010-9_33](https://doi.org/10.1007/978-3-319-08010-9_33).
- [27] D. P. Kingma and J. L. Ba. “Adam: A method for stochastic optimization”. In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings* (2015), pp. 1–15. arXiv: [1412.6980](https://arxiv.org/abs/1412.6980).
- [28] N. Srivastava, G. Hinton, A. Sutskever, I. Krizhevsky, and R. Salakhutdinov. “Dropout: A simple Way to Prevent Neural networks from Overfitting”. In: *Journal of Machine Learning Research* 15 (2014). ISSN: 1533-7928.
- [29] C. Serpell, I. Araya, C. Valle, and H. Allende. “Probabilistic Forecasting Using Monte Carlo Dropout Neural Networks”. In: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. 2019, pp. 387–397. ISBN: 9783030339036. DOI: [10.1007/978-3-030-33904-3_36](https://doi.org/10.1007/978-3-030-33904-3_36).
- [30] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor”. In: *35th International Conference on Machine Learning, ICML 2018* 5 (2018), pp. 2976–2989. arXiv: [1801.01290](https://arxiv.org/abs/1801.01290).
- [31] V. Mnih et al. “Human-level control through deep reinforcement learning”. In: *Nature* 518.7540 (2015), pp. 529–533. ISSN: 14764687. DOI: [10.1038/nature14236](https://doi.org/10.1038/nature14236).
- [32] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, and S. Levine. “Soft Actor-Critic Algorithms and Applications”. In: *arXiv* (2018), pp. 1–17. arXiv: [1812.05905](https://arxiv.org/abs/1812.05905).

5

PREDICTIVE AIRCRAFT MAINTENANCE AT FLEET LEVEL USING RUL PROGNOSTICS AND INTEGER LINEAR PROGRAMMING

This chapter proposes a framework to optimize fleet-level predictive aircraft maintenance (PdAM). For PdAM at the fleet-level, the Remaining-Useful-Life (RUL) of multiple components provides deadlines for replacements. The maintenance plan should satisfy operational constraints such as flight schedules and hangar availability. In particular, we are interested in minimizing maintenance costs including the cost of wasted RUL, the penalty cost of overdue replacements, and the setup cost for hangar visits. We formulate the fleet-level PdAM as an integer linear programming (ILP). The solution of the ILP provides an optimal maintenance plan that groups the replacements of components taking into account their RULs. A case study for a fleet of aircraft shows the benefit of predictive maintenance grouping for landing gear brakes.

Parts of this chapter have been published in the following research article:

J. Lee, I. de Pater, S. Boekweit, and M. Mitici, "Remaining-Useful-Life prognostics for opportunistic grouping of maintenance of landing gear brakes for a fleet of aircraft," in *Proceedings of the 7th European Conference of the Prognostics and Health Management Society 2022*, pp.278-285, Turin, Italy, July 6–8, 2022.

This paper has been awarded *Best Paper Award 2nd Prize*, *European Conference of the Prognostics and Health Management Society* in 2022.

5.1. INTRODUCTION

Remaining-useful-life (RUL) prognostics are regarded as a key enabler for predictive aircraft maintenance [1]. Using RUL prognostics, predictive maintenance aims to perform maintenance tasks in anticipation of failures of aircraft components. The expected impact of predictive maintenance is to reduce unexpected failures, increase system availability, and reduce overall maintenance costs [2].

Several studies have proposed algorithms for RUL prognostics for various aircraft systems. For example, Mitici and de Pater develop prognostics for aircraft cooling units using particle filtering [3]. Lee and Mitici propose a regression model to characterize the degradation of aircraft landing gear brakes [4]. Eleftheroglou et al. present the data-driven prognostics for batteries of unmanned aerial vehicles [5]. De Pater, Reijns, and Mitici predict the RUL of aircraft engines using a convolutional neural network and the C-MAPSS data set [6, 7].

Despite the increasing number of RUL prognostics for aircraft systems, few studies integrate these prognostics into actual maintenance planning frameworks to prescribe RUL-driven maintenance tasks [8, 9, 10]. Such integration is particularly complex since aircraft maintenance planning should consider, apart from RUL prognostics, additional factors such as the flight schedule, the limited availability of the hangar where aircraft are maintained, the cost of different maintenance tasks, and the management of spare parts [9].

Moreover, when considering multiple components, it is desirable to group maintenance tasks to reduce maintenance setup costs [11, 12]. The approach of grouping maintenance tasks is referred to as opportunistic maintenance (OM). Several studies have proposed OM for various applications, especially for the maintenance of wind turbines [13, 14, 15]. However, existing studies are not readily applicable for predictive maintenance of a fleet of aircraft because they consider neither RUL prognostics [13], nor the limited availability of critical resources such as hangars [14], nor the fact that the flight schedule of aircraft restricts the planning of maintenance [15]. Thus, these critical constraints need to be considered for the OM for a fleet of aircraft.

In this chapter, we integrate RUL prognostics of aircraft components into opportunistic maintenance (OM) for a fleet of aircraft. Our approach groups maintenance tasks for aircraft components based on their RUL prognostics. The goal of grouping the maintenance of the components is to reduce maintenance setup costs, i.e., the costs needed to initiate maintenance. We illustrate our approach for landing gear brakes of a fleet of aircraft. We first propose a Bayesian linear regression model to predict the RUL of aircraft landing gear brakes. The obtained RUL prognostics are validated against sensor measurements obtained during the actual operation of the aircraft. Then, taking into account these RUL prognostics, we propose an integer linear programming model to opportunistically plan maintenance for the brakes. Our model considers the limited availability of hangars where maintenance can be performed, as well as realistic flight schedules. The result shows that the proposed RUL-driven OM reduces by 20% the expected total maintenance cost for the brakes of a fleet of aircraft compared to traditional maintenance approaches.

5.2. RUL PROGNOSTICS FOR AIRCRAFT LANDING GEAR BRAKES

5.2.1. MAINTENANCE OF AIRCRAFT LANDING GEAR BRAKES

We consider the maintenance of landing gear brakes of wide-body aircraft. A wide-body aircraft is equipped with 8 landing gear brakes, 4 on each side of the wings. The carbon disks of the brakes are worn out when the aircraft decelerates. As soon as the remaining thickness of a braking disk is below an operational threshold, it needs to be replaced before the aircraft can perform another flight.

According to current maintenance practice, aircraft landing gear brakes are inspected periodically [4]. Every d flight cycles, mechanics measure the remaining thickness of the brakes. If the remaining thickness is below a pre-defined threshold, then the brake is replaced with a new one. In order to ensure a high reliability, the inspection interval d is often short, i.e., frequent inspections. Using RUL prognostics, predictive maintenance aims to reduce the wasted life of brakes due to too-early replacements, while limiting the cases when the degradation of a brake may unexpectedly exceed an operational threshold.

5.2.2. CONDITION MONITORING OF AIRCRAFT LANDING GEAR BRAKES

New aircraft are equipped with brake condition monitoring systems that measure the thickness of the brake disks. The thickness of a disk is a direct measure of the degradation level of a brake. Formally, let us denote the degradation level of a brake after ϕ^{th} flight cycle as Z_ϕ . We normalize this degradation level so that $Z_\phi = 0$ when the brake is new. As soon as $Z_\phi > \eta$, where $\eta = 1$ following normalization, the brake needs to be replaced. As soon as $Z_\phi > \eta$, we say that the brake becomes *inoperable*.

In this study, we analyze the actual brake degradation data collected from a fleet of aircraft. These aircraft have been in operation for a period of 6 months up to 3 years. Figure 5.1 shows the normalized degradation data recorded for several aircraft. The x -axis is the number of flight cycles (ϕ) during which a brake was used, and the y -axis is the degradation level (Z_ϕ) of the brakes. The line segments of different colors represent different brakes. Figure 5.1 shows that the degradation of a brake continuously and stochastically increases over time.

Under predictive maintenance, the goal is to use the information provided by RUL prognostics to replace brakes just before their degradation reaches an operational threshold ($\eta = 1$). In Figure 5.1, the end of a line segment is the moment when the brake is replaced under the current practice. We note that in current practice, RUL prognostics are not yet utilized to plan maintenance. Often, brakes are preventively replaced before their degradation level reaches threshold η , wasting the useful life of the brakes. Using RUL prognostics, the aim is to achieve a higher utilization of the brakes while minimizing maintenance costs.

5.2.3. RUL PROGNOSTICS OF AIRCRAFT LANDING GEAR BRAKES

Given the brake degradation data recorded for a fleet of aircraft, we use a Bayesian linear regression (BLR) to predict the remaining-useful-life (RUL) of the brakes. For the brake degradation data in Figure 5.1, its linearity allows the BLR model to achieve accurate RUL

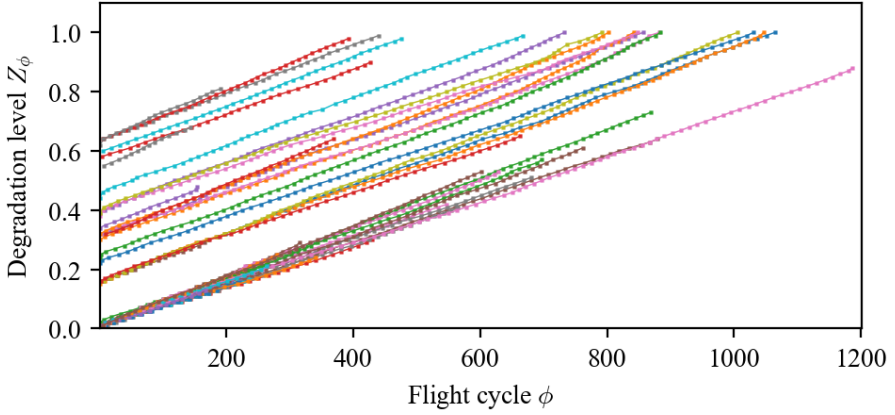


Figure 5.1: The degradation data of landing gear brakes.

5

predictions compared to advanced non-linear models such as artificial neural networks [16]. The input of the BLR model is the number of flight cycles ϕ , and the output is the (predicted) degradation level of a brake after this flight cycle \hat{Z}_ϕ . Formally, we consider the following probabilistic model:

$$P(\hat{Z}_\phi | \phi, \omega, \sigma) = \mathcal{N}(\hat{Z}_\phi | \phi\omega, \sigma^2), \quad (5.1)$$

where ω is the coefficient of the linear model, and σ^2 is the variance of the Gaussian model. The prior of the coefficient ω is assumed to be zero-mean Gaussian, i.e., $P(\omega) = \mathcal{N}(\omega | 0, \lambda \mathbf{I})$. Here, λ and σ^2 are the hyper-parameters of the model, and we consider a Gamma distribution as their prior. Finally, the parameters ω , λ , and σ^2 are jointly optimized by maximizing the log marginal likelihood [17].

Then, given that a brake is already operated for ϕ flight cycles, its RUL is the number of remaining flight cycles until the probability that the degradation level exceeds η is larger than a threshold ζ , i.e.,

$$RUL(\phi) = \min_{\delta} \left\{ \delta : P(\hat{Z}_{\phi+\delta} \geq \eta | Z_\phi) \geq \zeta \right\}, \quad (5.2)$$

where ζ is a given reliability threshold.

The RUL prognostics of the brakes are updated after every flight cycle, taking into account the most recently available degradation data collected from the on-board condition monitoring systems.

A result of RUL prognostics of a brake in the actual data set is shown in Figure 5.2. We predict the RUL of this brake after it has been operated for 748 flight cycles. Given the degradation, the degradation level is expected to exceed $\eta = 1$ after 40 flight cycles with probability $\zeta = 0.5$, and thus, the predicted RUL is $RUL(\phi) = 40$. Given the true RUL $RUL^* = 44$, the error of the RUL prediction is -4 flight cycles.

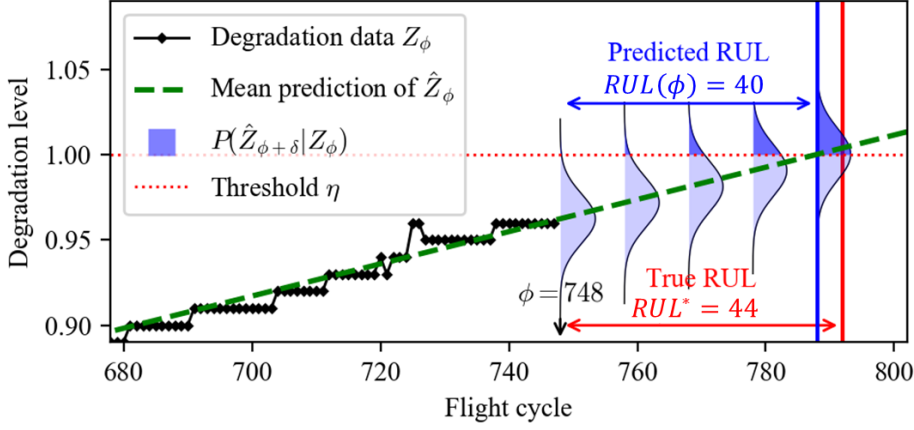


Figure 5.2: Result of RUL prognostics obtained for a brake in the data set. The predicted RUL is 40 cycles and true RUL is 44 cycles.

5

5.2.4. PERFORMANCE OF THE RUL PROGNOSTICS

The performance of the proposed RUL prognostics using BLR is validated based on the actual degradation data collected from a fleet of aircraft. We consider the sensor measurements of 40 brakes of a fleet of aircraft which have been operated in real-life conditions. Each of these 40 brakes have been operated for ϕ^* flight cycles until these brakes become inoperable, i.e., $Z_{\phi^*} = \eta$. Their recorded degradation data are used as a test set for our BLR model since we know the true RUL of the 40 brakes.

We apply BLR at several moments during the operation of the brakes: at 200, 100, 50, and 25 flight cycles before the brakes become inoperable, i.e., the true RUL at these moments in time is $RUL^* \in \{200, 100, 50, 25\}$ flight cycles. We predict the RUL of 40 test brakes at these moments, and plot the box plots of the error ($RUL - RUL^*$) in Figure 5.3. We also determine the mean-bias-error (MBE) and root-mean-squared-error (RMSE) as follows:

$$MBE = \frac{1}{K} \sum_{k=1}^K (\rho_k - \rho_k^*), \quad (5.3)$$

$$RMSE = \frac{1}{K} \sqrt{\sum_{k=1}^K (\rho_k - \rho_k^*)^2}, \quad (5.4)$$

where $K = 40$ brakes considered. Table 5.1 shows the MBE and RMSE of the proposed RUL prognostics.

The error of the RUL prognostics is smaller when true RUL is smaller, i.e., the accuracy of the prognostics increases as we approach the time of failure. In particular, MBE is smaller than 2 flight cycles when the true RUL is 100 flight cycles (see Table 5.1). Considering the fact that an aircraft makes 2 flights per day on average, the bias of the prognostics is roughly 1 day only. Moreover, the RMSE decreases to 5.4 flight cycles, which is

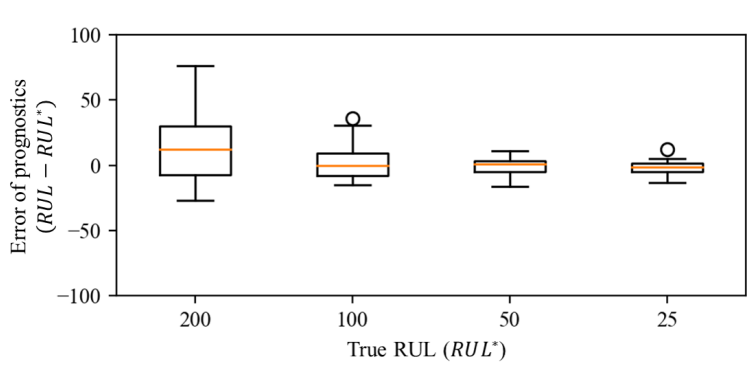


Figure 5.3: Error of the RUL prognostics for the brakes in the data set.

Table 5.1: Performance of the proposed RUL prognostics for the brakes in the data set.

True RUL (RUL^*) [Flight cycles]	200	100	50	25
MBE [Flight cycles]	8.4	1.7	-0.5	-1.7
RMSE [Flight cycles]	41.3	12.4	6.0	5.4

very small considering the average useful life of the brakes in our model (approximately 1250-1450 flight cycles) [2]. Based on this performance of the BLR, we conclude that our prognostics are reliable to be used for maintenance scheduling.

5.3. INTEGRATION OF RUL PROGNOSTICS INTO OPPORTUNISTIC MAINTENANCE SCHEDULING

We propose a RUL-driven opportunistic maintenance planning (RUL-driven OM) for a set of generic aircraft components whose degradation is monitored over time and whose RUL is updated over time. We propose an integer linear programming (ILP) model to group maintenance tasks for these components considering their RUL prognostics.

5.3.1. PROBLEM DESCRIPTION

Our goal is to schedule the maintenance of multiple components of a fleet of aircraft, while minimizing the total maintenance cost. We consider M aircraft ($i \in \mathcal{I} = \{1, \dots, M\}$), each equipped with N components ($j \in \mathcal{J} = \{1, \dots, N\}$). The aircraft perform a sequence of flights according to a flight schedule. Figure 5.4 shows an example of a historical flight schedule. The components are used during flight-time when their degradation evolves stochastically over time. Based on the flight schedule, we define maintenance slots, which are time periods when the aircraft is on-ground at an airport with a hangar. The aircraft can undergo maintenance only at the hangar. Due to the limited space and resources at the hangar, at most H aircraft can be maintained at the same time in the hangar.

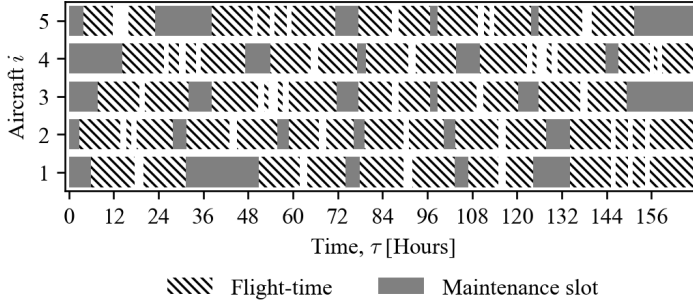


Figure 5.4: An example of flight schedules for 5 aircraft for a week.

The cost of aircraft maintenance consists of i) the setup cost and ii) the component replacement cost. The setup cost C_{set} is the cost to prepare the maintenance of an aircraft in the hangar. This cost can be reduced if multiple maintenance tasks are grouped and performed together during one hangar visit.

Over time, components are scheduled for replacement several weeks in advance. The cost of a scheduled replacement for a component is C_{sch} . If, however, this component becomes inoperable unexpectedly before the moment of the scheduled replacement, we perform an unscheduled replacement for this component at cost C_{uns} . In general, we assume $C_{\text{uns}} > C_{\text{sch}}$ [18].

5.3.2. ROLLING HORIZON FOR RUL-DRIVEN OM

We consider a sequence of time windows that move forward, using a rolling horizon approach (see Figure 5.5). The r^{th} time window is the time period $[T_0^r, T_1^r]$. At the beginning of each time window, we update the RUL prognostics using the most recent degradation data collected until time T_0^r . In addition, we know the maintenance slots available for the fleet of aircraft during this time window, and the availability of the hangar H . Taking into account this information, we optimize the maintenance schedule for the time window $[T_0^r, T_1^r]$ (see Section 5.3.3).

Having obtained a maintenance schedule for time window $[T_0^r, T_1^r]$, we roll forward Δ

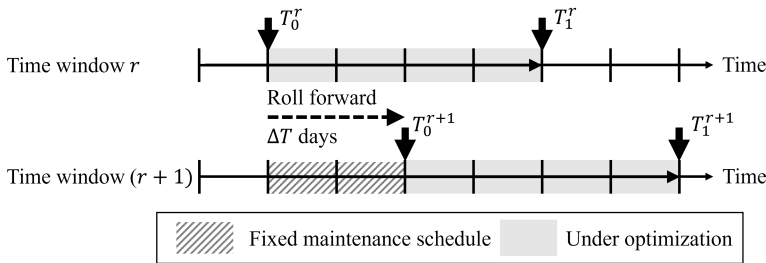


Figure 5.5: Rolling horizon approach.

days. The maintenance schedule for the time period $[T_0^r, T_0^{r+1}]$ is fixed, $T_0^{r+1} = T_0^r + \Delta$. If during $[T_0^r, T_0^{r+1}]$ a component becomes inoperable before its scheduled maintenance, then we perform unscheduled maintenance. We next optimize the maintenance schedule for the new time window $[T_0^{r+1}, T_1^{r+1}]$, updating the RUL prognostics.

5.3.3. INTEGER LINEAR PROGRAMMING OF RUL-DRIVEN OM DECISION VARIABLES

We define the following two decision variables $x_{i,j,t}$ and $y_{i,t}$:

$$x_{i,j,t} = \begin{cases} 1 & \text{if component } j \text{ of aircraft } i \text{ is scheduled} \\ & \text{for maintenance at time slot } t \\ 0 & \text{otherwise} \end{cases} \quad (5.5)$$

$$y_{i,t} = \begin{cases} 1 & \text{if aircraft } i \text{ is scheduled for maintenance} \\ & \text{at time slot } t \text{ but not at time slot } (t-1) \\ 0 & \text{otherwise} \end{cases} \quad (5.6)$$

Here, $x_{i,j,t}$ is a binary variable indicating the maintenance schedule, and $y_{i,t}$ is a binary variable indicating the hangar visit of an aircraft. If an aircraft is scheduled for the maintenance of more than 2 components in consecutive time slots, we regard this as one hangar visit, which requires the setup cost once. Thus, $\sum_{t \in \mathcal{T}_i} y_{i,t}$ is the number of hangar visits of aircraft i .

OBJECTIVE FUNCTION

We consider the following objective function:

$$\min \sum_{i \in \mathcal{I}} \sum_{t \in \mathcal{T}_i} \left(C_{\text{set}} y_{i,t} + \sum_{j \in \mathcal{J}_i} C_{\text{sch}} x_{i,j,t} + \sum_{j \in \mathcal{J}_i} c_{i,j,t} x_{i,j,t} \right), \quad (5.7)$$

where the first term is the setup cost for hangar visits, and the second term is the cost for scheduled replacements.

The third term of the objective in Equation (5.7) penalizes component replacements that are scheduled too early or too late relative to its predicted RUL. Specifically, the penalty $c_{i,j,t}$ is defined as follows:

$$c_{i,j,t} = \begin{cases} c_1 t - c_2 \rho_{i,j,t} & 0 \geq \rho_{i,j,t} \\ c_3 t & 0 < \rho_{i,j,t} \end{cases}. \quad (5.8)$$

Here, $\rho_{i,j,t}$ is the estimated RUL of component j of aircraft i at time slot t . This RUL is estimated using the prognostics model introduced in Section 5.2. Also, we assume that $0 < c_1 < c_2 < c_3$.

An example of a penalty $c_{i,j,t}$ in Equation (5.8) is shown in Figure 5.6. If time slot t is before the moment when component j is expected to become inoperable, i.e., if $\rho_{i,j,t} \geq 0$, then the penalty decreases after each flight cycle. Thus, this penalty incentivizes solutions that schedule replacements when RUL is small, i.e., small wasted useful life. When two time slots t_1 and t_2 have the same RUL ($\rho_{i,j,t_1} = \rho_{i,j,t_2}$), the first term

in Equation (5.8), $c_1 t$, leads to lower penalties for a replacement scheduled at an earlier time slot. On the other hand, if time slot t is after the moment when component j is expected to become inoperable, i.e., if $\rho_{i,j,t} < 0$, then the penalty rapidly increases by $c_3 t$. Thus, with this RUL related penalty $c_{i,j,t}$, our model avoids scheduling a component replacement at a later time than its predicted RUL.

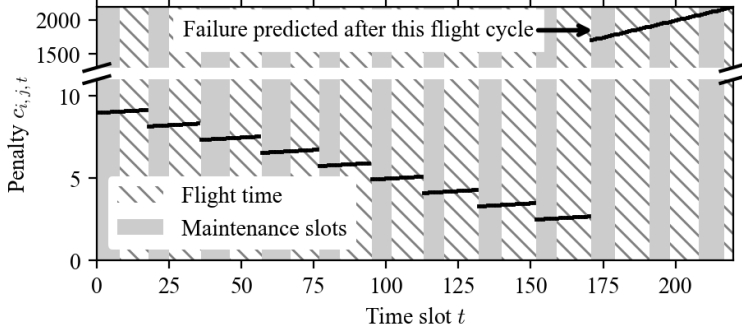


Figure 5.6: An example of penalty parameter $c_{i,j,t}$ in Equation (5.8).

5

CONSTRAINTS

The following constraints are considered:

$$\sum_{t \in \mathcal{T}_i} x_{i,j,t} = 1 \quad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i, \quad (5.9)$$

$$\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}_i} x_{i,j,t} \leq H \quad \forall t \in \mathcal{T}_i, \quad (5.10)$$

$$\sum_{j \in \mathcal{J}_i} x_{i,j,t} \leq 1 \quad \forall i \in \mathcal{I}, \forall t \in \mathcal{T}_i \quad (5.11)$$

$$\sum_{j \in \mathcal{J}_i} x_{i,j,t} = 0 \quad \forall i \in \mathcal{I}, \forall t \in \mathcal{T} : t \notin \mathcal{T}_i, \quad (5.12)$$

$$\sum_{j \in \mathcal{J}_i} x_{i,j,t} - \sum_{j \in \mathcal{J}_i} x_{i,j,(t-1)} \leq y_{i,t} \quad \forall i \in \mathcal{I}. \quad (5.13)$$

Constraint (5.9) ensures all components whose RUL is within the time horizon ($j \in \mathcal{J}_i$) are scheduled for replacements exactly once. Constraint (5.10) ensures that no more than H aircraft are maintained in the hangar at the same time. In addition, constraint (5.11) ensures that only one component of an aircraft can be maintained during a time slot t . This constraint (5.11) is necessary only if $H > 1$. If $H = 1$, then constraint (5.10) is sufficient. Constraint (5.12) prevents scheduling maintenance outside of available maintenance slots ($t \notin \mathcal{T}_i$).

Lastly, constraint (5.13) ensures that the variable $y_{i,t}$ satisfies its definition given in Equation (5.6). In particular, constraint (5.13) provides a lower bound of $y_{i,t}$. So, $y_{i,t} \geq 1$ if the aircraft is brought to the hangar at time slot t , i.e., it is scheduled for maintenance at time slot t , but not at time slot $(t-1)$. On the other hand, $y_{i,t} \geq 0$ if the aircraft is at the

hangar at both time slots t and $(t - 1)$, or if the aircraft is not at the hangar at both time slots t and $(t - 1)$. Since we are minimizing the objective and $C_{\text{set}} > 0$ (see the objective in Equation (5.7)), the optimal value of $y_{i,t}$ is its lower bound.

5.4. RESULTS: RUL-DRIVEN OPPORTUNISTIC MAINTENANCE OF AIRCRAFT LANDING GEAR BRAKES

5.4.1. ILLUSTRATION OF RUL-DRIVEN OM STRATEGY

The proposed RUL-driven OM is applied to the maintenance of aircraft landing gear brakes. A wide-body aircraft has 8 brakes ($N = 8$). We consider a fleet of 10 wide-body aircraft ($M = 10$), and assume that at most 1 aircraft can be maintained in a hangar ($H = 1$) at the same time. Using the rolling horizon approach (see Section 5.3.2), we simulate 10 years of maintenance. The actual degradation of the brakes is shown to follow a Gamma process whose parameters have been estimated in [4, 19].

An example of a maintenance schedule generated by our proposed RUL-driven OM is shown in Figure 5.7. We predict the RUL of components every 2 weeks (the grey vertical lines). The short black vertical lines indicate the moment when the RUL is predicted, the triangles indicate the moment when the component is expected to become inoperable (see Equation (5.2)), and the horizontal line segments indicate the length of RUL. Squares indicate the scheduled time of replacements. The optimal solution always allocates the aircraft to maintenance slots within the predicted RUL, i.e., squares are always on the horizontal line segments. The vertical red lines indicate the grouped maintenance tasks. For example, aircraft 1 replaces 6 components with only 3 hangar visits due to grouping: components 5 and 3, components 6 and 2, and components 8 and 4 are grouped together for maintenance. For aircraft 3, component 2 is replaced strictly at

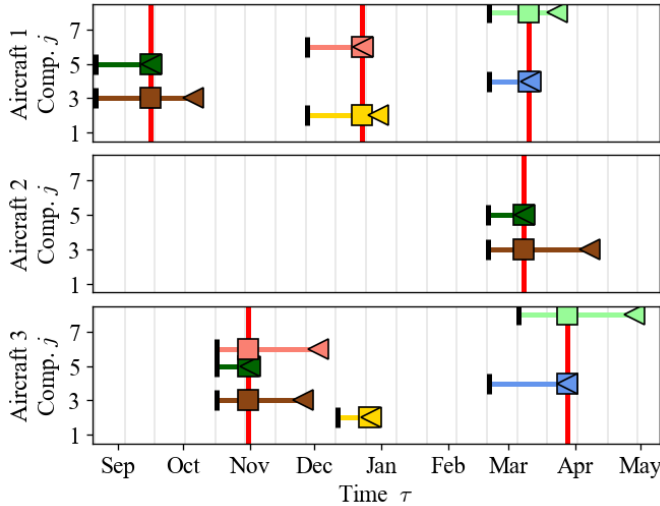


Figure 5.7: An example of optimal maintenance schedule generated by the proposed RUL-driven OM.

RUL without grouping because the closest group of tasks scheduled in November is too early for it, i.e., the benefit of grouping is small.

5.4.2. BENCHMARKS: TRADITIONAL MAINTENANCE STRATEGIES

The performance of the proposed RUL-driven OM is compared with respect to 3 traditional maintenance strategies shown in Table 5.2. Their schedules are optimized using the ILP in Section 5.3.3 with modified objective functions, as follows.

Table 5.2: Comparison of benchmark strategies.

Strategy	Replacement based on	Considering hangar setup cost
PM	Fixed-interval	No
OM	Fixed-interval	Yes
RUL-driven M	RUL-prognostics	No
RUL-driven OM	RUL-prognostics	Yes

5

PREVENTIVE MAINTENANCE (PM)

Under preventive maintenance (PM), the brakes are replaced at fixed time interval, without making use of the updated condition data or RUL prognostics. Thus, the PM schedule is obtained by modifying the penalty parameter $c_{i,j,t}$ in Equation (5.8) as follows:

$$c_{i,j,t} = \begin{cases} c_1 t - c_2 (d_{i,j} - \phi_{i,t}) & \phi_{i,t} \leq d_{i,j} \\ c_3 t & \phi_{i,t} > d_{i,j} \end{cases} \quad (5.14)$$

Here, $d_{i,j}$ is the deadline to replace brake j of aircraft i , and it is assumed to be the mean-cycles-to-failure of the brakes estimated in [4]. Also, we set $C_{\text{set}} = 0$ in the objective function in Equation (5.7) since the setup cost is not considered under PM.

OPPORTUNISTIC MAINTENANCE (OM)

Opportunistic maintenance (OM) also replaces components at fixed time intervals, but it does consider the grouping of maintenance tasks to minimize the setup cost. Thus, for OM, we consider a nonzero C_{set} in the objective function in Equation (5.7), and the penalty parameter $c_{i,j,t}$ defined in Equation (5.14).

RUL-DRIVEN MAINTENANCE (M)

RUL-driven maintenance (M) schedules all component replacements at the predicted RUL, but without grouping these components. The objective function of RUL-driven M has the same penalty parameter $c_{i,j,t}$ defined in Equation (5.8). However, grouping is not performed as setup cost at hangar is not considered, i.e., $C_{\text{set}} = 0$.

5.4.3. RUL-DRIVEN OM VS. BENCHMARK MAINTENANCE STRATEGIES

We perform Monte Carlo simulation to evaluate the expected long-run cost of the maintenance strategies in Table 5.2. The long-run cost is defined as:

$$C = C_{\text{set}} N_{\text{hv}} + C_{\text{sch}} N_{\text{sch}} + C_{\text{uns}} N_{\text{uns}}. \quad (5.15)$$

Here, N_{hv} , N_{sch} , and N_{uns} are the number of hangar visits, the number of scheduled replacements, unscheduled replacements, per year per aircraft, respectively. These values (N_{hv} , N_{sch} , and N_{uns}) are evaluated by Monte Carlo simulations (10^3 runs). Also, C_{set} , C_{sch} , and C_{uns} are the setup cost of a hangar visit, the cost of a scheduled replacement, and the cost of unscheduled replacement, respectively (see Section 5.3.1 for unscheduled replacements). The parameters C_{set} , C_{sch} and C_{uns} depend on the cost model of an aircraft operator. For this case study, we assume $C_{set} = 1$, $C_{sch} = 1$, and $C_{uns} = 2$.

The simulation results in Figure 5.8 and Table 5.3 show the benefit of utilizing RUL prognostics and considering component grouping, i.e., the benefit of the proposed RUL-driven OM. Figure 5.8 shows that the RUL-driven OM results in the lowest expected cost per aircraft per year. The results show that RUL-driven OM leads to 20% lower costs than PM, which is the traditional maintenance strategy.

Table 5.3 shows two reasons why the RUL-driven OM achieves the lowest expected cost. First, it has the smallest number of unscheduled replacements because it optimizes the moment of replacements using RUL prognostics. Second, the RUL-driven OM results in the smallest number of hangar visits, saving the setup cost. Compared to the OM that minimizes the setup cost without considering RUL prognostics, the RUL-driven OM further reduces the number of hangar visits.

In Table 5.3, it is also interesting to see that the total number of scheduled and unscheduled replacements are roughly the same for all strategies, e.g., $N_{sch} + N_{uns} \approx 0.47$. This implies that the best maintenance strategy does not reduce the total number of replacements, but rather optimizes the timing of the replacements so that there is sufficient time to prepare tasks in advance, and reduce the setup cost.

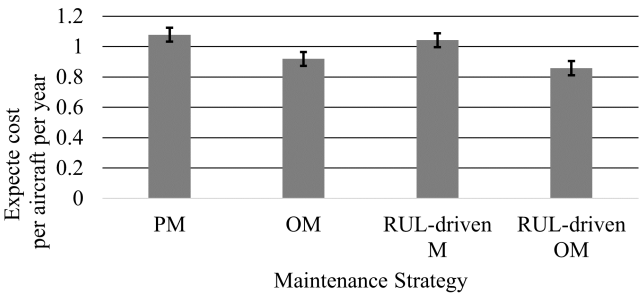


Figure 5.8: Expected cost and its 95% confidence interval.

Table 5.3: Performance of benchmarks.

	PM	OM	RUL-driven M	RUL-driven OM
Maintenance cost C	1.078	0.919	1.042	0.858
Scheduled replacements N_{sch}	0.255	0.327	0.291	0.370
Unscheduled replacements N_{uns}	0.223	0.154	0.186	0.111
Hangar visits N_{hv}	0.377	0.285	0.379	0.266

5.5. CONCLUSION

In this chapter, we integrate Remaining-Useful-Life (RUL) prognostics for aircraft components into opportunistic maintenance planning that groups the maintenance of multiple components. First, the RULs of aircraft landing gear brakes are estimated based on a Bayesian regression model and the actual degradation data collected from a fleet of aircraft. Then, these prognostics are integrated into a maintenance planning optimization - opportunistic maintenance. With this, we group replacements of several brakes to reduce the setup cost for hangar visits. The proposed maintenance planning is applied for a long time horizon using a rolling horizon. Finally, the numerical results show that our proposed RUL-driven opportunistic maintenance planning results in a 20% reduction of total costs compared with several traditional maintenance strategies.

ACKNOWLEDGEMENT

This work was in collaboration with MSc student Stan Boekweit and PhD candidate Ingeborg de Pater, supervised by Dr. Mihaela Mitici. I would like to thank them for sharing the initial code for the schedule optimization.

REFERENCES

- [1] J. P. Sprong, X. Jiang, and H. Polinder. "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*. Vol. 11. 1. 2019, pp. 1–12. ISBN: 9781936263059. DOI: [10.36001/phmconf.2019.v11i1.776](https://doi.org/10.36001/phmconf.2019.v11i1.776).
- [2] J. Lee and M. Mitici. "Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling". In: *Reliability Engineering and System Safety* 218 (2022), p. 108123. ISSN: 09518320. DOI: [10.1016/j.ress.2021.108123](https://doi.org/10.1016/j.ress.2021.108123).
- [3] M. Mitici and I. de Pater. "Online model-based remaining-useful-life prognostics for aircraft cooling units using time-warping degradation clustering". In: *Aerospace* 8.6 (2021). ISSN: 22264310. DOI: [10.3390/aerospace8060168](https://doi.org/10.3390/aerospace8060168).
- [4] J. Lee and M. Mitici. "An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets". In: *Reliability Engineering and System Safety* 202 (2020), p. 107052. ISSN: 0951-8320. DOI: [10.1016/j.ress.2020.107052](https://doi.org/10.1016/j.ress.2020.107052).
- [5] N. Eleftheroglou, S. S. Mansouri, T. Loutas, P. Karvelis, G. Georgoulas, G. Nikolakopoulos, and D. Zarouchas. "Intelligent data-driven prognostic methodologies for the real-time remaining useful life until the end-of-discharge estimation of the Lithium-Polymer batteries of unmanned aerial vehicles with uncertainty quantification". In: *Applied Energy* 254. August (2019), p. 113677. ISSN: 03062619. DOI: [10.1016/j.apenergy.2019.113677](https://doi.org/10.1016/j.apenergy.2019.113677).

- [6] I. de Pater, A. Reijns, and M. Mitici. “Alarm-based predictive maintenance scheduling for aircraft engines with imperfect Remaining Useful Life prognostics”. In: *Reliability Engineering & System Safety* 221 (2022), p. 108341. ISSN: 09518320. DOI: [10.1016/j.ress.2022.108341](https://doi.org/10.1016/j.ress.2022.108341).
- [7] A. Saxena and K. Goebel. *Turbofan Engine Degradation Simulation Data Set*. Moffett Field, CA, 2008.
- [8] B. de Jonge and P. A. Scarf. “A review on maintenance optimization”. In: *European Journal of Operational Research* 285.3 (2020), pp. 805–824. ISSN: 03772217. DOI: [10.1016/j.ejor.2019.09.047](https://doi.org/10.1016/j.ejor.2019.09.047).
- [9] I. de Pater and M. Mitici. “Predictive maintenance for multi-component systems of repairables with Remaining-Useful-Life prognostics and a limited stock of spare components”. In: *Reliability Engineering and System Safety* 214 (2021), p. 107761. ISSN: 0951-8320. DOI: [10.1016/j.ress.2021.107761](https://doi.org/10.1016/j.ress.2021.107761).
- [10] S. Kim, J. H. Choi, and N. H. Kim. “Inspection schedule for prognostics with uncertainty management”. In: *Reliability Engineering and System Safety* 222. February (2022), p. 108391. ISSN: 09518320. DOI: [10.1016/j.ress.2022.108391](https://doi.org/10.1016/j.ress.2022.108391).
- [11] R. E. Wildeman, R. Dekker, and A. C. Smit. “A dynamic policy for grouping maintenance activities”. In: *European Journal of Operational Research* 99.3 (1997), pp. 530–551. ISSN: 03772217. DOI: [10.1016/S0377-2217\(97\)00319-6](https://doi.org/10.1016/S0377-2217(97)00319-6).
- [12] K. Bouvard, S. Artus, C. Bérenguer, and V. Cocquempot. “Condition-based dynamic maintenance operations planning & grouping. Application to commercial heavy vehicles”. In: *Reliability Engineering and System Safety* 96.6 (2011), pp. 601–610. ISSN: 09518320. DOI: [10.1016/j.ress.2010.11.009](https://doi.org/10.1016/j.ress.2010.11.009).
- [13] H. C. Vu, P. Do, M. Fouladirad, and A. Grall. “Dynamic opportunistic maintenance planning for multi-component redundant systems with various types of opportunities”. In: *Reliability Engineering and System Safety* 198. July 2019 (2020), p. 106854. ISSN: 09518320. DOI: [10.1016/j.ress.2020.106854](https://doi.org/10.1016/j.ress.2020.106854).
- [14] J. I. Aizpurua, V. M. Catterson, Y. Papadopoulos, F. Chiacchio, and D. D’Urso. “Supporting group maintenance through prognostics-enhanced dynamic dependability prediction”. In: *Reliability Engineering and System Safety* 168. April (2017), pp. 171–188. ISSN: 09518320. DOI: [10.1016/j.ress.2017.04.005](https://doi.org/10.1016/j.ress.2017.04.005).
- [15] T. Xia, Y. Dong, E. Pan, M. Zheng, H. Wang, and L. Xi. “Fleet-level opportunistic maintenance for large-scale wind farms integrating real-time prognostic updating”. In: *Renewable Energy* 163 (2021), pp. 1444–1454. ISSN: 18790682. DOI: [10.1016/j.renene.2020.08.072](https://doi.org/10.1016/j.renene.2020.08.072).
- [16] A. Oikonomou, N. Eleftheroglou, F. Freeman, T. Loutas, and D. Zarouchas. “Remaining Useful Life Prognosis of Aircraft Brakes”. In: *International Journal of Prognostics and Health Management* 13.1 (2022), pp. 1–11. ISSN: 21532648. DOI: [10.36001/ijphm.2022.v13i1.3072](https://doi.org/10.36001/ijphm.2022.v13i1.3072).
- [17] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python Fabian”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830. ISSN: 1533-7928.

- [18] D. P. Pereira, I. L. Gomes, R. Melicio, and V. M. Mendes. “Planning of aircraft fleet maintenance teams”. In: *Aerospace* 8.5 (2021). ISSN: 22264310. DOI: [10.3390/aerospace8050140](https://doi.org/10.3390/aerospace8050140).
- [19] J. M. van Noortwijk. “A survey of the application of gamma processes in maintenance”. In: *Reliability Engineering and System Safety* 94 (2009), pp. 2–21. ISSN: 09518320. DOI: [10.1016/j.ress.2007.03.019](https://doi.org/10.1016/j.ress.2007.03.019).

6

MULTI-OBJECTIVE OPTIMIZATION OF PREDICTIVE AIRCRAFT MAINTENANCE AT STRATEGY LEVEL USING GAUSSIAN PROCESS LEARNING

For practical implementation of predictive aircraft maintenance (PdAM), we should formulate PdAM strategies that guide maintenance decisions based on RUL prognostics. PdAM strategies are specified by several parameters, such as thresholds of RUL and safety margins. These parameter values should be optimized considering the objectives identified in Chapter 3. Since aircraft maintenance is a complex process, identifying Pareto optimal parameters is a computationally-intensive problem. A Gaussian process (GP) learning model is proposed to efficiently explore the design space of the PdAM parameters. This new algorithm efficiently generates Pareto optimal designs of PdAM strategies. This chapter presents two case studies. The first case study illustrates the benefit of PdAM strategies against traditional time-based maintenance strategies. The second case study optimizes PdAM using probabilistic Remaining-Useful-Life (RUL) prognostics.

Parts of this chapter have been published in the following research articles:

J. Lee and M. Mitici, “Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling,” *Reliability Engineering and System Safety*, vol. 218, p. 108123, 2022.

J. Lee, M. Mitici, S. Geng, and M. Yang, “Designing reliable, data-driven maintenance for aircraft systems with applications to the aircraft landing gear brakes,” in *Proceedings of the 32nd European Safety and Reliability Conference (ESREL)*, pp.25-32, Dublin, Ireland, August 28 – September 1, 2022.

This paper has been awarded *Innovation in Transport Applications, European Safety and Reliability Conference* in 2022, supported by PayPal.

6.1. 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, in [10], a genetic algorithm is used to optimize the design variables of a CBM strategy, but the dominance of this CBM strategy is not demonstrated against other maintenance strategies. Also, in [11, 12, 13], only traditional TBM strategies are explored 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. For example, in [14, 15, 16], benchmark maintenance strategies are limited to only a simple TBM strategy that replaces components at fixed 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, in [15, 16],

costs of component inspections and replacement are minimized. Some studies model reliability as a penalty cost: the cost of stopping system to prevent failure [14], the downtime cost during maintenance [20, 21], the penalty cost of operating failed components [22], the unavailability cost due to a failed systems [23, 24]. Ultimately, these reliability-related costs are integrated in a single cost objective [14, 20, 21, 22, 23, 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 maximizes 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 [34, 32]. As such, further algorithmic improvements are needed to be able to efficiently select designs to be explored for aircraft maintenance.

In this chapter, 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 chapter 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 vs 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. The PdM strategies, however, dominate other maintenance strategies in the *knee region* where conflicting objectives are balanced.
- 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 chapter is organized as follows. In Section 6.2, we formulate the problem of multi-objective aircraft maintenance design. In Section 6.3, we propose a framework to address this problem and an algorithm to adaptively explore the design space of aircraft maintenance. In Section 6.4, we design the maintenance of landing gear brakes, and discuss the benefits of novel predictive maintenance designs. In Section 6.5, we compare the performance of our proposed algorithm for the aircraft maintenance design problem, against several algorithms commonly used for multi-objective design problems. In Section 6.6, we design an advanced predictive maintenance strategy using probabilistic RUL prognostics. Conclusions are provided in section 6.7.

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

$$\mathcal{D} = \{(s, \mathbf{x}^s) | s \in \mathcal{S}, \mathbf{x}^s \in \mathcal{X}^s\}. \quad (6.1)$$

Here, $\mathbf{x}^s \in \mathcal{X}^s$ consists of both continuous and integer variables, while $s \in \mathcal{S}$ 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. The vector of M objectives for a specific maintenance design (s, \mathbf{x}^s) is denoted as follows:

$$\mathbf{f}(s, \mathbf{x}^s) = [f_1(s, \mathbf{x}^s), \dots, f_M(s, \mathbf{x}^s)]. \quad (6.2)$$

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:

$$\begin{aligned} & \min/\max f_m(s, \mathbf{x}^s), m \in \{1, 2, \dots, M\} \\ & \text{subject to } (s, \mathbf{x}^s) \in \mathcal{D} \end{aligned} \quad (6.3)$$

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 $\mathbf{f}(s_1, \mathbf{x}^{s_1}) > \mathbf{f}(s_2, \mathbf{x}^{s_2})$, if and only if

$$\begin{aligned} & (\forall m \in \{1, \dots, M\}, f_m(s_1, \mathbf{x}^{s_1}) \leq f_m(s_2, \mathbf{x}^{s_2})) \\ & \wedge (\exists m \in \{1, \dots, M\}, f_m(s_1, \mathbf{x}^{s_1}) < f_m(s_2, \mathbf{x}^{s_2})). \end{aligned} \quad (6.4)$$

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

$$\nexists (s', \mathbf{x}^{s'}) \in \mathcal{D} \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) > \mathbf{f}(s, \mathbf{x}^s). \quad (6.5)$$

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

$$\mathcal{D}^* = \left\{ (s, \mathbf{x}^s) \in \mathcal{D} \mid \nexists (s', \mathbf{x}^{s'}) \in \mathcal{D} \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) > \mathbf{f}(s, \mathbf{x}^s) \right\}. \quad (6.6)$$

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

$$\mathcal{F}^* = \left\{ \mathbf{f}(s, \mathbf{x}^s) \mid (s, \mathbf{x}^s) \in \mathcal{D}^* \right\}. \quad (6.7)$$

The set of Pareto optimal maintenance designs \mathcal{D}^* and the Pareto front \mathcal{F}^* are the solutions of the multi-objective design space exploration problem in Equation (6.3).

6.3. FRAMEWORK FOR MULTI-OBJECTIVE DESIGN OF AIRCRAFT MAINTENANCE

The design of reliable and cost-efficient aircraft maintenance, given in Equation (6.3) 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 \mathcal{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. Figure 6.1 shows the overview of our framework.

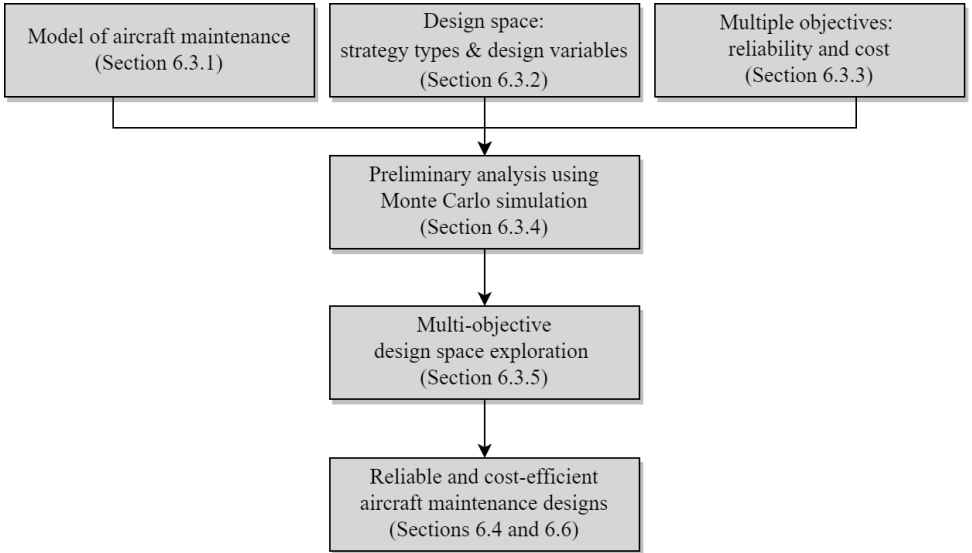


Figure 6.1: Framework of multi-objective aircraft maintenance design.

6.3.1. 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 Figure 6.2, 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.

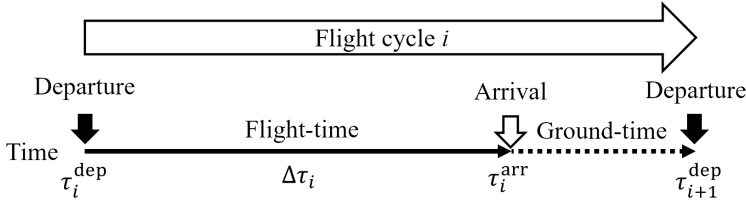


Figure 6.2: Operation of an aircraft.

STOCHASTIC DEGRADATION OF AIRCRAFT COMPONENTS

In this study, we focus on the maintenance of aircraft components whose condition degrades over time due to a 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:

$$Z(\tau_i^{\text{arr}}) - Z(\tau_i^{\text{dep}}) \sim \text{Gamma}(\alpha, \beta), \quad (6.8)$$

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

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.,

$$Z(\tau_{i+1}^{\text{dep}}) - Z(\tau_i^{\text{arr}}) = 0. \quad (6.9)$$

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

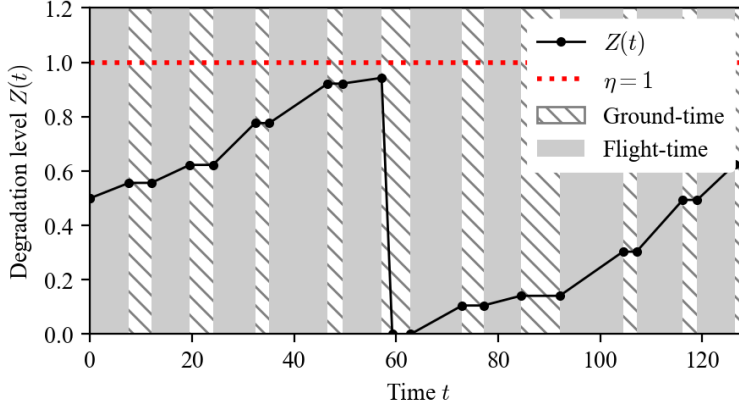


Figure 6.3: A realization of the degradation of a component.

Following Equation (6.8) and (6.9), $Z(t)$ becomes a piece-wise Gamma process. A realization of $Z(t)$ for 10 flight cycles is shown in Figure 6.3. 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. Figure 6.4 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 Equation (6.8). 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 Figure 6.3, 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

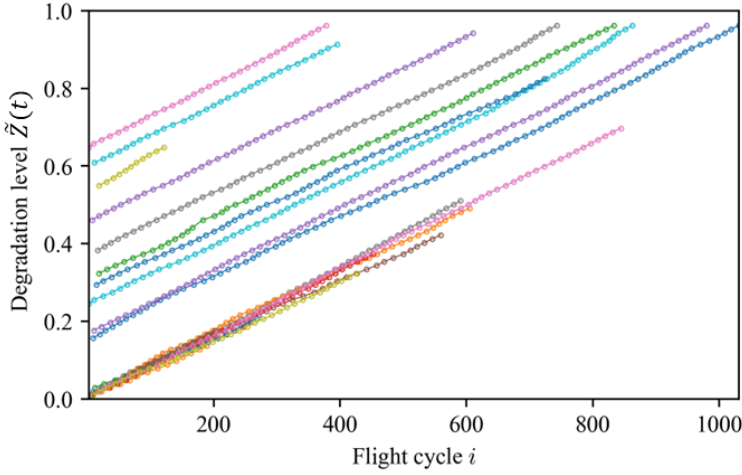


Figure 6.4: Degradation level data of an aircraft brake [17].

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,

$$\hat{Z}(t) = Z(t) + \epsilon_{\text{Ins}}, \quad (6.10)$$

where $\epsilon_{\text{Ins}} \sim \mathcal{N}(0, \sigma_{\text{Ins}}^2)$ 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,

$$\tilde{Z}(t) = Z(t) + \epsilon_{\text{Sen}}, \quad (6.11)$$

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

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.

6.3.2. 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., $\mathcal{D} = \{(s, \mathbf{x}^s) | s \in \mathcal{S}, \mathbf{x}^s \in \mathcal{X}^s\}$. 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]. 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, $\mathcal{S} = \{\text{FIR, FII, VII, SBI, SBR, RBR}\}$. For each strategy type $s \in \mathcal{S}$, the design variables \mathbf{x}^s and their domains are summarized in Table 6.1. 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

Table 6.1: 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
TBM ¹	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]
CBM ²	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]
PdM ³	RUL-based replacement (RBR)	$x_1^{\text{RBR}} = \rho_{\text{Rep}}$	RUL threshold to replace [0, 50]

¹ Time-based maintenance (TBM), ² Condition-based maintenance (CBM), ³ Predictive maintenance (PdM)

life cycle of aircraft landing gear brakes. The expected life cycle of a component following the Gamma process in Equation (6.8) 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 (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 (FII, $[D_{\text{Ins}}, \eta_{\text{Rep}}]$) consists of the FII strategy and specific values for D_{Ins} and η_{Rep} .

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 in-

spection. Then, the next inspection interval $D_{\text{Ins}}(\hat{Z}(t))$ is determined as follows, [24]:

$$D_{\text{Ins}}(\hat{Z}(t)) = 20 + \max\left(a_{\text{Ins}} - \frac{a_{\text{Ins}}}{b_{\text{Ins}}} \hat{Z}(t), 0\right), \quad (6.12)$$

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). Figure 6.5 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.

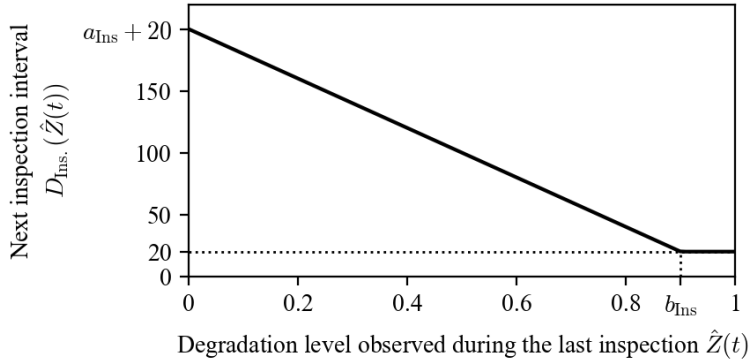


Figure 6.5: The next inspection interval $D_{\text{Ins}}(\hat{Z}(t))$ is defined as a function of the last inspection result $\hat{Z}(t)$ under VII strategy [24], $a_{\text{Ins}} = 180$, $b_{\text{Ins}} = 0.9$.

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 (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 (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 Equation (6.11)), 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 (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]:

$$\tilde{Z}(t + \Delta t) = c_0 + c_1 \Delta t. \quad (6.13)$$

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]:

$$\min\{\Delta t | c_0 + c_1 \Delta t \geq \eta\} \quad (6.14)$$

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 Equation (6.14). Thus, a maintenance design (RBR, $[\rho_{\text{Rep}}]$) consists of the RBR strategy and a specific value for ρ_{Rep} .

6.3.3. MULTIPLE OBJECTIVES OF AIRCRAFT MAINTENANCE

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

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 6.3.1). 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 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 Figure 6.6. 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.

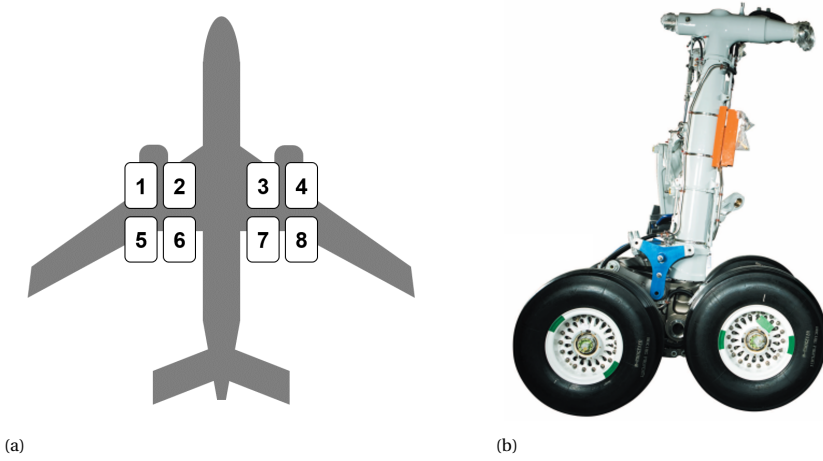


Figure 6.6: An example of multi-component system of aircraft, 8 brakes of wide-body aircraft. (a) Configuration [17]. (b) A brake of landing gear. Image source: <https://www.safran-group.com/products-services/>

MINIMIZATION OF DELAY DUE TO MAINTENANCE (T_D)

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 Figure 6.2 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.

6.3.4. 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. Based on this analysis, we select two objectives for the multi-objective design of aircraft maintenance.

CRUDE MONTE CARLO SIMULATION OF MAINTENANCE DESIGNS

Using the aircraft maintenance model in Section 6.3.1, the strategy types presented in Table 6.1 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 $\sum_{s \in \mathcal{S}} 7^{N^s}$, 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 6.1. 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 $N_{MC} = 1000$ 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_D), of 756 maintenance designs sampled from different strategy types. Table 6.2 shows the Spearman's rank correlation coefficient [44] of these objectives.

Table 6.2: Spearman's rank correlation coefficients between pairs of objectives.

Group 1 Cost-related objectives				Group 2 Reliability-related objectives		
$-MCTR$	N_{Rep}	N_{Ins}		N_{Uns}	N_{Inc}	T_D
$-MCTR$		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_D	-0.907	-0.907	-0.210	0.982	0.877	

Based on the Spearman's coefficients in Table 6.2, 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_D (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.

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 6.2 we select objectives: $MCTR$ and N_{Inc} , i.e.,

$$\begin{aligned} &\text{maximize } f_1 = MCTR \text{ (cost-related objective)} \\ &\text{minimize } f_2 = N_{Inc} \text{ (reliability-related objective)} \end{aligned}$$

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

Figure 6.7 shows the Pareto optimal maintenance designs obtained using crude Monte Carlo simulation of the 756 maintenance designs sampled for the 2 conflicting objectives $f_1 = MCTR$ and $f_2 = N_{Inc}$.

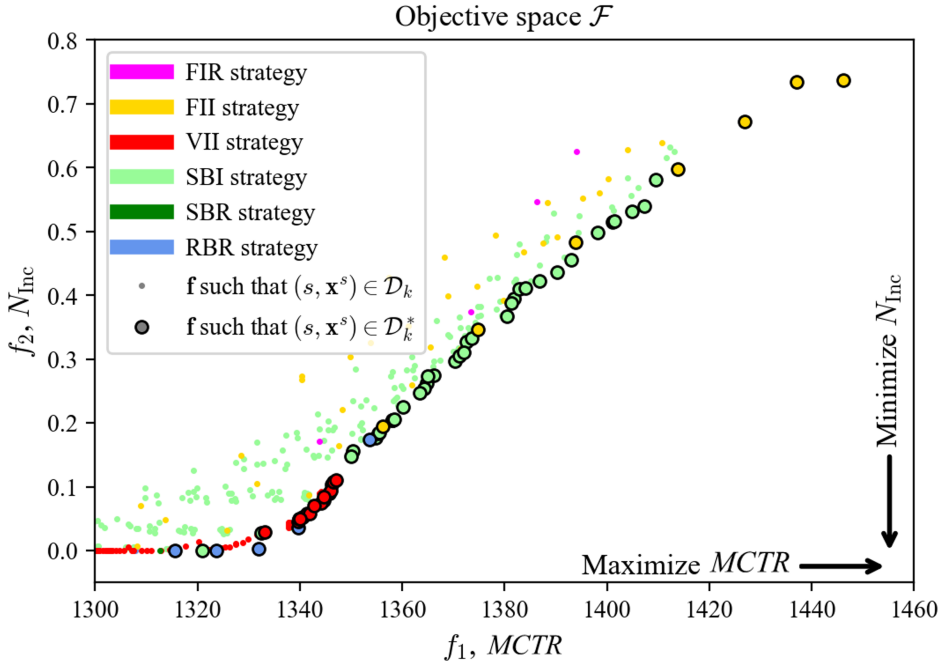


Figure 6.7: Pareto optimal aircraft maintenance designs using crude Monte Carlo simulation.

However, the Pareto front in Figure 6.7 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 6.1. 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.

6.3.5. AN ADAPTIVE ALGORITHM FOR MULTI-OBJECTIVE DESIGN SPACE EXPLORATION OF AIRCRAFT MAINTENANCE

In this section, we propose an algorithm to Explore the design space \mathcal{D} of the multi-objective aircraft maintenance problem using Gaussian process Learning and adaptive **S**Ampling (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.

Figure 6.8 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. 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 .

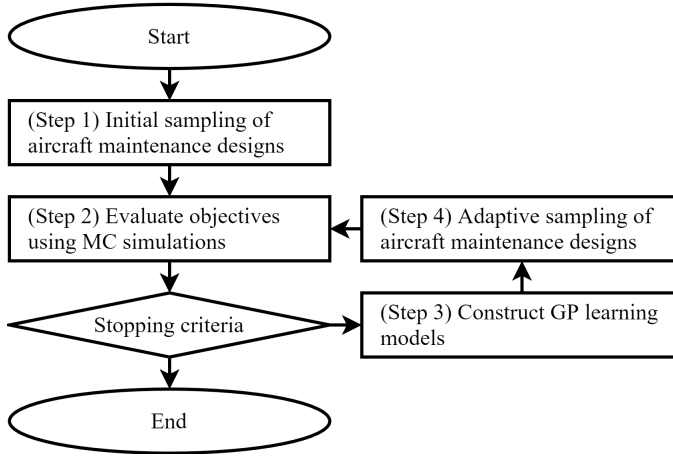


Figure 6.8: Overview of ELSA.

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 6.3.1. 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:

$$\mathcal{D}_k^* = \left\{ (s, \mathbf{x}^s) \in \mathcal{D}_k \mid \nexists (s', \mathbf{x}^{s'}) \in \mathcal{D}_k \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) > \mathbf{f}(s, \mathbf{x}^s) \right\}. \quad (6.15)$$

$$\mathcal{F}_k^* = \{ \mathbf{f}(s, \mathbf{x}^s) \mid (s, \mathbf{x}^s) \in \mathcal{D}_k^* \}. \quad (6.16)$$

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 Equations (6.6) and (6.7). 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:

$$f_m(s, \mathbf{x}^s) \sim \mathcal{GP}_m^s(0, \kappa_m^s(\mathbf{x}^s, \mathbf{x}'^s)), \quad (6.17)$$

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 Equation (6.17), the prior distribution of training data F and a test output f at a test input \mathbf{x} is:

$$\begin{bmatrix} F \\ f \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X, X) & K(X, \mathbf{x}) \\ K(\mathbf{x}, X) & K(\mathbf{x}, \mathbf{x}) \end{bmatrix}\right), \quad (6.18)$$

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

$$f \mid X, F, \mathbf{x} \sim \mathcal{N}\left(\mathbb{E}[f], \mathbb{V}[f]\right). \quad (6.19)$$

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

$$\mathbb{E}[f] = K(\mathbf{x}, X) [K(X, X)]^{-1} F, \quad (6.20)$$

$$\mathbb{V}[f] = K(\mathbf{x}, \mathbf{x}) - K(\mathbf{x}, X) [K(X, X)]^{-1} K(X, \mathbf{x}). \quad (6.21)$$

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:

$$\kappa(\mathbf{x}, \mathbf{x}') = \kappa_{\text{RBF}}(\mathbf{x}, \mathbf{x}') + \kappa_{\text{WN}}(\mathbf{x}, \mathbf{x}'), \quad (6.22)$$

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:

$$\kappa_{\text{RBF}}(\mathbf{x}, \mathbf{x}') = \sigma_{\text{RBF}}^2 \exp \left(-\frac{1}{2} \sum_{j=1}^N \left(\frac{x_j - x'_j}{l_j} \right)^2 \right), \quad (6.23)$$

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:

$$\kappa_{\text{WN}}(\mathbf{x}, \mathbf{x}') = \begin{cases} \sigma_{\text{WN}}^2, & \text{if } \mathbf{x} = \mathbf{x}' \\ 0, & \text{otherwise,} \end{cases} \quad (6.24)$$

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

In Equations (6.23) and (6.24), l_j , σ_{RBF}^2 and σ_{WN}^2 are hyper-parameters, and are optimized using the maximum likelihood estimation [40].

Finally, we train the GP models \mathcal{GP}_m^s for all strategy types $s \in \mathcal{S}$ 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 $s \in \mathcal{S}$ 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 Figure 6.9.

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}_1^s and \mathbf{x}_2^s of strategy type s , and consider a random weight w , such that

$$(s, \mathbf{x}_1^s) \in \mathcal{D}_k^*, (s, \mathbf{x}_2^s) \in \mathcal{D}_k^*, 0 < w < 1. \quad (6.25)$$

Then, the convex combination of \mathbf{x}_1^s and \mathbf{x}_2^s with weight w is sampled as:

$$\mathbf{x}^s = w\mathbf{x}_1^s + (1 - w)\mathbf{x}_2^s. \quad (6.26)$$

Following this approach, we sample n_L vectors of design variables \mathbf{x}^s from \mathcal{X}^s . Figure 6.9 shows an example of global and local sampling in a 2-dimensional domain \mathcal{X}^s .

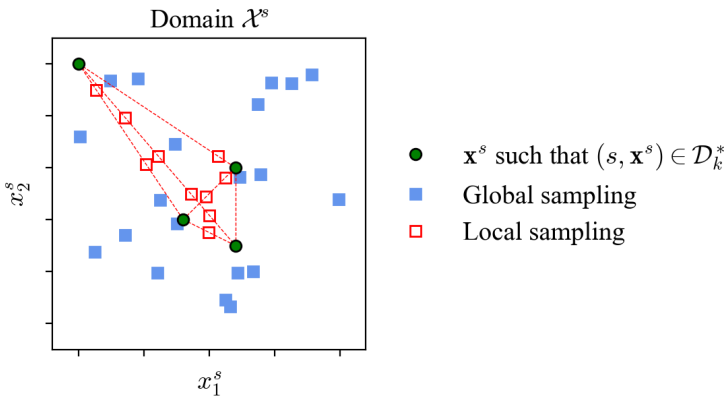


Figure 6.9: An example of global and local sampling in a 2-dimensional domain \mathcal{X}^s of strategy type s .

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 $\hat{f}_m(s, \mathbf{x}^s)$, $1 \leq m \leq M$, is the objective value pre-estimated by the GP model \mathcal{GP}_m^s 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 Equations (6.19)-(6.21)). 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}_m^s , i.e.,

$$\hat{f}_m(s, \mathbf{x}^s) \simeq \mathbb{E}[f_m(s, \mathbf{x}^s)] - v\sqrt{\mathbb{V}[f_m(s, \mathbf{x}^s)]}, \quad (6.27)$$

where $v \geq 0$ 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 \mathcal{D}_{k+1} . Formally, a sampled maintenance design (s, \mathbf{x}^s) is selected and added to \mathcal{D}_{k+1} , when:

$$(s, \mathbf{x}^s) \in \mathcal{D}_{k+1} \iff \left(\exists (s', \mathbf{x}^{s'}) \in \mathcal{D}_k^* \text{ such that } \mathbf{f}(s', \mathbf{x}^{s'}) > \hat{\mathbf{f}}(s, \mathbf{x}^s) \right) \wedge \left(\exists (s', \mathbf{x}^{s'}) \in \mathcal{D}_{k+1} \text{ such that } \hat{\mathbf{f}}(s', \mathbf{x}^{s'}) > \hat{\mathbf{f}}(s, \mathbf{x}^s) \right). \quad (6.28)$$

Figure 6.10 shows an example of the adaptive sampling step when considering the objective space \mathcal{F} . Circle-points represent the Pareto optimal objective vectors obtained

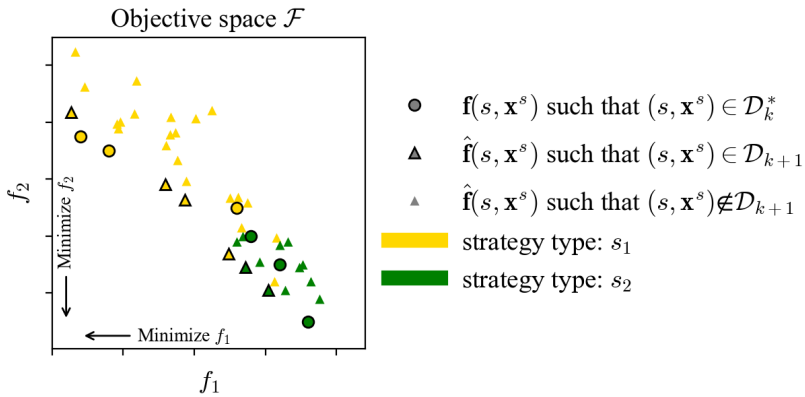


Figure 6.10: An example of an objective space \mathcal{F} during Step 4 of ELSA.

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., $(s, \mathbf{x}^s) \in \mathcal{D}_{k+1}$.

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 \mathcal{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 Equation (6.27)) is influenced mainly by the uncertainty term $\mathbb{V}[f_m]$ and less by 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:

1) 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]. Figure 6.11 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}^*).

2) THE NUMBER OF PARETO OPTIMAL MAINTENANCE DESIGNS.

We count the *number of Pareto optimal maintenance designs* obtained after k iterations, i.e., $|\mathcal{D}_k^*|$. A large $|\mathcal{D}_k^*|$ implies that the generated Pareto front is densely populated. Un-

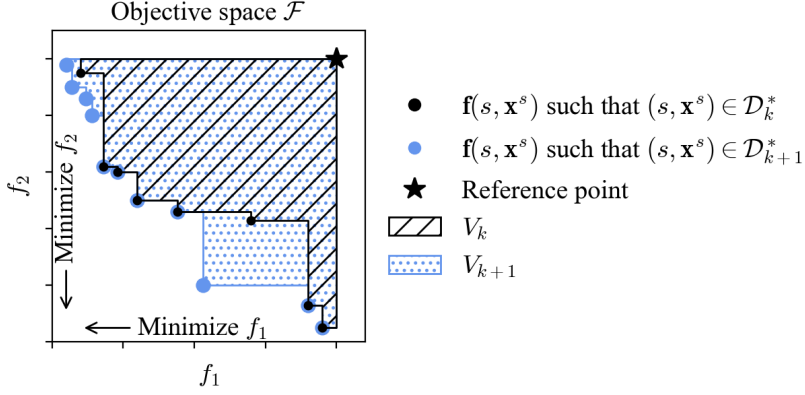


Figure 6.11: A visualization of hyper-volume indicator V at iteration k and iteration $(k+1)$.

like V_k , $|\mathcal{D}_k^*|$ is not monotonically increasing. For example, when a newly identified Pareto optimal maintenance design dominates many solutions of the previous Pareto front, then $|\mathcal{D}_k^*|$ will actually decrease from iteration k to iteration $(k+1)$.

We say that V_k represents the degree of *exploration* of the domains, while $|\mathcal{D}_k^*|$ represents the degree of *exploitation*. This is because V_k is sensitive to a new objective vector far from the current Pareto front, while $|\mathcal{D}_k^*|$ can be increased by a large number of objective vectors close to the current Pareto front. For example, in the upper left corner of Figure 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 Figure 6.11.

6.4. CASE STUDY I: 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.

6.4.1. 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 Figure 6.6). 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 6.3). 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

Table 6.3: Parameter values of Gamma process for the aircraft brake degradation model $\text{Gamma}(\alpha, \beta)$ [17]. L and R indicate whether the brake is on the left or right side of the wing.

Brake position	Side	Shape parameter α	Scale parameter 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

required to be replaced.

As maintenance tasks for landing gear brakes, we consider replacements, visual inspections, and condition monitoring using sensors (see Section 6.3.1). Following interviews with maintenance experts, we assume that the mean time spent for a brake replacement is 3 hours ($\bar{t}_{\text{Rep}} = 3\text{hrs}$). For unscheduled maintenance, we assume an additional time $\bar{t}_{\text{Rep,L}} = 6\text{hrs}$ is needed to supply the required component. A visual inspection requires 2 minutes on average ($\bar{t}_{\text{Ins}} = 2\text{min}$), with $\sigma_{\text{Ins}} = 0.0075$ [17]. For the condition monitoring systems, the sensor error σ_{Sen} is assumed to be 0.0204 [17].

6.4.2. 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 6.3.2). The ranges of their design variables are shown in Table 6.1.

Using ELSA, 195 Pareto optimal designs are identified after simulating 1035 maintenance designs during 19 iterations, i.e., $|\mathcal{D}_{19}^*| = 195$, and $|\mathcal{D}_{19}| = 1035$. The process of iteratively generating Pareto optimal maintenance designs using ELSA is shown in Figure 6.12. Each point in Figure 6.12 represents the objective vector $\mathbf{f}(s, \mathbf{x}^s)$ of one maintenance design $(s, \mathbf{x}^s) \in \mathcal{D}_k$, where s is the strategy type and \mathbf{x}^s are the associated design variables. The large circles represent the Pareto optimal maintenance designs $((s, \mathbf{x}^s) \in \mathcal{D}_k^*)$, 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 obtained after simulating 85 initial maintenance designs, i.e., $|\mathcal{D}_0^*| = 19$, and $|\mathcal{D}_0| = 85$. As an example, in Figure 6.12 the objective vector $\mathbf{f} = [1361.4, 0.2590]$ is annotated as MD0, which corresponds to the maintenance design specified by the strategy $s = \text{FII}$ 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 inci-

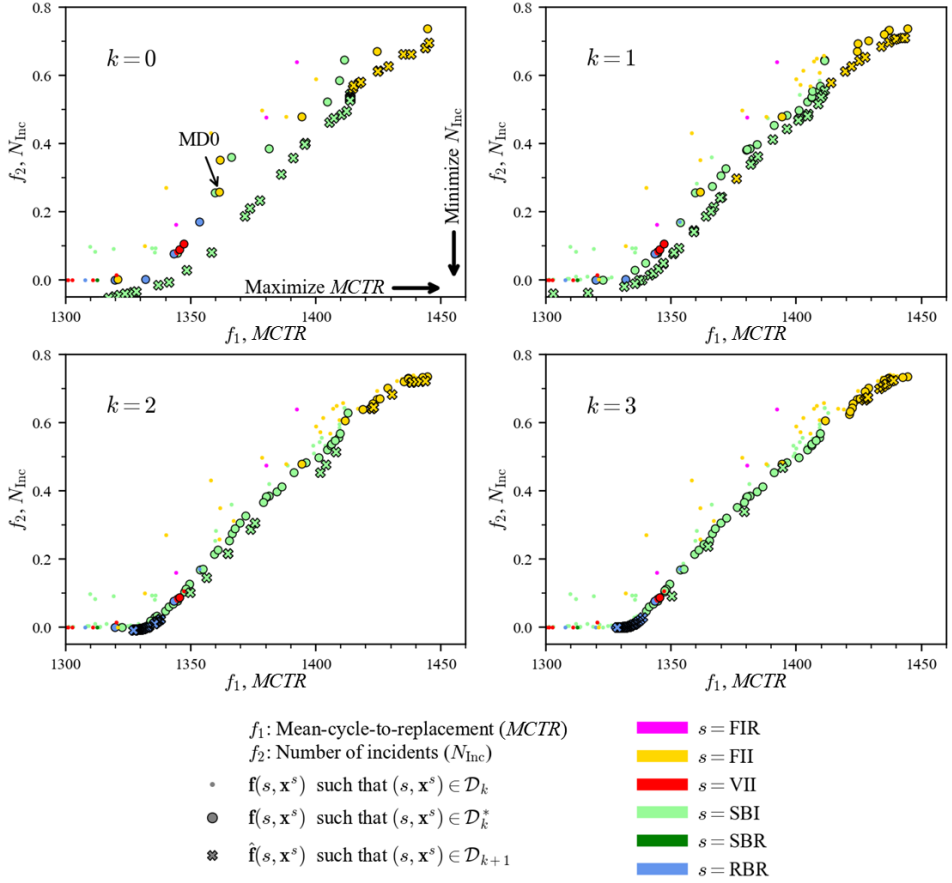


Figure 6.12: Development of Pareto front in the first 4 iterations ($k = 0, 1, 2, 3$) of ELSA.

dents 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 Figure 6.12. Here, these pre-estimated objective vectors include the uncertainty of the GP models (see Equation (6.28)), 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 Figure 6.12. ELSA repeats the same steps of training the GP models, adaptive

sampling, and simulation, for the next iterations $k = 1, 2, 3, \dots, 19$. During the following iterations, the Pareto front is gradually improved, identifying new maintenance designs that dominates 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.

In Figure 6.13, 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 Figure 6.13 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 balances reliability and cost, i.e., having moderate N_{Inc} and $MCTR$. Such a trade-off is often the consideration of aircraft maintenance decision-makers.

6.4.3. 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 Figure 6.13, 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 Figure 6.13 and Table 6.4. 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.

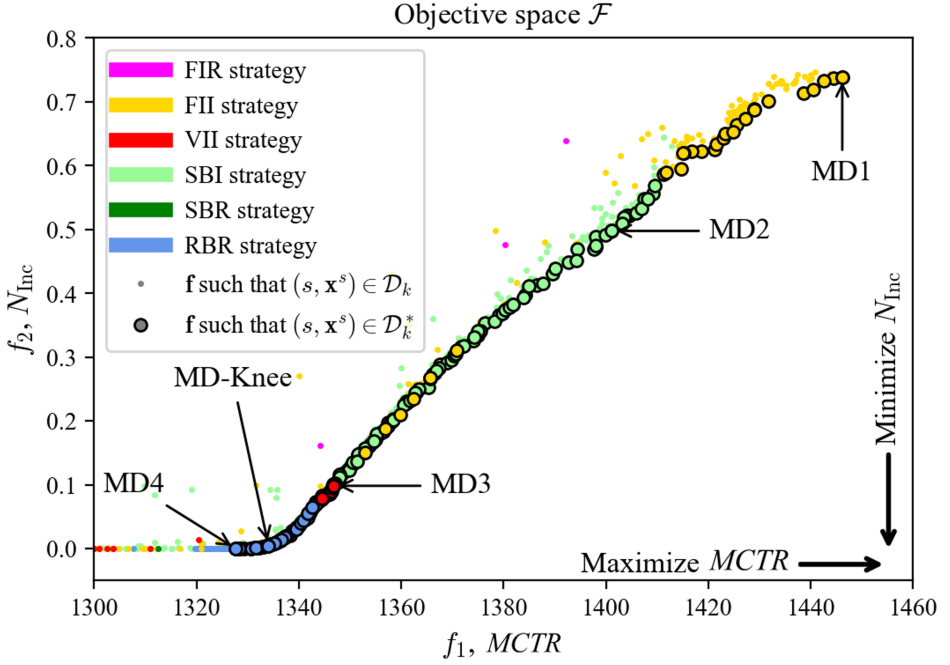


Figure 6.13: Pareto optimal designs of aircraft maintenance strategies obtained by the proposed framework.

Table 6.4: 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 Figure 6.13.

Annotation	Strategy type (s)	Design vector (\mathbf{x}^s)	Objective vector		Selected under the following preference
			$MCTR(f_1)$	$N_{\text{Inc}}(f_2)$	
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}

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 Figure 6.13 and Table 6.4). 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 Figure 6.13 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.

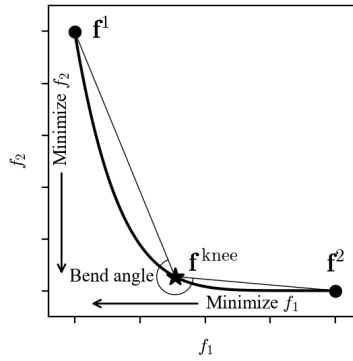


Figure 6.14: Knee point defined by the bend angle.

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 f^1 and f^2 are obtained. Then, for a Pareto optimal point f , the angle between two lines $L^1(f^1, f)$ and $L^2(f^2, f)$ is defined as the bend angle of f . The Pareto optimal point having the maximum bend angle is defined as the *knee point* [50, 49]. Figure 6.14 visualizes this definition of the knee point.

Figure 6.13 shows that the knee point MD-Knee is also obtained using the RBR strategy with $\rho_{Rep} = 22.86$, 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). Figure 6.13 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. Figure 6.15 shows that

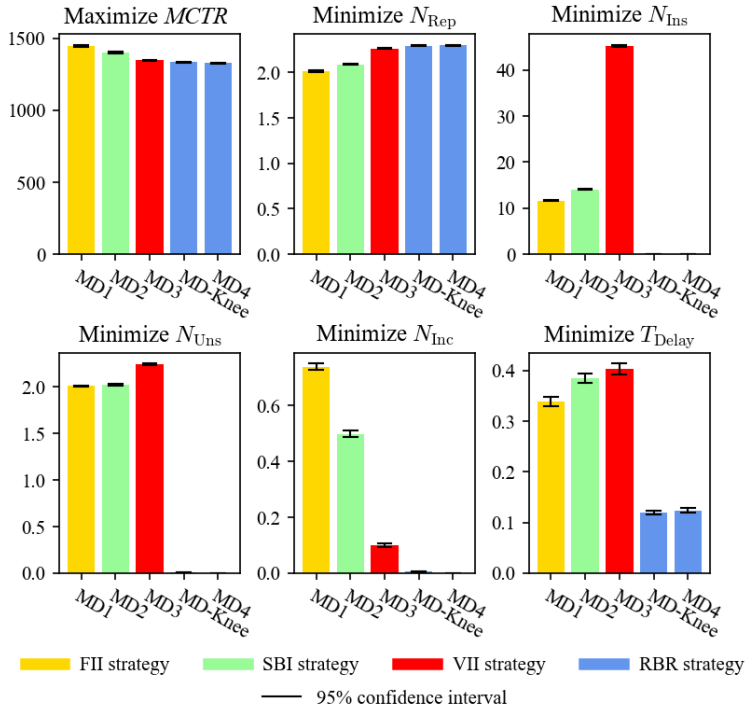


Figure 6.15: Multiple objectives of the Pareto optimal designs shown in Table 6.4.

design MD-Knee outperforms the other Pareto optimal maintenance designs (see Table 6.4) in terms of N_{Uns} , N_{Inc} , and T_{D} . 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.

6.5. QUALITY OF THE PARETO FRONT

In this section, we analyze the quality of the Pareto front generated by ELSA, and compare its performance against other state-of-the-art optimization algorithms.

6.5.1. 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 6.3.5). 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.,

$$\text{RMSE} = \sqrt{\mathbb{E}_{\mathcal{D}_k - \mathcal{D}_{(k-1)}} \left(\hat{f}_m(s, \mathbf{x}^s) - f_m(s, \mathbf{x}^s) \right)^2}, \quad (6.29)$$

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

Figure 6.16 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 Figure 6.13. This shows that the GP models provide reliable pre-estimations in the adaptive sampling step of ELSA. In addition, the RMSE decreases

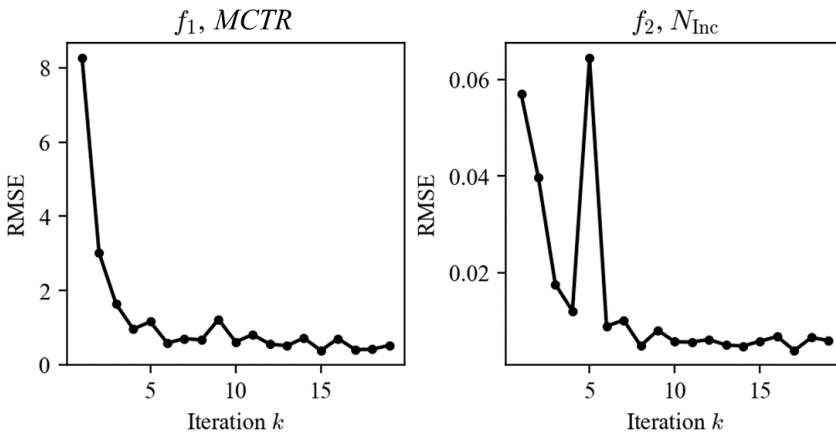


Figure 6.16: RMSE of the pre-estimation made by the GP model at each iteration.

further in later iterations as the GP models are updated with more training data.

6.5.2. THE QUALITY OF THE PARETO FRONT OBTAINED USING ELSA

Figure 6.17 shows the hyper-volume V_k and the number of Pareto optimal designs $|\mathcal{D}_k^*|$ obtained using ELSA. Here, V_k indicates the level of exploration achieved by ELSA, while $|\mathcal{D}_k^*|$ 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 Figure 6.17a). 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 Figure 6.12.

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 Figure 6.17b. ELSA starts with 19 non-dominated maintenance designs, i.e., $|\mathcal{D}_0^*| = 19$. During the following 19 iterations, $|\mathcal{D}_k^*|$ increases gradually to a total of 195 maintenance designs that are Pareto optimal. This continuous and gradual increase of $|\mathcal{D}_k^*|$ 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 Figure 6.17c. 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 $|\mathcal{D}_k^*|$ (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 6.3.5).

6.5.3. PERFORMANCE OF ELSA VS. OTHER ALGORITHMS

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.

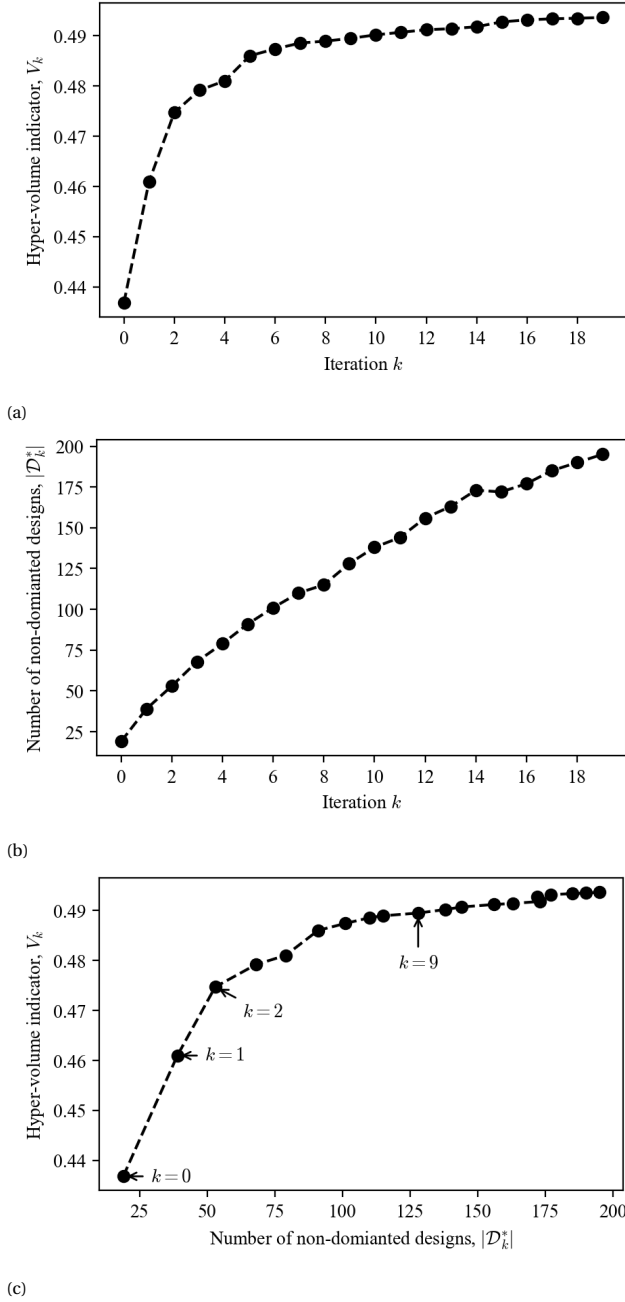


Figure 6.17: (a) Hyper-volume indicator (V_k) at iteration k . (b) Number of Pareto optimal maintenance strategies ($|\mathcal{D}_k^*|$) at iteration k . (c) The balance between exploration (V_k) and exploitation ($|\mathcal{D}_k^*|$).

ReSPIR is an algorithm that iteratively generate 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 Equation (6.3). Figure 6.18 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 $|\mathcal{D}_k^*|$. Here, the number of maintenance designs that are simulated $|\mathcal{D}_k|$ is used as a metric for the computational cost of the algorithms.

Figure 6.18a 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 6.3.5). 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.

Figure 6.18b 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 generates only 57 – 80% of the Pareto optimal maintenance designs generated by ELSA. Compared to the performance difference in V_k (exploration), the difference in $|\mathcal{D}_k^*|$ 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

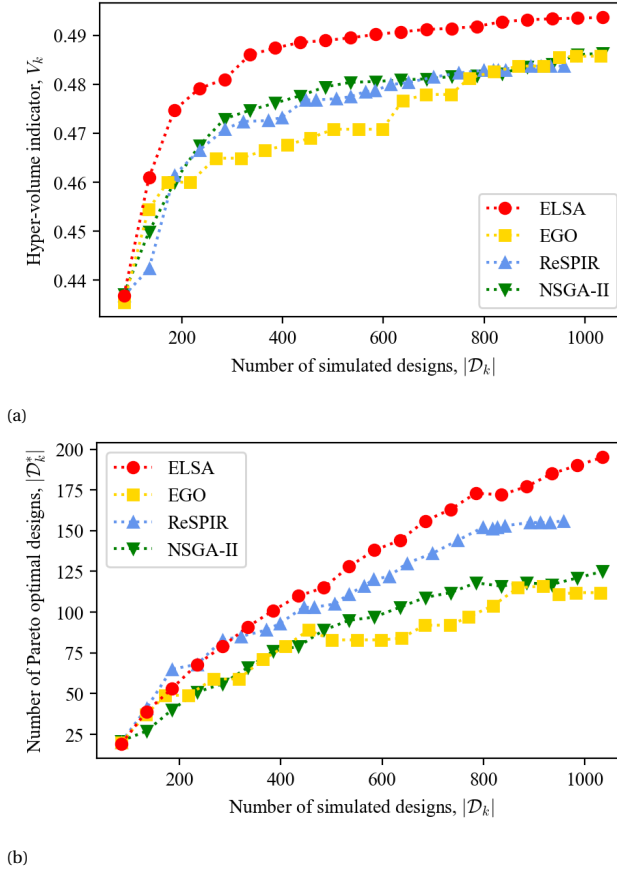


Figure 6.18: The performance of ELSA relative to benchmark algorithms EGO, ReSPIR and NSGA-II.

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.

6.6. CASE STUDY II: PREDICTIVE MAINTENANCE FOR LANDING GEAR BRAKES

In this section, we apply our proposed framework to explore the design space of predictive maintenance for landing gear brakes using probabilistic Remaining-Useful-Life (RUL) prognostics.

6.6.1. PROBABILISTIC RUL PROGNOSTICS FOR LANDING GEAR BRAKES

We apply a Bayesian linear regression (BLR) to estimate the distribution of RUL of the brakes [51]. The input of the BLR model is the degradation data of the brake observed up to current flight cycle i_1 , $\mathcal{Z} = \{(i, \tilde{Z}_i) | i_0 \leq i \leq i_1\}$, where $\tilde{Z}_i = \tilde{Z}(\tau_i^{\text{arr}})$ is the degradation level of the brake obtained from the sensor after flight cycle i . We estimate the degradation level after flight cycle i as:

$$\tilde{Z}_i \sim \mathcal{N}(\omega_0 + \omega_1 i, \sigma_{BLR}^2) \quad (6.30)$$

where ω_0 is the intercept, ω_1 is the coefficient of the linear model, and σ_{BLR}^2 is the variance of the Gaussian model. The prior of the coefficient ω_1 is assumed to be zero-mean Gaussian, i.e., $P(\omega_1) = \mathcal{N}(\omega_1 | 0, \lambda \mathbf{I})$. Here, λ and σ^2 are the hyper-parameters of the model, and we consider a Gamma distribution as their prior. Finally, the parameters ω_1 , λ , and σ^2 are jointly optimized by maximizing the log marginal likelihood. The intercept ω_0 is the mean bias of the model in the input data \mathcal{Z} , i.e., $\omega_0 = \sum_{(i, \tilde{Z}_i) \in \mathcal{Z}} [\tilde{Z}_i - \omega_1 i] / |\mathcal{Z}|$.

Given that a brake has already been used for i flight cycles, its RUL $\rho(i)$ is the number of remaining flight cycles until the probability that the degradation level exceeds η , is larger than a reliability threshold ζ , i.e.,

$$\rho(i) = \min_{\Delta i} \{ \Delta i : P(\tilde{Z}_{i+\Delta i} \geq \eta | \mathcal{Z}) \geq \zeta \}. \quad (6.31)$$

6

The RUL prognostics $\rho(i)$ of the brakes are updated after every flight cycle, taking into account the most recently available degradation data \mathcal{Z} collected from the on-board sensors.

6.6.2. PREDICTIVE MAINTENANCE USING PROBABILISTIC RUL PROGNOSTICS

We propose a predictive maintenance strategy that schedules brake replacements based on probabilistic RUL prognostics proposed in Section 6.6.1. Specifically, a brake replacement is scheduled after $(\rho(i) - \mu)$ flight cycles, where $\rho(i)$ is the RUL prognostics of the brake obtained after the brake has been used for i flight cycles, and μ is a safety margin. So, we schedule a brake replacement μ flight cycles *earlier* than the estimated RUL $\rho(i)$.

Our predictive maintenance strategy has two design parameters to be optimized:

- μ : the safety margin based on which a replacement is scheduled,
- ζ : the reliability threshold used when determining the RUL $\rho(i)$.

These two design parameters are selected from the continuous ranges $\mu \in [\mu_{\min}, \mu_{\max}]$ and $\zeta \in [\zeta_{\min}, \zeta_{\max}]$.

The goal is to efficiently search in the the design space $\mathcal{X} = [\mu_{\min}, \mu_{\max}] \times [\zeta_{\min}, \zeta_{\max}]$ for those values ζ and μ that optimize the cost-related objective ($f_1 = MCTR$) and the reliability-related objective ($f_2 = N_{\text{Inc}}$) (see Section 6.3.4 for the selection of objectives).

6.6.3. MODEL PARAMETERS

We consider 2 aircraft landing gear systems, one on each side of the wing. Each landing gear system has 4 brakes with 3-out-of-4 redundancy. The degradation of the brakes

are assumed to follow a Gamma process with parameters $\alpha = 0.8$ and $\beta = 0.001$. Sensors monitor the degradation of the brakes. The measurement error of the sensors is assumed to be normally distributed with mean zero and $\sigma_s = 0.0204$. [17]

The design space \mathcal{X} is defined by the range of the two design parameters μ (safety margin used when scheduling maintenance) and ζ (reliability threshold used to determine RUL). We explore the following ranges:

$$\begin{aligned}\mu &\in [\mu_{\min}, \mu_{\max}] = [0, 30], \\ \zeta &\in [\zeta_{\min}, \zeta_{\max}] = [0.01, 0.99].\end{aligned}$$

6.6.4. GENERATING PARETO FRONT OF PREDICTIVE MAINTENANCE DESIGNS

Figure 6.19 shows the exploration of the design space during iterations $k = 1$ and 2. Here, the Pareto optimal parameter values $\mathbf{x} \in \mathcal{X}_k^*$ are marked with green squares in the design space (\mathcal{X}), and their objective vectors $\mathbf{f}(\mathbf{x})$ are marked with green circles in the objective space (\mathcal{F}). At iteration $k = 1$, we conducted Monte Carlo simulations of 12 parameter values $\mathbf{x} \in \mathcal{X}_1$, i.e., $|\mathcal{X}_1| = 12$. Among them, seven parameter values were identified as Pareto optimal solutions, i.e., $|\mathcal{X}_1^*| = 7$.

Next, we trained the GP models using the training data obtained by the Monte Carlo

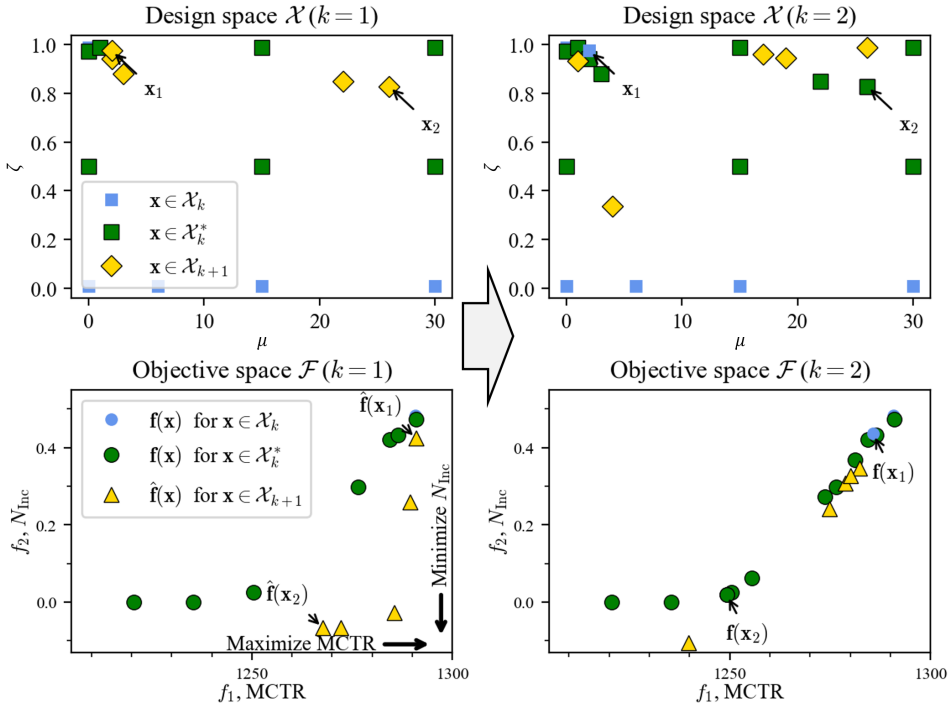


Figure 6.19: Pareto optimal parameter values in objective space \mathcal{F} and design space \mathcal{D} during iterations $k = 1$ and 2.

simulation, i.e., $\{(\mathbf{x}, \mathbf{f}(\mathbf{x})) \mid \mathbf{x} \in \mathcal{X}_1\}$. Based on the adaptive sampling approach, we selected five new design parameters \mathbf{x} to be simulated in the next iteration $k = 2$, i.e., $|\mathcal{X}_2| = 5$. These $\mathbf{x} \in \mathcal{X}_2$ are marked with yellow diamonds in the design space ($k = 1$), and their objective vectors $\hat{\mathbf{f}}(\mathbf{x})$ predicted by the GP models are marked as yellow triangles in the objective space ($k = 1$). For example, see $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}_2$ in the design space ($k = 1$) in Figure 6.19.

At iteration $k = 2$, we conducted Monte Carlo simulations for the selected parameter values $\mathbf{x} \in \mathcal{X}_2$. Given their objective vectors assessed by Monte Carlo simulation, we updated the Pareto optimal solutions \mathcal{X}_2^* . For instance, \mathbf{x}_2 was identified as Pareto optimal solution and was included to \mathcal{X}_2^* . On the other hand, \mathbf{x}_1 was dominated by solutions already obtained at iteration $k = 1$, and thus it was excluded from \mathcal{X}_2^* . We repeated these steps until iteration $k = 19$.

PARETO OPTIMAL PREDICTIVE MAINTENANCE

The final design for the predictive maintenance using probabilistic RUL prognostics is shown in Figure 6.20. Two design parameters, the safety margin μ and the reliability threshold ζ are optimized considering two objectives representing cost and reliability of landing gear brake maintenance. Using the proposed ELSA algorithm, we have simulated 102 parameters during 19 iterations, and identified 55 Pareto optimal parameters. For example, \mathbf{x}_3 in Figure 6.20 corresponds to the parameter values $\mu = 28$ and $\zeta = 0.58$, and it leads to $N_{\text{Inc}} = 0.0003$ and $MCTR = 1228$. This is a dominated solution because if we choose parameter values $\mu = 0$ and $\zeta = 0.26$ (\mathbf{x}_4), then N_{Inc} is reduced to 0.0002 and $MCTR$ is increased to 1233. In other words, we can improve both objectives. Thus, \mathbf{x}_4 is a Pareto optimal solution, while \mathbf{x}_3 is not.

The Pareto front in Figure 6.20 shows a trade-off between reliability ($f_2 : N_{\text{Inc}}$) and cost-efficiency ($f_1 : MCTR$). To increase the reliability of maintenance by reducing N_{Inc} , there must be a reduction in $MCTR$ or a decrease of cost-effectiveness. The choice among the Pareto optimal solutions depends on the preference of decision makers. For instance, if the expected number of incidents is preferred to be below 0.05, then an optimal choice of parameters are $\mu = 23$ and $\zeta = 0.84$ (\mathbf{x}_5 in Figure 6.20). This choice of design parameters lead to $N_{\text{Inc}} = 0.046$ and $MCTR = 1254$. Since \mathbf{x}_5 is a Pareto optimal solution, there is no other solution that leads to a higher $MCTR$ while maintaining $N_{\text{Inc}} \leq 0.5$.

QUALITY OF THE PARETO FRONT

The quality of the Pareto front obtained in Figure 6.20 is evaluated based on two indicators: the hyper-volume indicator V_k of the Pareto front, and the number of Pareto optimal parameters $|\mathcal{X}_k^*|$. V_k shows the level of exploration, while $|\mathcal{X}_k^*|$ shows the level of exploitation (see Section 6.3.5). Figure 6.21 shows these quality indicators at iteration k , where its x-axes are the number of simulated parameters $|\mathcal{X}_k|$ representing the computational cost used until iteration k .

Based on Figure 6.21, we conclude that the identified Pareto front is satisfactory since the improvement of V_k (exploration) was slowed down. On the other hand, the number of Pareto optimal parameters $|\mathcal{X}_k^*|$ (exploitation) still increased significantly until the last iteration. This improvement of $|\mathcal{X}_k^*|$ was achieved by finding the Pareto optimal parameters that lead to very similar objective vectors $\mathbf{f}(\mathbf{x})$. This is clear in the final Pareto front

of Figure 6.20, where many Pareto optimal objective vectors are similar.

The different trends of V_k and $|\mathcal{X}_k^*|$ in Figure 6.21 show that our approach balances exploration and exploitation automatically. Large improvements of V_k during early iterations show that the ELSA algorithm first focused on exploration. For example, 78% of total improvements were made in $k \leq 5$. In the later iterations, however, the ELSA algorithm focused on the exploitation of the design space to identify more Pareto optimal parameters. Such a balancing is achieved by the utilization of both the GP models and our adaptive sampling approach.

6.7. CONCLUSION

In this chapter, 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

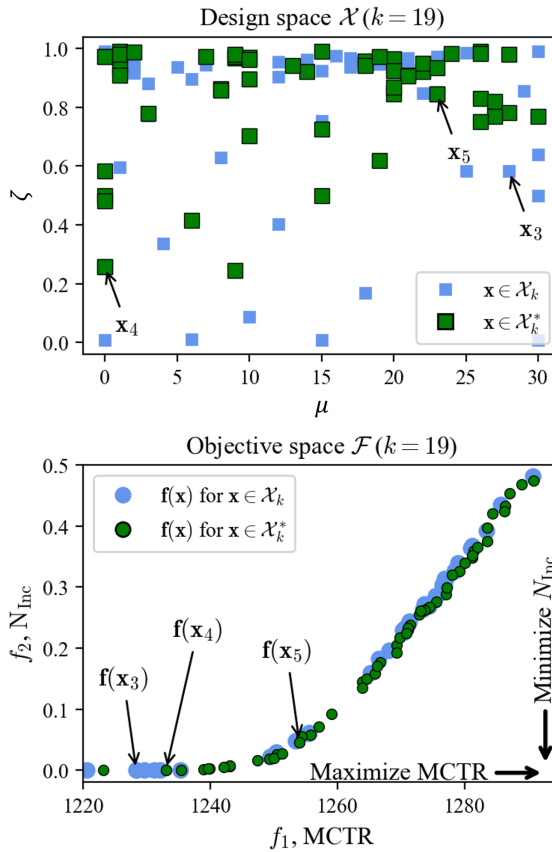


Figure 6.20: Final result for the design of brake maintenance. (Up) The Pareto optimal parameters in the design space. (Down) The Pareto front in the objective space.

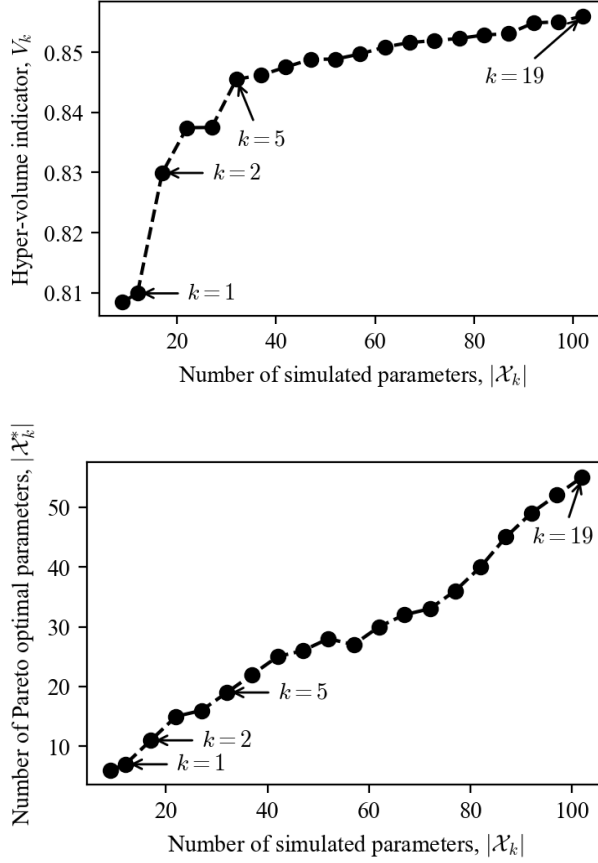


Figure 6.21: Quality indicators at iteration k with respect to the number of simulated parameters.

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. First, we explore 6 different maintenance strategy types which range from traditional time-based maintenance strategies to predictive maintenance strategies. Second, we propose a novel predictive maintenance strategy using probabilistic RUL prognostics, which considers the safety margin and the reliability threshold. With our proposed approach, we identify Pareto optimal predictive maintenance designs for landing gear brakes.

The first case study 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.

The second case study shows that our proposed framework is readily applicable for a novel predictive maintenance strategy using probabilistic RUL prognostics. The obtained Pareto front of predictive maintenance provides multiple Pareto optimal solutions, clearly visualizing the trade-off between the conflicting interests (cost-efficiency and reliability). Based on this Pareto front, the stakeholders of aircraft maintenance can choose a preferable solution considering their interests. Hence, this framework can support multi-objective optimization of generic predictive aircraft maintenance.

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 hypervolume. This is due to the fact that adaptive sampling method of ELSA balances between exploration and exploitation of the design space.

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.

REFERENCES

- [1] IATA's Maintenance Cost Task Force. *Airline Maintenance Cost Executive Commentary (FY2016 data)*. Tech. rep. IATA's Maintenance Cost Task Force, 2017.
- [2] Air Transportation Association. *ATA MSG-3, Operator/Manufacturer Scheduled Maintenance Volume 1 - Fixed Wing Aircraft*. Vol. 1 - Fixed. Air Transport Association of America, 2013.
- [3] H. Wang. "A survey of maintenance policies of deteriorating systems". In: *European Journal of Operational Research* 139.3 (2002), pp. 469–489. DOI: [10.1016/S0377-2217\(01\)00197-7](https://doi.org/10.1016/S0377-2217(01)00197-7).
- [4] S. Alaswad and Y. Xiang. "A review on condition-based maintenance optimization models for stochastically deteriorating system". In: *Reliability Engineering & System Safety* 157 (2017), pp. 54–63. ISSN: 09518320. DOI: [10.1016/j.ress.2016.08.009](https://doi.org/10.1016/j.ress.2016.08.009).
- [5] J. P. Sprong, X. Jiang, and H. Polinder. "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*. Vol. 11. 1. 2019, pp. 1–12. ISBN: 9781936263059. DOI: [10.36001/phmconf.2019.v11i1.776](https://doi.org/10.36001/phmconf.2019.v11i1.776).
- [6] X. S. Si, W. Wang, C. H. Hu, and D. H. Zhou. "Remaining useful life estimation - A review on the statistical data driven approaches". In: *European Journal of Operational Research* 213.1 (2011), pp. 1–14. ISSN: 03772217. DOI: [10.1016/j.ejor.2010.11.018](https://doi.org/10.1016/j.ejor.2010.11.018).

- [7] Z. Zhao, Bin Liang, X. Wang, and W. Lu. "Remaining useful life prediction of aircraft engine based on degradation pattern learning". In: *Reliability Engineering and System Safety* 164.457 (2017), pp. 74–83. ISSN: 09518320. DOI: [10.1016/j.ress.2017.02.007](https://doi.org/10.1016/j.ress.2017.02.007).
- [8] X. Li, Q. Ding, and J. Q. Sun. "Remaining useful life estimation in prognostics using deep convolution neural networks". In: *Reliability Engineering and System Safety* 172.June 2017 (2018), pp. 1–11. ISSN: 09518320. DOI: [10.1016/j.ress.2017.11.021](https://doi.org/10.1016/j.ress.2017.11.021).
- [9] International Maintenance Review Board Policy Board (IMRBPB). *Aircraft Health Monitoring (AHM) integration in MSG-3, IP180*. 2018.
- [10] M. Compare, F. Martini, and E. Zio. "Genetic algorithms for condition-based maintenance optimization under uncertainty". In: *European Journal of Operational Research* 244.2 (2015), pp. 611–623. ISSN: 03772217. DOI: [10.1016/j.ejor.2015.01.057](https://doi.org/10.1016/j.ejor.2015.01.057).
- [11] R. J. Ferreira, A. T. de Almeida, and C. A. Cavalcante. "A multi-criteria decision model to determine inspection intervals of condition monitoring based on delay time analysis". In: *Reliability Engineering and System Safety* 94.5 (2009), pp. 905–912. ISSN: 09518320. DOI: [10.1016/j.ress.2008.10.001](https://doi.org/10.1016/j.ress.2008.10.001).
- [12] C. A. Cavalcante, R. S. Lopes, and P. A. Scarf. "Inspection and replacement policy with a fixed periodic schedule". In: *Reliability Engineering and System Safety* 208.November 2020 (2021), p. 107402. ISSN: 09518320. DOI: [10.1016/j.ress.2020.107402](https://doi.org/10.1016/j.ress.2020.107402).
- [13] M. J. Kallen and J. M. van Noortwijk. "Optimal maintenance decisions under imperfect inspection". In: *Reliability Engineering and System Safety* 90.2-3 (2005), pp. 177–185. ISSN: 09518320. DOI: [10.1016/j.ress.2004.10.004](https://doi.org/10.1016/j.ress.2004.10.004).
- [14] G. Curcuru, G. Galante, and A. Lombardo. "A predictive maintenance policy with imperfect monitoring". In: *Reliability Engineering and System Safety* 95.9 (2010), pp. 989–997. ISSN: 09518320. DOI: [10.1016/j.ress.2010.04.010](https://doi.org/10.1016/j.ress.2010.04.010).
- [15] K. T. Nguyen and K. Medjaher. "A new dynamic predictive maintenance framework using deep learning for failure prognostics". In: *Reliability Engineering and System Safety* 188.September 2018 (2019), pp. 251–262. ISSN: 09518320. DOI: [10.1016/j.ress.2019.03.018](https://doi.org/10.1016/j.ress.2019.03.018).
- [16] B. Lu, Z. Chen, and X. Zhao. "Data-driven dynamic predictive maintenance for a manufacturing system with quality deterioration and online sensors". In: *Reliability Engineering and System Safety* 212.March (2021), p. 107628. ISSN: 09518320. DOI: [10.1016/j.ress.2021.107628](https://doi.org/10.1016/j.ress.2021.107628).
- [17] J. Lee and M. Mitici. "An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets". In: *Reliability Engineering and System Safety* 202 (2020), p. 107052. ISSN: 0951-8320. DOI: [10.1016/j.ress.2020.107052](https://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*. 2017.

- [19] FAA Flight Operations Evaluation Board (FOEB). *Master Minimum Equipment List BOEING 787*. 2015.
- [20] K. T. Huynh, A. Barros, C. Bérenguer, and I. T. Castro. “A periodic inspection and replacement policy for systems subject to competing failure modes due to degradation and traumatic events”. In: *Reliability Engineering and System Safety* 96.4 (2011), pp. 497–508. ISSN: 09518320. DOI: [10.1016/j.ress.2010.12.018](https://doi.org/10.1016/j.ress.2010.12.018).
- [21] K. T. Huynh. “An adaptive predictive maintenance model for repairable deteriorating systems using inverse Gaussian degradation process”. In: *Reliability Engineering and System Safety* 213. April (2021). ISSN: 09518320. DOI: [10.1016/j.ress.2021.107695](https://doi.org/10.1016/j.ress.2021.107695).
- [22] N. Chen, Z. S. Ye, Y. Xiang, and L. Zhang. “Condition-based maintenance using the inverse Gaussian degradation model”. In: *European Journal of Operational Research* 243.1 (2015), pp. 190–199. ISSN: 03772217. DOI: [10.1016/j.ejor.2014.11.029](https://doi.org/10.1016/j.ejor.2014.11.029).
- [23] P. Do, A. Voisin, E. Levrat, and B. Iung. “A proactive condition-based maintenance strategy with both perfect and imperfect maintenance actions”. In: *Reliability Engineering and System Safety* 133 (2015), pp. 22–32. ISSN: 09518320. DOI: [10.1016/j.ress.2014.08.011](https://doi.org/10.1016/j.ress.2014.08.011).
- [24] A. Grall, L. Dieulle, C. Bérenguer, and M. Roussignol. “Continuous-time predictive-maintenance scheduling for a deteriorating system”. In: *IEEE Transactions on Reliability* 51.2 (2002), pp. 141–150. ISSN: 00189529. DOI: [10.1109/TR.2002.1011518](https://doi.org/10.1109/TR.2002.1011518).
- [25] C. S. Syan and G. Ramsoobag. “Maintenance applications of multi-criteria optimization: A review”. In: *Reliability Engineering and System Safety* 190. May (2019), p. 106520. ISSN: 09518320. DOI: [10.1016/j.ress.2019.106520](https://doi.org/10.1016/j.ress.2019.106520).
- [26] E. Kang, E. Jackson, and W. Schulte. “An approach for effective design space exploration”. In: *Monterey Workshop*. Vol. 6662 LNCS. Berlin: Springer, 2010, pp. 33–54. ISBN: 9783642212918. DOI: [10.1007/978-3-642-21292-5_3](https://doi.org/10.1007/978-3-642-21292-5_3).
- [27] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. “A fast and elitist multiobjective genetic algorithm: NSGA-II”. In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. ISSN: 1089778X. DOI: [10.1109/4235.996017](https://doi.org/10.1109/4235.996017).
- [28] M. Palesi and T. Givargis. “Multi-objective design space exploration using genetic algorithms”. In: *Hardware/Software Codesign - Proceedings of the International Workshop* (2002), pp. 67–72. DOI: [10.1145/774801.774804](https://doi.org/10.1145/774801.774804).
- [29] G. Palermo, C. Silvano, and V. Zaccaria. “ReSPIR: A response surface-based pareto iterative refinement for application-specific design space exploration”. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 28.12 (2009), pp. 1816–1829. ISSN: 02780070. DOI: [10.1109/TCAD.2009.2028681](https://doi.org/10.1109/TCAD.2009.2028681).
- [30] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. MIT Press, 2006. ISBN: 9780262182539.
- [31] P. I. Frazier. “A Tutorial on Bayesian Optimization”. In: *arXiv Section 5* (2018), pp. 1–22. arXiv: [1807.02811](https://arxiv.org/abs/1807.02811).

- [32] S. Rojas-Gonzalez and I. Van Nieuwenhuysse. “A survey on kriging-based infill algorithms for multiobjective simulation optimization”. In: *Computers and Operations Research* 116 (2020), p. 104869. ISSN: 03050548. DOI: [10.1016/j.cor.2019.104869](https://doi.org/10.1016/j.cor.2019.104869).
- [33] D. Zhan, Y. Cheng, and J. Liu. “Expected improvement matrix-based infill criteria for expensive multiobjective optimization”. In: *IEEE Transactions on Evolutionary Computation* 21.6 (2017), pp. 956–975. ISSN: 1089778X. DOI: [10.1109/TEVC.2017.2697503](https://doi.org/10.1109/TEVC.2017.2697503).
- [34] I. Couckuyt, D. Deschrijver, and T. Dhaene. “Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization”. In: *Journal of Global Optimization* 60.3 (2014), pp. 575–594. ISSN: 15732916. DOI: [10.1007/s10898-013-0118-2](https://doi.org/10.1007/s10898-013-0118-2).
- [35] D. Simon. *Evolutionary Optimization Algorithms*. Wiley, 2013. ISBN: 978-0470937419.
- [36] J. M. van Noortwijk. “A survey of the application of gamma processes in maintenance”. In: *Reliability Engineering and System Safety* 94 (2009), pp. 2–21. ISSN: 09518320. DOI: [10.1016/j.ress.2007.03.019](https://doi.org/10.1016/j.ress.2007.03.019).
- [37] S. Zhu, W. van Jaarsveld, and R. Dekker. “Spare parts inventory control based on maintenance planning”. In: *Reliability Engineering and System Safety* 193. August 2019 (2020), p. 106600. ISSN: 09518320. DOI: [10.1016/j.ress.2019.106600](https://doi.org/10.1016/j.ress.2019.106600).
- [38] M. Braglia, A. Grassi, and R. Montanari. “Multi-attribute classification method for spare parts inventory management”. In: *Journal of Quality in Maintenance Engineering* 10.1 (2004), pp. 55–65. ISSN: 13552511. DOI: [10.1108/13552510410526875](https://doi.org/10.1108/13552510410526875).
- [39] I. de Pater and M. Mitici. “Predictive maintenance for multi-component systems of repairables with Remaining-Useful-Life prognostics and a limited stock of spare components”. In: *Reliability Engineering and System Safety* 214 (2021), p. 107761. ISSN: 0951-8320. DOI: [10.1016/j.ress.2021.107761](https://doi.org/10.1016/j.ress.2021.107761).
- [40] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python Fabian”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830. ISSN: 1533-7928.
- [41] J. Sheng and D. Prescott. “A coloured Petri net framework for modelling aircraft fleet maintenance”. In: *Reliability Engineering and System Safety*. Vol. 189. Elsevier Ltd, 2019, pp. 67–88. DOI: [10.1016/j.ress.2019.04.004](https://doi.org/10.1016/j.ress.2019.04.004).
- [42] M. Zámková, M. Prokop, and R. Stolin. “Factors influencing flight delays of a European airline”. In: *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis* 65.5 (2017), pp. 1799–1807. ISSN: 12118516. DOI: [10.11118/actaun201765051799](https://doi.org/10.11118/actaun201765051799).
- [43] J. P. Kleijnen. “Design Of Experiments: Overview”. In: *Proceedings of the 2008 Winter Simulation Conference*. IEEE, 2008, pp. 479–488. ISBN: 9781424427086. DOI: [10.1109/WSC.2008.4736103](https://doi.org/10.1109/WSC.2008.4736103).

- [44] A. Almotahari and A. Yazici. “A computationally efficient metric for identification of critical links in large transportation networks”. In: *Reliability Engineering and System Safety* 209 (January (2021)), p. 107458. ISSN: 09518320. DOI: [10.1016/j.ress.2021.107458](https://doi.org/10.1016/j.ress.2021.107458).
- [45] X. He, X. Chen, C. Xiong, Z. Zhu, and L. Zhang. “Optimal time-varying pricing for toll roads under multiple objectives: A simulation-based optimization approach”. In: *Transportation Science* 51.2 (2017), pp. 412–426. ISSN: 15265447. DOI: [10.1287/trsc.2015.0661](https://doi.org/10.1287/trsc.2015.0661).
- [46] J. Chen, B. Xin, Z. Peng, L. Dou, and J. Zhang. “Optimal contraction theorem for exploration-exploitation tradeoff in search and optimization”. In: *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans* 39.3 (2009), pp. 680–691. ISSN: 10834427. DOI: [10.1109/TSMCA.2009.2012436](https://doi.org/10.1109/TSMCA.2009.2012436).
- [47] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca. “Performance assessment of multiobjective optimizers: An analysis and review”. In: *IEEE Transactions on Evolutionary Computation* 7.2 (2003), pp. 117–132. ISSN: 1089778X. DOI: [10.1109/TEVC.2003.810758](https://doi.org/10.1109/TEVC.2003.810758).
- [48] M. Fleischer. “The measure of Pareto optima applications to multi-objective metaheuristics”. In: *International Conference on Evolutionary Multi-Criterion Optimization*. Berlin: Springer, 2003, pp. 519–533. DOI: [10.1007/3-540-36970-8_37](https://doi.org/10.1007/3-540-36970-8_37).
- [49] S. Greco, M. Ehrgott, and J. R. Figueira. *Multiple Criteria Decision Analysis*. Springer New York, 2016. ISBN: 978-1-4939-3093-7. DOI: [10.1007/978-1-4939-3094-4](https://doi.org/10.1007/978-1-4939-3094-4).
- [50] K. Deb and S. Gupta. “Understanding knee points in bicriteria problems and their implications as preferred solution principles”. In: *Engineering Optimization* 43.11 (2011), pp. 1175–1204. ISSN: 0305215X. DOI: [10.1080/0305215X.2010.548863](https://doi.org/10.1080/0305215X.2010.548863).
- [51] A. Oikonomou, N. Eleftheroglou, F. Freeman, T. Loutas, and D. Zarouchas. “Remaining Useful Life Prognosis of Aircraft Brakes”. In: *International Journal of Prognostics and Health Management* 13.1 (2022), pp. 1–11. ISSN: 21532648. DOI: [10.36001/ijphm.2022.v13i1.3072](https://doi.org/10.36001/ijphm.2022.v13i1.3072).

7

EMERGING CHALLENGES OF PREDICTIVE AIRCRAFT MAINTENANCE

Finally, we identify emerging challenges of predictive aircraft maintenance (PdAM) following the adoption of data-driven Remaining-Useful-Life (RUL) prognostics. Experts representing maintenance stakeholders, such as maintenance planners and pilots, discuss the emerging challenges of PdAM in a structured brainstorming session. During this session, the agent-based model of PdAM introduced in Chapter 2 is used to facilitate and guide the brainstorming. The brainstorming results identify three main challenges: 1) the reliability of data-driven technologies (e.g., condition monitoring systems, RUL prognostics algorithms, and decision support systems), 2) the timely and accurate communication between stakeholders, 3) and the stakeholders' trust in these new technologies.

Parts of this chapter are under review for publication:

J. Lee, M. Mitici, H. A. P. Blom, P. Bieber, and F. Freeman, "Identification of emerging hazards for the data-driven predictive aircraft maintenance process," submitted to *Safety Science* in 2021 (under review).

7.1. INTRODUCTION

The increasing use of on-board sensors, aircraft condition monitoring systems (ACMS) and data-driven predictive algorithms are dramatically changing the aircraft maintenance process. Traditionally, the aircraft maintenance process consists of periodic tasks performed by mechanics at pre-determined, fixed time intervals, i.e. time-based maintenance (TBM) [1]. In the last years, however, aircraft maintenance has increasingly made use of data-driven predictive algorithms to increase the level of automatization of the aircraft maintenance process. For example, on-board sensors and ACMS are used to continuously monitor the health condition of aircraft systems. As a result, mechanics need to perform less inspections and checks [2]. Also, data-driven algorithms are developed to detect damages (diagnostics) and predict Remaining-Useful-Life of aircraft systems (prognostics) [3]. Using such predictive algorithms, maintenance tasks are generated only when needed. We refer to this process of using sensor data and predictive algorithms to generate maintenance tasks as data-driven predictive aircraft maintenance (PdAM).

However, the use of data-driven technologies for aircraft maintenance poses novel challenges. On one hand, the retrieval, storage, processing and utilization of sensor data involves risks such as data loss, data corruption, data transmission delays, lack of accuracy of failure prediction algorithms. On the other hand, new experts handling the data and algorithms need to be involved in the traditional aircraft maintenance process. The manner in which these new experts interact with the existing maintenance teams may lead to new challenges. Thus, to safely implement data-driven PdAM, an analysis of emerging challenges is required.

To the best of our knowledge, emerging challenges of data-driven PdAM have not yet been identified and discussed. Existing studies discuss challenges associated with the traditional aircraft maintenance process, TBM. In [4], the authors use an extensive safety questionnaire and show that the behavior of the maintenance personnel is a critical contributing factor to errors in aircraft maintenance. In [5], the authors show that the manner in which the maintenance personnel interact with each other, and their use of hardware/software are the main contributing factors to human errors in aircraft maintenance. However, these studies are not considering the use of data-driven technologies for aircraft maintenance. Since 2018 when the EASA¹ integrated aircraft health monitoring (AHM) into the regulatory basis for aircraft maintenance [6], no studies have discussed emerging challenges of data-driven PdAM, taking into account the entire maintenance process and interactions between maintenance personnel and new data-driven technologies.

The aim of this chapter is to discuss emerging challenges of the data-driven PdAM, based on the identification and analysis of new hazards associated with the new data-driven technologies. Generally, a *hazard* in aviation is defined as follows:

Definition (Hazard)

Any condition, event, or circumstance which could induce an accident [7]; or a condition that could foreseeably cause or contribute to an aircraft accident [8].

In this chapter, we consider hazards related to aircraft maintenance. Especially, we focus

¹EASA: European Union Aviation Safety Agency

on the hazards associated with the adoption of new data-driven technologies, and the hazards related to the interactions between the maintenance personnel involved in new data-driven PdAM.

Traditional hazard identification methods, such as FMEA² or HAZOP³, look at individual process components. For each such component, potential failure modes, their causes and effects are identified [9]. However, these methods fail to capture the interactions between process components, and the hazards associated with these interactions [10, 11]. For the case of aircraft maintenance, the interactions between maintenance personnel, and the manner in which the personnel interacts with the digital systems, are important contributing factors to hazards [5]. Moreover, due to the only recent consideration of data-driven technologies for aircraft maintenance, there is a very limited amount of data and experience of data-driven PdAM.

To address the drawbacks of traditional methods and the lack of data and experience of data-driven PdAM, we apply a structured hazard identification brainstorming [11, 12, 13]. The brainstorming is especially suited to identify emerging hazards associated with novel processes [14, 15, 16], as is the case for data-driven PdAM. We facilitate this brainstorming using an agent-based model of the aircraft maintenance [2], which provides an intuitive understanding of the interactions between agents. The identified hazards are validated in the context of maintenance-related aircraft accidents reported between 2008 and 2013. Finally, in the light of the identified hazards, we discuss emerging challenges for a safe implementation of data-driven PdAM.

The remainder of this chapter is organized as follows. Section 7.2 introduce an agent-based model showing the stakeholders, digital systems, and their interactions in the data-driven PdAM. Section 7.3 identifies and discusses the hazards associated with the data-driven PdAM. Section 7.4 validates the identified hazards in the context of past aircraft accidents related to maintenance. In Section 7.5, we discuss the emerging challenges of data-driven PdAM based on the identified hazards. Finally, we provide conclusions in Section 7.6.

7.2. PROCESS IDENTIFICATION OF DATA-DRIVEN PREDICTIVE AIRCRAFT MAINTENANCE

In this section, we model data-driven predictive aircraft maintenance process (PdAM) using an agent-based model [2]. Here, an *agent* is defined as an independent entity that makes decisions based on a set of rules, interacts with other agents, and has its own goals [17, 18].

The purpose of this process identification is to facilitate brainstorming for hazard identification. The agent-based model of data-driven PdAM is first presented to the experts participating in the brainstorming to provide a solid understanding of this new aircraft maintenance process, and to trigger ideas about emerging hazards.

Table 7.1 and Figure 7.1 show the main agents of data-driven PdAM process and the interactions between them, respectively. In particular, we consider PdAM where a new data management team is introduced to the traditional aircraft maintenance process

²FMEA: Failure Mode and Effects Analysis

³HAZOP: Hazard and Operability Study

[2]. The main agents identified for PdAM are: i) the task generating team (TG), ii) the task planning team (TP), iii) the mechanics team (ME), iv) the flight crews (CR), and v) the data management team (DM). Among them, four agents (TG, TP, ME, and CR) are involved in both the traditional aircraft maintenance process (TBM) and new aircraft maintenance process (PdAM), while DM is a new agent specifically supporting PdAM.

Below we characterize the agents of aircraft maintenance process, by describing their roles and interactions with other agents. In particular, we first elaborate the role and interactions under traditional TBM, and then describe the changes under new PdAM. A detailed model for each agent is given in [2].

TASK GENERATING TEAM (TG)

The role of the task generating team (TG) is to define the type, the due date, and the method used for a maintenance task. TG generates two types of tasks: periodic tasks and one-time tasks. The periodic tasks are generated based on the regulations intro-

Table 7.1: Agents of data-driven predictive aircraft maintenance (PdAM).

Agent Name	Acronym
Task Generating Team	TG
Task Planning Team	TP
Mechanics Team	ME
Flight Crew	CR
Data Management Team	DM

7

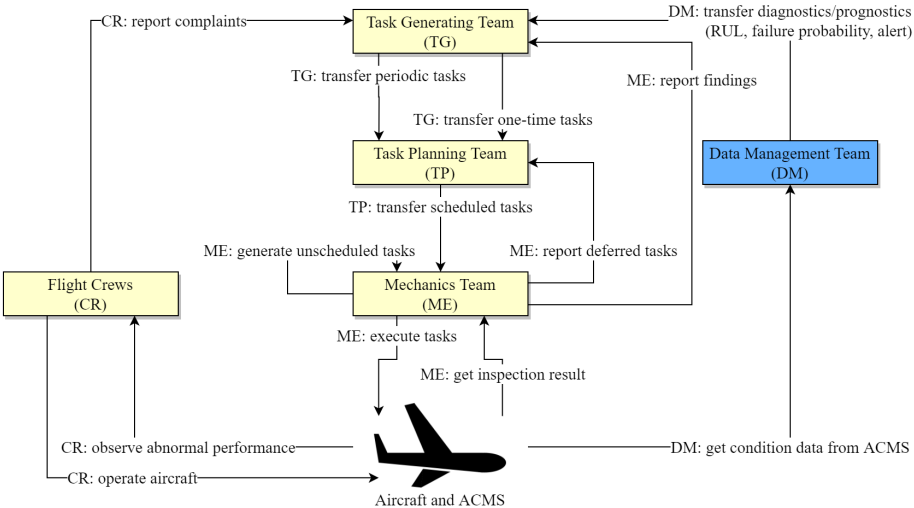


Figure 7.1: Interaction of agents in data-driven predictive aircraft maintenance (PdAM) process. The data management team (DM) is a new agent that supports the transition to data-driven PdAM.

duced by air authorities such as EASA, the manuals provided by aircraft manufacturers, and the analysis of airlines' operation data. TG integrates all this information, and generates periodic tasks (type, due date, and method). Under TBM, these periodic tasks are extensively used as the primary measure to prevent failures. Apart from periodic tasks, one-time maintenance tasks are generated whenever TG receives complaints or findings from flight crews or mechanics. For example, if flight crews observe an abnormal performance of the aircraft during a flight, then they submit a complaint to TG. Similarly, during an inspection, if the mechanics observe an issue, then they submit a finding to TG. Finally, TG analyzes the submitted complaints and findings, and generates necessary tasks to address these issues.

Under PdAM, TG receives additional input such as diagnostics and Remaining-Useful-Life (RUL) prognostics based on DM's data analytics with aircraft condition data. This input is verified and analyzed by TG. When needed, TG asks TP to plan necessary one-time tasks. For example, let the RUL prognostic of a brake indicates that the brake is expected to wear out within 50 flight cycles. If this is shorter than the remaining number of flight cycles before a planned periodic replacement for this brake (traditional TBM), then TG asks TP to reschedule the replacement of the brake earlier (PdAM). In this example, TG anticipates a maintenance issue before it happens, i.e., the maintenance tasks triggered by the prognostics are predictive.

DATA MANAGEMENT TEAM (DM)

The data management team (DM) is a new agent specifically introduced to support data-driven PdAM process. DM is responsible for handling the aircraft condition data and generating diagnostics and RUL prognostics. DM first collects the condition monitoring data from aircraft condition monitoring systems (ACMS), the sensors installed on board of the aircraft. Here, DM may also integrate external databases such as weather data, airport data, and/or data shared by other airlines or maintenance organizations [2]. Data processing and validation is also part of the role of DM. With such data, DM generates diagnostics and RUL prognostics for aircraft systems and structures. In this step, various data-driven algorithms are utilized to generate diagnostics and prognostics, depending on the characteristics of the target system, the inspection/monitoring intervals, the redundancy of the system [19, 20, 21]. Finally, DM transfers the diagnostics and prognostics information to TG.

During the entire process, DM uses a digitalized platform to collect, validate, analyze, and transfer the data and prognostics information. Such platforms to monitor condition data of an aircraft fleet are, for instance, Skywise of Airbus [22], and Airplane Health Management of Boeing [23].

TASK PLANNING TEAM (TP)

The task planning team (TP) schedules in time for the execution of maintenance tasks. The tasks are given by TG (periodic and one-time tasks), as well as by mechanics (deferred tasks) in case additional issues are observed during inspections. TP finds available time slots when the aircraft can undergo maintenance, given the flight schedule of the aircraft, the due dates of each maintenance task, the availability of the mechanics, and the availability of necessary materials and resources. Ultimately, TG generates a schedule for the maintenance tasks. A scheduled task specifies the aircraft, the target

system/structure, the maintenance tasks type, and the mechanics that need to execute the task.

Under PdAM, the role of TP does not change significantly since the tasks generated by TG using diagnostics and prognostics will be given to TP in a similar format as the non-data-driven tasks.

MECHANICS TEAM (ME)

The mechanics team (ME) executes the scheduled tasks received from TP. Various types of maintenance tasks are executed, such as system/structure replacement, restoration, lubrication, inspection [1]. During an inspection, ME may observe additional issues such as an unexpected level of degradation in aircraft structure. Based on the manuals, ME reports such findings. The necessary tasks addressing these findings are executed on-site (unscheduled tasks), or reported to TG for rescheduling in other maintenance slots (deferred tasks).

Similar to TP, the role of ME does not change significantly under PdAM since the task type and the schedules are already specified by TG and TP.

FLIGHT CREW (CR)

The flight crew (CR) includes pilots and cabin crews who actually operate the aircraft. During a flight, CR monitors the condition of the aircraft using on-board ACMS. CR reports a complaint to TG when any abnormality is noticed. The complaints reported by CR are analyzed by TG who may generate additional tasks to address these issues.

Given that the operation of the aircraft is not subject to change under PdAM, the role of CR is not expected to change significantly under PdAM.

7.3. HAZARD IDENTIFICATION FOR DATA-DRIVEN PREDICTIVE AIRCRAFT MAINTENANCE

In this section we identify emerging hazards associated with data-driven predictive aircraft maintenance process (PdAM) by means of a structured brainstorming conducted with the aircraft maintenance agents (see Section 7.2). Thereby, the hazards are identified from divers perspectives of multiple agents. The obtained hazards are analyzed and clustered relative to the agents.

7.3.1. METHODOLOGY

BRAINSTORMING FOR HAZARD IDENTIFICATION

Inspired by [11, 12, 13], a structured hazard identification brainstorm has been performed on February 28th 2019. A total of 10 experts in aircraft maintenance participated at the brainstorming session. Table 7.2 shows the expertise of the participants and their role in the session. Each participant had at least 2 years of experience in the indicated domain. During the brainstorming session, they represented one of the agents identified for data-driven PdAM (see Table 7.1 and Figure 7.1). For the mechanics, their point of view was delegated to the task generating team since their role is not expected to change significantly under PdAM, relative to the changes envisioned for the other agents. We

note that two safety managers represented the overall safety point of view of the aircraft maintenance process. Finally, the session was conducted by a moderator with expertise in aviation safety and experience with the brainstorming methodology for hazard identification. During the session, notes were taken by a secretary.

Table 7.2: Participants at the brainstorming session.

Role in Brainstorming	Expertise & Experience	Number of Attendees
Domain expert	Data management team	2
Domain expert	Task generating team	4
Domain expert	Task planning team	1
Domain expert	Flight crew (Pilot)	1
Domain expert	Safety manager	2
Moderator	Brainstorming method	1
Secretary	N/A	1

At the beginning of the brainstorming session, the agent-based model of the aircraft maintenance process in Figure 7.1 is presented to and discussed with the participating domain experts. This way it is ensured that each participant knows its own agent role as well as the agent roles of the other participants during the brainstorm. It is also verified with each participant if its agent role is correctly presented in the figure. If not then the agent-based figure has to be adapted prior to starting the brainstorm. The validated agent-based model in Figure 7.1 is projected throughout the entire brainstorm. This allows participants to easily express their brainstorm inputs relative to this agent-based model.

The two main rules used for the brainstorming were i) to obtain as many hazards as possible, and ii) criticism and analysis during the session is not allowed. These rules are motivated by cognitive science. The amount of ideas generated are regarded as more important than the quality of the ideas generated during brainstorming [16, 24]. Criticism has been shown to have a negative impact on the open atmosphere necessary for productive brainstorming [16, 24]. In order to avoid discussions about the validity of a hazard, prior to the start of the brainstorm the participants are explained that the brainstorm is about “wide-sense hazards”, i.e., anything that may influence the operation. This means that, in later safety analysis, some of the generated hazards may turn out to pose negligible safety issues and therefore are no true hazards.

During the brainstorming session, the moderator encouraged the participants to share their ideas, opinions, and to interact with each other. The participants are also asked to use the cognitive flow that they are accustomed to in their professional work situations. Once the brainstorm is started each participant easily recognize that in its own professional cognitive flow to have access to the wealth of their operational knowledge and experience.

During the brainstorm session all inputs generated by the participants are written down, and presented to the participants. In case of an error or misunderstanding the contributor of the input can point the need of a correction. To each hazard also the name of the contributor is noted; this allows to contact the contributor in case of follow-

up questions during later safety analysis.

POST-BRAINSTORMING DATA PROCESSING

After the brainstorming session, the raw data are post-processed by independent safety analysts. First, as the generated raw data are “wide-sense hazards”, it is analyzed which are true hazards, i.e., condition, event, or circumstance in aircraft maintenance which could cause or contribute to aircraft incident. Second, the terminology and acronyms used in the formulation of the ideas are unified and, when possible, the terminology used for the agent-based model in Section 7.2 is used. Third, the repetitions of the same idea are analyzed. Ultimately, a list of unique ideas is generated, repetitions being discarded. As a last step, the obtained hazards have been clustered based on whether the hazards are associated only with data-driven PdAM, or with both TBM and PdAM, based on the agent primarily involved with the hazards.

As a result, 41 unique aircraft maintenance hazards are obtained. Out of them, 21 hazards are applicable to generic aircraft maintenance, i.e., these hazards can occur under either TBM or PdAM. The remaining 20 hazards are new hazards associated to the introduction of PdAM, i.e., these hazards can occur only under PdAM. Table 7.3 shows the number of hazards identified from the brainstorming.

7.3.2. ANALYSIS OF BRAINSTORMING RESULTS

In this section we analyze the obtained hazards relative to each aircraft maintenance agent, focusing on the 20 new hazards associated with the introduction of data-driven PdAM.

Table 7.3 shows the number of hazards identified for general aircraft maintenance (both for TBM and PdAM), and for the data-driven predictive aircraft maintenance only (PdAM only). The results show that most number of hazards are identified relative to the task generating team (TG). The main explanation for this result is that TG plays a key role in aircraft maintenance, determining which tasks need to be planned and executed based on the feedback from ME, CR, and DM. The data management team (DM), a new agent supporting PdAM, is associated with 10 new hazards of data-driven PdAM. The mechanics team (ME) is associated with 10 hazards out of the 41 hazards, but only 2 of them are the new hazards of PdAM. This is due to the perception that the role of the mechanics will not change significantly under PdAM because the execution of the

Table 7.3: Number of hazards per involved agent.

	Total	Both TBM & PdAM	Only PdAM
Total number of hazards	41	21	20
Task generating team (TG)	13	5	8
Data management team (DM)	10	0	10
Mechanics team (ME)	10	8	2
Task planning team (TP)	5	5	0
Flight crews (CR)	3	3	0

tasks is expected to be similar to the execution of tasks under current TBM. Similarly, the task planning team (TP) and the flight crew (CR) do not have new hazards under PdAM because they are expected to work in a similar fashion as under current TBM.

Below we discuss and analyze in detail the 20 new hazards of data-driven PdAM, identified for the three agents: DM (Table 7.4), TG (Table 7.5), and ME (Table 7.6).

HAZARDS ASSOCIATED WITH THE DATA MANAGEMENT TEAM (DM)

The 10 hazards associated with DM are related to i) the performance of the aircraft condition monitoring systems (ACMS), ii) the performance of the data-driven algorithms used to generate diagnostics and RUL prognostics for aircraft systems and structures, iii) communication issues between agents, and iv) delay in the knowledge and data transfer between agents. Their descriptions and IDs are given in Table 7.4.

Four hazards are identified relative to the performance of the ACMS (see hazard H_{01} , H_{02} , H_{03} , and H_{04}). First, the ACMS itself can be subject to malfunction or become inoperable (see hazard H_{01}). In this case, the streams of condition data are no longer available, and thus DM cannot generate any diagnostics or prognostics. A worse case is when DM does not notice the malfunction of the ACMS. In this case, the malfunction results in the ACMS collecting corrupted data, which is used for diagnostics and prognostics. This is the subject of hazards H_{02} , and H_{03} . Hazard H_{02} refers to the case when incorrect or inaccurate condition data is used by DM. In turn, the resulting diagnostics and prognostics become unreliable. If these unreliable diagnostics and prognostics are transferred to TG to generate maintenance tasks, then the impact of this hazard is propagated to the entire aircraft maintenance process. Even when the ACMS collects accurate condition data, this data can still become corrupted during data transfer from ACMS to DM (see hazard H_{03}). This hazard may trigger additional hazards following the same propagation path as for hazard H_{02} . Another important aspect is to obtain the condition monitoring data

Table 7.4: Hazards of data-driven PdAM, associated with the Data Management team (DM).

ID	Description
H_{01}	DM could not get data because aircraft condition monitoring system is not functioning, or inoperative.
H_{02}	DM gets incorrect/inaccurate data because aircraft condition monitoring system is malfunctioning.
H_{03}	DM gets incorrect/inaccurate data that is corrupted during data transfer.
H_{04}	DM gets data too late because of delays in data transfer from aircraft condition monitoring systems.
H_{05}	DM generates wrong prognostics/diagnostics.
H_{06}	DM uses unreliable algorithm for prognostics/diagnostics
H_{07}	DM does not alert when there is a fault because the threshold is not met.
H_{08}	DM alerts when there is no fault because the monitoring parameter is above threshold.
H_{09}	DM generates unclear/ambiguous prognostics/diagnostics.
H_{10}	DM generates prognostics/diagnostics too late.

on-time (see hazard H_{04}). Hazard H_{04} describes a case when DM obtains the condition monitoring data with delay. Since aircraft are operated under tight and dynamic flight schedules, timely scheduling of maintenance tasks cannot be sustained if the diagnostics and prognostics are generated with delay.

Four hazards are identified related to the accuracy of the diagnostics and prognostics algorithms and their results (see hazard H_{05} , H_{06} , H_{07} , and H_{08}). During the brainstorming, erroneous diagnostics and prognostics were identified as the foremost critical hazards (see hazard H_{05}). If the diagnostics/prognostics are erroneous, then either no trigger is generated for necessary maintenance tasks in order to prevent failures/malfunctions, or triggers are generated for redundant, unnecessary maintenance tasks. The former case may cause incidents/accidents, while the latter case may cause additional, unnecessary work and costs [25].

The possible causes of hazard H_{05} is also identified as hazards, i.e., conditions that make diagnostics/prognostics results erroneous. The errors in the data is already discussed as hazards H_{02} and H_{03} . In addition, the used algorithm itself may be unreliable (see hazard H_{06}). In this case, regardless of the quality of the data, the diagnostics/prognostics would be unreliable. Furthermore, two different modes of potential error of the prognostics result were discussed. The first case occurs when DM does not provide an alert when there is a fault, i.e., a *false negative* (see hazard H_{07}). Given a false negative, a necessary maintenance tasks is not triggered. The second case occurs when DM provides an alert when there is actually no fault, i.e., *false positive* (see hazard H_{08}). Although a false positive may not directly affect the safety of the aircraft, it can reduce the efficiency of aircraft maintenance [25]. Moreover, in the case of frequent false positives, the other agents may ignore alerts generated by DM.

Communication issues between agents were also indicated as a hazard during the brainstorming. Assuming that the prognostics results are reliable, an ambiguous or unclear communication between agents about these results was identified as a hazard (see hazard H_{09}). Hazard H_{09} outlines various types of miscommunication regarding the diagnostics and RUL prognostics: i) information or alerts generated by DM are not considered by TG because this information is ambiguous or insufficient to determine effective measurement; ii) the digital platform used for communication between DM and TG presents the information in a non-intuitive form (ambiguous graphics, unclear metadata descriptions).

Lastly, the domain experts discussed the delay in obtaining diagnostics and prognostics. If the diagnostics/prognostics results are generated with delay by DM, then the other agents, and especially TG, do not have enough time to generate necessary tasks to address the issues raised (see hazard H_{10}). In order to cope with tight aircraft flight schedules, it is desirable that diagnostic and prognostic results are delivered to TG, TP, and ME without delay so that necessary tasks can be generated and executed on time. This hazard is related to hazard H_{04} , because H_{04} is likely to trigger hazard H_{10} . Moreover, these hazards are expected to be propagated to all agents.

HAZARDS ASSOCIATED WITH THE TASK GENERATING TEAM (TG)

There are 8 hazards identified for TG under PdAM (see Table 7.5). Among these 8 hazards, 3 hazards are related to the communication with DM, 2 hazards are related to TG's

trust in the diagnostics and prognostics generated by DM, and 3 hazards are related to the process of generating tasks.

Table 7.5: Hazards of data-driven PdAM, associated with the Task Generating team (TG).

ID	Description
<i>H₁₁</i>	TG does not notice the alert from DM.
<i>H₁₂</i>	TG misunderstands alerts from DM.
<i>H₁₃</i>	TG does not generate a task due to misunderstanding regarding the diagnostics/prognostics.
<i>H₁₄</i>	TG does not examine/verify the diagnostics/prognostics.
<i>H₁₅</i>	TG does not rely on diagnostics/prognostics from DM.
<i>H₁₆</i>	TG generates inadequate/ineffective task for a given diagnostics/prognostics.
<i>H₁₇</i>	TG generates two identical tasks from two triggers.
<i>H₁₈</i>	TG generates a task from prognostics too late.

Hazards *H₁₁*, *H₁₂*, and *H₁₃* address the issue of misunderstanding and miscommunication associated with TG under PdAM. Hazard *H₁₁* refers to the case when TG does not notice an alert from DM, and thus necessary maintenance task are not generated. Hazard *H₁₂* refers to the case when TG notice the alert from DM, but misread its meaning. This hazard *H₁₂* is likely to happen when DM generates unclear/ambiguous diagnostics and prognostics (see hazard *H₀₉*). If either hazards *H₁₁* or *H₁₂* occurs, TG is likely to not generate a task as required by the alerts (see hazard *H₁₃*).

Hazards *H₁₄* and *H₁₅* discuss the level of trust of TG in the data-driven PdAM technologies, such as sensors and data-driven diagnostics and prognostics algorithms. Hazard *H₁₅* discusses the case when TG does not use the diagnostics and prognostics generated by DM for task generation due to lack of trust. The trust in the new PdAM technologies is constructed not only based on numerical results from experiments, but also based on an accumulated trust over time between the users and the technology [26]. At the other extreme, hazard *H₁₄* addresses the case when TG fully trusts the new technology and thus TG does not examine or verify the diagnostic and prognostic results. This hazard becomes critical when DM transfers erroneous diagnostics and prognostics (see hazard *H₀₅*). Using erroneous diagnostics and prognostics, TG may not generate necessary tasks (see hazard *H₁₃*) or generates inadequate tasks (see hazard *H₁₆*). Thus, hazard *H₁₄* links the propagation of hazards from *H₀₅* to *H₁₃* and *H₁₆*.

Hazards *H₁₆*, *H₁₇*, and *H₁₈* are related to the case when the generated tasks are not effective in resolving the issue raised. Hazard *H₁₆* addresses the case when inadequate / ineffective tasks are generated. In this case, either additional costs are incurred to perform additional tasks which are actually not necessary, and inadequate tasks are performed for on the aircraft's systems/structures. Hazard *H₁₇* addresses the case when TG generates two identical tasks from two different triggers. For example, when an air conditioning system of an aircraft needs maintenance, this task can be generated as a response to a complaint generated by a flight crew, a report filed by the mechanics, and/or following the prognostics results generated by the data management team (see Figure

7.1). These three independent sources of feedback ensure that an abnormal system performance is indeed reported. However, if the three sources of feedback are not managed properly and as a result multiple identical tasks are generated (see hazard H_{17}), then this leads to confusion in the task generation, task planning, and task execution processes. Lastly, hazard H_{18} discusses the issue of delay in the task generation process. If a task triggered by the diagnostics and prognostics is generated with delay, then the other agents such as TP and ME do not have enough time to plan and execute this task. As such, this hazard is expected to result in missed tasks.

HAZARDS ASSOCIATED WITH THE MECHANICS TEAM (ME)

The two hazards associated with ME under data-driven PdAM are given in Table 7.6. Here, fewer hazards are identified relative to TG and DM since the role of ME under PdAM is envisioned to be similar as in the case of the traditional TBM. However, these two hazards need careful consideration because ME executes the maintenance tasks in the final stage of the aircraft maintenance process, with direct impact on the aircraft airworthiness.

Table 7.6: Hazards of data-driven PdAM, associated with the Mechanics team (ME).

ID	Description
H_{19}	Data-driven PdAM would cause more maintenance tasks triggered by diagnostics/prognostics, leading to a higher risk of human error in maintenance by ME.
H_{20}	ME performs conventional inspection less carefully due to overconfidence in data-driven PdAM.

The main concern discussed during the brainstorming relative to ME under PdAM was the quality of the task execution under PdAM. Hazard H_{19} refers to the case when the mechanics are potentially overloaded under PdAM due to additional tasks that are triggered by diagnostics and prognostics algorithms. Also, an overload may occur for the mechanics if DM provides diagnostic and prognostic results with delay (see hazard H_{10}), or if TG generates tasks with delay (see hazard H_{18}). Under the pressure of executing these data-driven tasks, the risk of human error increases [5, 27]. Furthermore, a premature tasks can be triggered by the prognostics, which increases the chance of having human errors [28].

Hazard H_{20} describes the case when ME performs the conventional inspection less carefully due to overconfidence in PdAM. This is the result of the ME over-trusting the new PdAM technologies. This hazard is similar to hazard H_{14} for TG.

7.4. VALIDATION OF THE IDENTIFIED HAZARDS USING REPORTED AIRCRAFT INCIDENTS

In this section, we discuss past aircraft accidents/incidents as a means to validate the hazards identified in the brainstorming session. We first outline the chronology of the events leading to these incidents based on the final, official investigation reports. Using

these reports, we identify similar hazards as those identified in the brainstorming session (see Tables 7.4-7.6). This analysis shows that the hazards identified in the brainstorming session are also observed in the context of past incidents.

NUISANCE FALSE POSITIVE ALERTS LEAD TO AGENTS IGNORING A TRUE POSITIVE ALERT

An aircraft incident reported in 2017 illustrates how the inadequate handling of alerts from ACMS contributes to the incident [29]. On 29-April-2017 (Day 0), an aircraft was dispatched while the left air conditioning system (ACS) had been disabled, in accordance with the Minimum Equipment List. During the flight, the cabin pressure was lost because the right ACS failed while the left ACS was disabled. The incident investigation established that the component on the right ACS had been changed 11 days before the day of the incident (Day -11). After the aircraft returned to service at Day -9, the on-board aircraft health monitoring (AHM) system sent an alert message to the operator's AHM ground-based data system and their engineering department⁴, indicating that a 'high leakage/low inflow' of the cabin pressurization system had been detected. The operator assessed the message and the necessary task was planned at Day +6. Thereafter, during all the subsequent flights between Day -9 and Day 0, maintenance alert messages were sent by AHM, but no further action was taken by the operator.

From the investigation report of this incident, we identify the following hazards that contributed to the incident. The operator generated an inadequate task with too late due date (see hazard *H*₁₆). More importantly, the continuous alert was not taken seriously by the operator because they regarded this as 'nuisance' (see hazard *H*₁₅).

In addition, an indirect, but crucial hazard is identified — the generated diagnostics results had been frequently faulty in the past (see hazards *H*₀₅ and *H*₀₈), and therefore the engineering department classified the actually correct alert as faulty (see hazard *H*₁₃). Regarding these hazard, we quote from the investigation report [29]:

The operator later stated that the AHM system provides just over 1,200 maintenance alerts. From experience, some maintenance alert messages are inadvertently triggered, which has led to refinements to improve the robustness of the system and reduce the level of 'nuisance' alerts. The operator had seen alert message 21-0209-C740 triggered 'intermittently' on other aircraft before and this had caused maintenance staff to question the reliability of this particular alert message.

This incident shows that it is critical to ensure the reliability of the diagnostics / prognostics algorithms and the alert systems, in order to make the agents trust the new PdAM technologies.

DAMAGE NOT IDENTIFIED BY SENSORS AND INSPECTIONS

Several incidents are caused by the damage done during hard landings, which was identified neither by the on-board sensors nor by inspections [30, 31, 32]. Generally, on-board aircraft condition monitoring systems (ACMS) indicate hard landings to the flight

⁴Aircraft health monitoring (AHM) ground-based data system and their engineering department perform the role of DM and TG in Figure 7.1

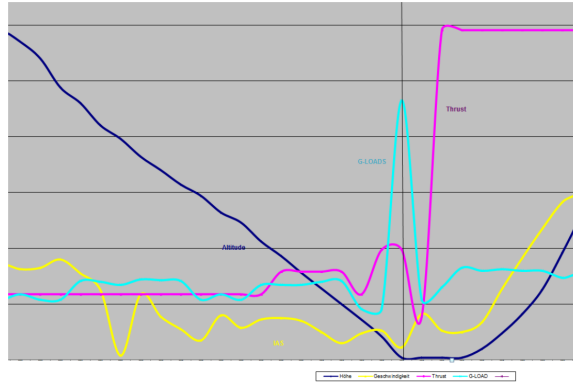


Figure 7.2: Curves for the G-load (cyan), altitude (blue), indicated air speed (yellow), and thrust (magenta) parameters during the hard landing.

Image source: Civil Aviation Accident and Incident Investigation Commission (CAAIC). *Incident involving an Airbus A-321-211 registration D-ASTP, operated by Germania, at the Fuerteventura Airport, Canary Islands, Spain on 16 July 2016*. 2016. [30]

crew. In this case, the pilots and the mechanics conduct inspections to identify and evaluate the potential damage, following the manuals.

In 2016, an aircraft damaged by a hard landing was released without addressing the damage [30]. Although the subsequent flight was completed uneventfully, it was found later that the aircraft was in an unsafe condition due to the serious damage made by the previous hard landing.

In the investigation report, it was found that the ACMS did not submit the ‘G-Load’ report to the pilots because the peak load persisted for 1 second only (see Figure 7.2), while the report is issued when the load persists for at least 2 seconds [30]. Also, the ACMS sent ‘A15 hard landing report’ to the Maintenance Operation Center (MOC)⁵, but the MOC was not able to interpret the report properly (see hazard H_{12}) and on-time (see hazards H_{10} and H_{18}). Because the subsequent inspection did not find any damage (see hazard H_{20}), the aircraft was released back to service.

Two similar aircraft incidents occurred in 2013 and 2008 [31, 32]. In both cases, the damages to the landing gears were not identified after hard landings. A contributing factor to these incidents was that the on-board ACMS did not trigger an alert for hard landing since the predefined load threshold had not been exceeded (see hazard H_{07}). For the incident in 2008, the engineers reasoned that no inspection was needed because the recorded parameters had not exceeded a predefined threshold, which is in accordance with the aircraft maintenance manual [32]. For the incident in 2013, inspections were performed regardless of the ACMS alert, but the damage was not identified (see hazard H_{20}). According to the investigation, the other contributing factors were the bad meteorological conditions during the outdoor inspection, and the use of inspection procedures which were not consistent with the aircraft maintenance manual [31].

These incidents show that the parameters and algorithms used for ACMS need to be

⁵Maintenance Operation Center (MOC) performs the role of DM and TG in Figure 7.1

updated continuously based on the actual operation data in order to properly identify hard landing or other abnormal events (see hazard H_{07}). In addition, the inspections carried out by mechanics need to be performed carefully, especially when there is a conflict between reports submitted by flight crew and aircraft condition monitoring systems (see hazard H_{20}).

UNIDENTIFIED DAMAGE DUE TO INCOMPREHENSIBLE DATA PRESENTATION

In 2016, a helicopter lost its yaw control during landing [33]. The helicopter has in place the Health and Usage Monitoring System (HUMS)⁶, which monitors the condition parameters such as engine vibration, rotor track balance, engine shaft balance, etc. A day before the incident (Day -1), during flight, HUMS recorded vibration data including a series of exceedences related to the tail rotor pitch change shaft (TRPCS) bearing. In the routine maintenance following this flight, the HUMS data was downloaded and analyzed. During the analysis, an abnormality for the tail rotor gear box bearing was detected, but the exceedence was not identified. During the first flight of the day of the incident (Day 0), the HUMS recorded further exceedence. However, it was planned to download and analyze the data only after the helicopter returns to the base. During the lift-off of the second flight in Day 0, the helicopter went through an uncommanded yaw. However, this was regarded as the influence of the wind on the helicopter. During landing of the same flight, the helicopter totally lost yaw control and landed expeditiously and heavily. The root cause of the lost yaw control was identified as the damage on the TRPCS caused by the failed bearing. The following two contributing factors were discussed in the investigation report [33]:

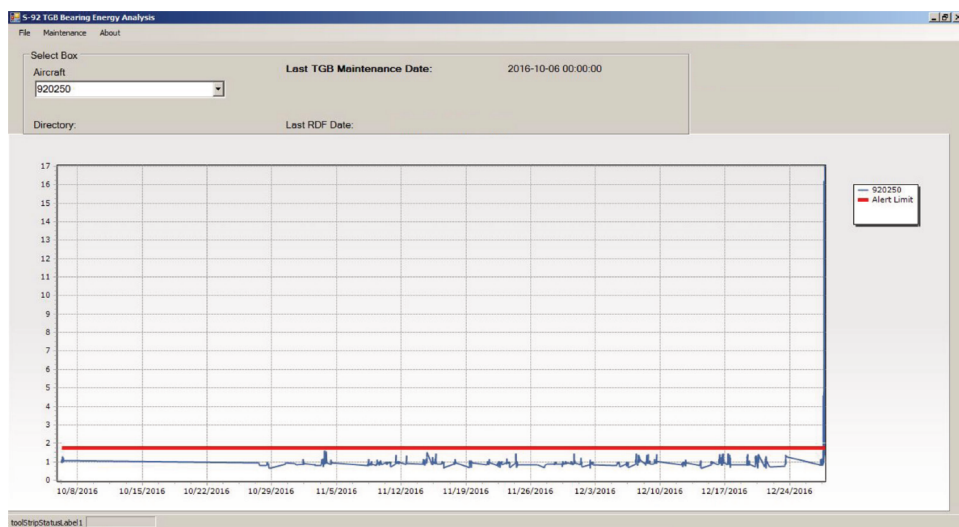
Impending failure of the TRPCS bearing was detected by HUMS but was not identified during routine maintenance due to human performance limitations and the design of the HUMS Ground Station Human Machine Interface.

The HUMS Ground Station software in use at the time had a previously-unidentified and undocumented anomaly in the way that data could be viewed by maintenance personnel. The method for viewing data recommended in the manufacturer's user guide was not always used by maintenance personnel.

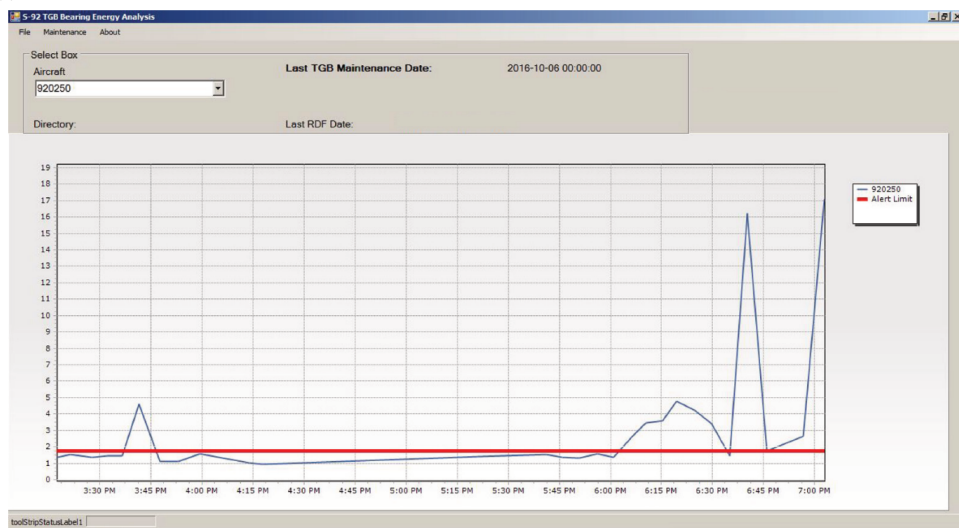
For this incident, we identify the hazards related to the unclear communication (see hazards H_{09} , H_{11} , H_{12} , and H_{13}), and the delayed data/information sharing (see hazard H_{18}). The damage to TRPCS was properly detected by the HUMS before the incident, but this was not identified and resolved by the operator⁷ (see hazards H_{11} , H_{12} , and H_{13}). The first contributing factor was the design of the HUMS Ground Station Human Machine Interface [33]. The information available through this interface needs to be zoomed in to identify the exceedence (see Figure 7.3), but the two engineers did not address this (see hazard H_{09} , H_{11} and H_{12}). As a result, a proper inspections was not conducted (see hazard H_{13}). In addition, the HUMS data was not shared on-line, rather the storage card was supposed to be brought back to the base. Thus, the exceedence

⁶Health and Usage Monitoring System (HUMS) records the status of helicopter to detect or predict defects. This system performs the role of ACMS for aircraft in Figure 7.1.

⁷These operators are performing the role of TG in Figure 7.1.



(a)



(b)

Figure 7.3: Human Machine Interface of the helicopter that had damage on TRPCS.

(a) Time-history chart. The exceedence of the monitoring parameter is shown at the right end of the graph, but it is not clearly visible.

(b) Time-history chart zoomed to the last flight on 27 December 2016. The exceedence is obvious.

Image source: Air Accidents Investigation Branch (AAIB), *Report on the accident to Sikorsky S-92A, G-WNSR West Franklin wellhead platform, North Sea on 28 December 2016*. 2018. [33]

recorded during the first flight was not reported (see hazard H_{04}). Moreover, the global support team⁸ who received the HUMS data of the previous day (Day –1) identified the exceedence and contacted the operator, but the communication was not done on time (see hazards H_{04} and H_{18}) as the incident already occurred by the time the support team transmitted their report.

This case shows the importance of the digital communication platform for data-driven PdAM. The digital platform should visualize the data in an intuitive manner and highlight crucial information to prevent the hazards such as hazards H_{09} , H_{11} , H_{12} , and H_{13} . In addition, the on-line data sharing is needed to prepare necessary maintenance tasks in advance (see hazard H_{18}).

With the analysis above, we validate the hazard list identified during the brainstorming session by revealing similar hazards encountered for actual incidents.

7.5. EMERGING CHALLENGES OF PREDICTIVE AIRCRAFT MAINTENANCE

In the context of the identified hazards of data-driven PdAM, we discuss its three main challenges. In Figure 7.4, we group the hazards based on the associated maintenance agents, and mark each hazard based on the associated emerging challenges.

RELIABILITY OF NEW TECHNOLOGIES

The biggest challenge is to guarantee the reliability of new technologies introduced in data-driven PdAM, e.g., aircraft condition monitoring systems (ACMS), diagnostics and prognostics algorithms, and decision support systems of PdAM. 9 out of 20 hazards are related to the reliability of new technologies (see hazards H_{01} , H_{02} , H_{03} , H_{04} , H_{05} , H_{06} , H_{07} , H_{08} , and H_{10}). The majority of the maintenance experts perceive the low reliability of diagnostics and prognostics algorithms as a main trigger for most of the hazards associated with data-driven PdAM. Therefore, it is recommended to test the data-driven diagnostics and prognostics algorithms using multiple operational data sets. After all, adequate approval procedures for the design and implementation of data-driven PdAM is needed.

COMMUNICATION BETWEEN THE MAINTENANCE AGENTS

The second challenge is related to communication between the maintenance agents, which is related to 5 out of 20 hazards (see hazards H_{09} , H_{11} , H_{12} , H_{13} , and H_{17}). In this light, the maintenance experts emphasize the need for an intuitive and effective digital platform to support timely communication at all levels of the data-driven PdAM. Interactive user interfaces and informative visualizations are seen as a means to avoid missed alerts [34]. However, not enough studies discuss user interfaces on aircraft maintenance, although intensive studies are made for other data-driven technologies, such as self-driving cars [35]. Only a few studies discuss the user interface supporting aircraft maintenance tasks (agent ME) [36, 37]. Thus, further investigation is necessary to improve the effectiveness of communication between all maintenance agents, especially

⁸The global support team performs as DM and TG in Figure 7.1.

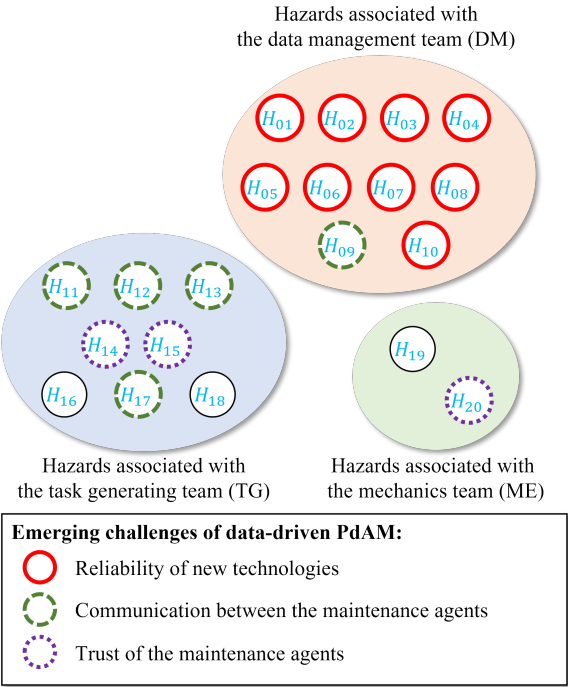


Figure 7.4: The identified hazards and the emerging challenges of data-driven predictive aircraft maintenance (PdAM). The description of hazards associated with the DM, TG, and ME are given in Tables 7.4, 7.5, and 7.6, respectively.

agent TG who is associated with the most number of hazards related to communication (see Table 7.5 and Figure 7.4).

TRUST OF THE MAINTENANCE AGENTS

A third challenge for data-driven PdAM is to build the trust of the maintenance agents in the new data-driven technologies, which is related to 3 out of 20 hazards (see hazards H_{14} , H_{15} , and H_{20}). The trust in a new technology is based on more than just having systems and algorithms of high accuracy [26]. In fact, *trust* is equally based on users' personal cognition on the reputation of these new technologies (cognitive trusting base), their understanding that these new technologies benefit them (calculative trusting base) and their confidence in the human operators behind these new technologies (institutional trusting base), [26]. For the case of aircraft maintenance, the process is even more complex, with multiple agents who use different data-driven technologies locally and who interact with each other at the system level. Therefore, we should build trust both at the level of individual agent, as well as at system-level. At the individual level, trust needs to be built between each agent and the new technologies that they use. At the system level, trust needs to be built in the information transferred from one agent to another.

7.6. CONCLUSIONS

In this chapter, we identify hazards associated with the introduction of data-driven predictive aircraft maintenance (PdAM), and discuss the emerging challenges of implementing data-driven PdAM. As a first step, the main agents of data-driven PdAM and their interactions are recognized. Then, a structured brainstorming for hazard identification is conducted with aircraft maintenance experts, each representing one of the maintenance agents. We focus on the emerging hazards associated with the adoption of new technologies, such as aircraft condition monitoring systems (ACMS), data-driven diagnostics and prognostics algorithms, and decision support systems for PdAM. As a result, 20 emerging hazards are uncovered for data-driven PdAM. Two agents, the data management team and task generating team, are associated with the largest number of new hazards of data-driven PdAM. These hazards are validated in the context of past aircraft incidents that occurred between 2008 and 2013.

Following the analysis of the hazards, we discuss three main challenges for safe implementation of data-driven PdAM: i) guaranteeing the reliability of new data-driven technologies of PdAM, ii) designing intuitive communication platforms that can facilitate communication between agents under PdAM, and iii) building the agent's trust in the new data-driven PdAM process. These challenges guide the future research direction for the successful implementation of data-driven PdAM.

REFERENCES

- [1] Air Transportation Association. *ATA MSG-3, Operator/Manufacturer Scheduled Maintenance Volume 1 - Fixed Wing Aircraft*. Vol. 1 - Fixed. Air Transport Association of America, 2013.
- [2] J. Lee and M. Mitici. "An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets". In: *Reliability Engineering and System Safety* 202 (2020), p. 107052. ISSN: 0951-8320. DOI: [10.1016/j.ress.2020.107052](https://doi.org/10.1016/j.ress.2020.107052).
- [3] X. Chen, J. Yu, D. Tang, and Y. Wang. "Remaining useful life prognostic estimation for aircraft subsystems or components: A review". In: *Proceedings - IEEE 2011 10th International Conference on Electronic Measurement and Instruments, ICEMI 2011* 2 (2011), pp. 94–98. DOI: [10.1109/ICEMI.2011.6037773](https://doi.org/10.1109/ICEMI.2011.6037773).
- [4] A. Hobbs and A. Williamson. "Associations between errors and contributing factors in aircraft maintenance". In: *Human Factors* 45.2 (2003), pp. 186–201. ISSN: 00187208. DOI: [10.1518/hfes.45.2.186.27244](https://doi.org/10.1518/hfes.45.2.186.27244).
- [5] Y. H. Chang and Y. C. Wang. "Significant human risk factors in aircraft maintenance technicians". In: *Safety Science* 48.1 (2010), pp. 54–62. ISSN: 09257535. DOI: [10.1016/j.ssci.2009.05.004](https://doi.org/10.1016/j.ssci.2009.05.004).
- [6] International Maintenance Review Board Policy Board (IMRBPB). *Aircraft Health Monitoring (AHM) integration in MSG-3, IP180*. 2018.
- [7] Eurocontrol Safety Regulation Commission. *EUROCONTROL Safety Regulatory Requirement - ESARR 4 Risk Assessment and Mitigation in ATM*. Tech. rep. Belgium, Brussels: Eurocontrol, 2001.

- [8] Federal Aviation Administration. *120-92B - Safety Management Systems for Aviation Service Providers*. 2015.
- [9] I. Cameron, S. Mannan, E. Németh, S. Park, H. Pasman, W. Rogers, and B. Seligmann. "Process hazard analysis, hazard identification and scenario definition: Are the conventional tools sufficient, or should and can we do much better?" In: *Process Safety and Environmental Protection* 110 (2017), pp. 53–70. ISSN: 09575820. DOI: [10.1016/j.psep.2017.01.025](https://doi.org/10.1016/j.psep.2017.01.025).
- [10] P. Baybutt. "A critique of the Hazard and Operability (HAZOP) study". In: *Journal of Loss Prevention in the Process Industries* 33 (2015), pp. 52–58. ISSN: 09504230. DOI: [10.1016/j.jlp.2014.11.010](https://doi.org/10.1016/j.jlp.2014.11.010).
- [11] H. H. de Jong, H. A. P. Blom, and S. H. Stroeve. "How to identify unimaginable hazards?" In: *Proc. 25th ISSC*. Baltimore, Maryland, 2007, pp. 456–465.
- [12] H. H. de Jong, H. A. P. Blom, and S. H. Stroeve. "Unimaginable hazards and emergent behavior in air traffic operations". In: *Proc. ESREL2007 Conference*. Ed. by Aven & Vinnem. Stavanger, Norway: Taylor & Francis, 2007.
- [13] H. A. P. Blom, S. H. Stroeve, and T. Bosse. "Modelling of potential hazards in agent-based safety risk analysis". In: *Europe Air Traffic Management Research and Development Seminar* (2013).
- [14] H. Bağan and E. Gerede. "Use of a nominal group technique in the exploration of safety hazards arising from the outsourcing of aircraft maintenance". In: *Safety Science* 118, June (2019), pp. 795–804. ISSN: 18791042. DOI: [10.1016/j.ssci.2019.06.012](https://doi.org/10.1016/j.ssci.2019.06.012).
- [15] O. Čokorilo and G. Dell'Acqua. "Aviation Hazards Identification Using Safety Management System (SMS) Techniques". In: *16th International conference on transport science (ICTS 2013)* (2013), pp. 66–73.
- [16] B. E. Smith, H. H. de Jong, and M. H. Everdij. "A prognostic method to identify hazards for future aviation concepts". In: *26th International congress of the Aeronautical Sciences, ICAS*. 2008, pp. 1–12. ISBN: 9781605607153.
- [17] C. M. Macal and M. J. North. "Tutorial on agent-based modelling and simulation". In: *Proceedings of the Winter Simulation Conference*. IEEE, 2005, p. 14. DOI: [10.1057/jos.2010.3](https://doi.org/10.1057/jos.2010.3).
- [18] C. M. Macal. "Tutorial on agent-based modeling and simulation: ABM design for the zombie apocalypse". In: *Proceedings of the 2018 Winter Simulation Conference*. 2018, pp. 207–221. ISBN: 9781538665725. DOI: [10.1109/WSC.2018.8632240](https://doi.org/10.1109/WSC.2018.8632240).
- [19] A. K. Jardine, D. Lin, and D. Banjevic. "A review on machinery diagnostics and prognostics implementing condition-based maintenance". In: *Mechanical Systems & Signal Processing* 20 (2006), pp. 1483–1510. ISSN: 08883270. DOI: [10.1016/j.ymssp.2005.09.012](https://doi.org/10.1016/j.ymssp.2005.09.012).
- [20] X. S. Si, W. Wang, C. H. Hu, and D. H. Zhou. "Remaining useful life estimation - A review on the statistical data driven approaches". In: *European Journal of Operational Research* 213.1 (2011), pp. 1–14. ISSN: 03772217. DOI: [10.1016/j.ejor.2010.11.018](https://doi.org/10.1016/j.ejor.2010.11.018).

- [21] Z. Zhang, X. Si, C. Hu, and Y. Lei. “Degradation data analysis and remaining useful life estimation: A review on Wiener-process-based methods”. In: *European Journal of Operational Research* 271.3 (2018), pp. 775–796. ISSN: 03772217. DOI: [10.1016/j.ejor.2018.02.033](https://doi.org/10.1016/j.ejor.2018.02.033).
- [22] R. Meissner, F. Raschdorff, H. Meyer, and T. Schilling. “Digital Transformation in Maintenance on the Example of a Tire Pressure Indicating System”. In: *International workshop on Aircraft System Technologies*. Hamburg, Germany, 2019, pp. 1–10.
- [23] J. Hale. “Boeing 787 from the Ground Up”. In: *Aero* (2006), pp. 17–23.
- [24] B. A. Nijstad and W. Stroebe. “How the group affects the mind: A cognitive model of idea generation in groups”. In: *Personality and Social Psychology Review* 10.3 (2006), pp. 186–213. ISSN: 10888683. DOI: [10.1207/s15327957pspr1003_1](https://doi.org/10.1207/s15327957pspr1003_1).
- [25] N. B. Hölzel and V. Gollnick. “Cost-benefit analysis of prognostics and condition-based maintenance concepts for commercial aircraft considering prognostic errors”. In: *Annual Conference of the Prognostics and Health Management Society*. 2015, pp. 1–16.
- [26] X. Li, T. J. Hess, and J. S. Valacich. *Why do we trust new technology? A study of initial trust formation with organizational information systems*. Vol. 17. 2008, pp. 39–71. ISBN: 9105216311. DOI: [10.1016/j.jsis.2008.01.001](https://doi.org/10.1016/j.jsis.2008.01.001).
- [27] R. Islam, F. Khan, R. Abbassi, and V. Garaniya. “Human error assessment during maintenance operations of marine systems – What are the effective environmental factors?” In: *Safety Science* 107. April (2018), pp. 85–98. ISSN: 18791042. DOI: [10.1016/j.ssci.2018.04.011](https://doi.org/10.1016/j.ssci.2018.04.011).
- [28] N. Safaei. “Premature Aircraft Maintenance: A Matter of Cost or Risk?” In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* PP (2019), pp. 1–11. ISSN: 2168-2216. DOI: [10.1109/TSMC.2019.2895207](https://doi.org/10.1109/TSMC.2019.2895207).
- [29] Air Accidents Investigation Branch (AAIB). *Emergency descent due to loss of cabin pressure, en route from London Heathrow to Delhi, 29 April 2017*. 2018.
- [30] Civil Aviation Accident and Incident Investigation Commission (CAAIC). *Incident involving an Airbus A-321-211 registration D-ASTP, operated by Germania, at the Fuerteventura Airport, Canary Islands, Spain on 16 July 2016*. 2016.
- [31] Bureau d’Enquêtes et d’Analyses (BEA). *Hard landing, inappropriate stopover maintenance procedure, take-off with a substantially damaged aeroplane*. 2013.
- [32] Air Accidents Investigation Branch (AAIB). *AAIB Bulletin: 6/2009 EW/C2008/07/05 Airbus A321-231, G-MARA, 28 July 2008*. 2009.
- [33] Air Accidents Investigation Branch (AAIB). *Report on the accident to Sikorsky S-92A, G-WNSR West Franklin wellhead platform, North Sea on 28 December 2016*. 2018.
- [34] M. J. Eppler and M. Aeschmann. “A systematic framework for risk visualization in risk management and communication”. In: *Risk Management* 11.2 (2009), pp. 67–89. ISSN: 14603799. DOI: [10.1057/rm.2009.4](https://doi.org/10.1057/rm.2009.4).

- [35] S. Kim, R. van Egmond, and R. Happee. “Effects of user interfaces on take-over performance: A review of the empirical evidence”. In: *Information* 12.4 (2021), pp. 1–16. ISSN: 20782489. DOI: [10.3390/info12040162](https://doi.org/10.3390/info12040162).
- [36] H. Witt, T. Nicolai, and H. Kenn. “Designing a wearable user interface for hands-free interaction in maintenance applications”. In: *Proceedings - Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2006* 2006 (2006), pp. 652–655. DOI: [10.1109/PERCOMW.2006.39](https://doi.org/10.1109/PERCOMW.2006.39).
- [37] S. Utzig, R. Kaps, S. M. Azeem, and A. Gerndt. “Augmented Reality for Remote Collaboration in Aircraft Maintenance Tasks”. In: *IEEE Aerospace Conference Proceedings* 2019-March (2019), pp. 1–10. ISSN: 1095323X. DOI: [10.1109/AERO.2019.8742228](https://doi.org/10.1109/AERO.2019.8742228).

8

CONCLUSION

8.1. REVIEW OF RESEARCH OBJECTIVES

The [Research Objectives](#) have been addressed over the course of this dissertation, and are reflected upon in this section.

Obj.1 Construct mathematical models of predictive aircraft maintenance and assess its performance.

In Chapter 2, I constructed a Petri net model of predictive aircraft maintenance (PdAM). I focused on the interactions between various agents during aircraft maintenance. Six agents were modeled: aircraft, data management team, task generating team, task planning team, mechanics team, and flight crews. Their interactions, such as transferring condition monitoring data, triggering alarms based on Remaining-Useful-Life (RUL) prognostics, etc., were formulated by means of stochastically and dynamically colored Petri nets (SDCPNs). This Petri net model also integrated the degradation model of aircraft landing gear brakes, which was validated based on the operational data collected from a fleet of Boeing 787 aircraft. I used the Petri net model to simulate the consequences of PdAM, such as the number of performed inspections and replacements, the wasted life of components, and the probability of a system failure. Based on these results, the performance of PdAM was assessed. With this study, the first objective of constructing mathematical models of PdAM and evaluating its performance, was achieved.

Obj.2 Identify the key performance indicators (KPIs) of predictive aircraft maintenance, and analyze their trade-offs.

This research objective was addressed in Chapter 3. I identified two sets of key performance indicators (KPIs) representing reliability and cost-efficiency of aircraft maintenance. The KPIs representing reliability are the number of component replacements and the mean number of flight cycles to component replacements (MCTR). The KPIs of cost-efficiency are the number of degradation incidents, the number of unscheduled maintenance tasks, and the total accrued flight delay due to maintenance. Using the model

of PdAM, I showed that the KPIs of reliability are all positively correlated, while they conflict against the KPIs of cost-efficiency. For instance, it was observed that flight delay increases together as the number of degradation incidents increases; but more flight delays are expected when the number of component replacements is reduced. This observation suggested that the goal of PdAM should be the simultaneous maximization of reliability and cost-efficiency.

Obj.3 Integrate RUL prognostics into predictive aircraft maintenance planning.

This dissertation tackled **Obj.3** by dividing it into three levels of PdAM: component-level (**Obj.3.1**), fleet-level (**Obj.3.2**), and strategy-level (**Obj.3.3**). I proposed three optimization approaches that best address the main challenges of the three levels. For component-level PdAM, a deep reinforcement learning approach was used to consider the uncertainty associated with RUL prognostics (Chapter 4). For fleet-level PdAM, an integer linear programming (ILP) considered the operational requirements of an aircraft fleet (Chapter 5). For strategy-level PdAM, a multi-objective optimization was proposed to efficiently optimize PdAM strategies considering conflicting objectives (Chapter 6). Finally, this dissertation provided methodologies to integrate RUL prognostics into predictive aircraft maintenance planning.

Obj.3.1 Optimize a predictive maintenance plan for an aircraft component based on RUL prognostics and associated uncertainty (Component-level PdAM).

In Chapter 4, I developed probabilistic RUL prognostics for an aircraft turbofan engine. My approach provided the uncertainty information of RUL prognostics by means of a probability distribution. The estimated RUL distribution was used to schedule engine replacements, using a deep reinforcement learning (DRL) approach. The DRL approach resulted in optimal replacement schedules that minimize costs while avoiding unexpected engine failures. This framework successfully addressed one of the biggest challenges of PdAM at the component-level, which is the maintenance planning considering the uncertainty of RUL prognostics. The component-level PdAM was illustrated for aircraft turbofan engines.

Obj.3.2 Optimize a predictive maintenance plan for a fleet of aircraft with multiple components, considering operational requirements (Fleet-level PdAM).

A framework to plan predictive maintenance for a fleet of aircraft was proposed in Chapter 5. The fleet-level PdAM integrated RUL prognostics and operational requirements such as required flight schedules and limited hangar availability. I formulated the PdAM at fleet-level as an integer linear programming (ILP) problem, where the operational requirements were given as constraints. The objective of ILP was to minimize maintenance costs, including the cost of wasted component life, the penalty of overdue replacements, and the setup cost at the hangar. This approach was illustrated for the maintenance of landing gear brakes, whose RUL was predicted by a Bayesian regression model. Compared with traditional time-based maintenance strategies, the proposed framework reduced up to 20% of maintenance costs.

Obj.3.3 Design a predictive maintenance strategy by optimizing parameters such as thresholds of RUL, considering multiple objectives (Strategy-level PdAM).

In Chapter 6, I designed a Pareto optimal PdAM strategy. First, I proposed an efficient algorithm to optimize the parameters of maintenance strategies, using Gaussian process (GP) learning models. Using this algorithm, I optimized the PdAM strategies for the maintenance of landing gear brakes. The results showed that the optimal PdAM strategy Pareto-dominates other strategies by balancing the conflict between reliability and efficiency. Although the case study considered landing gear brakes only, the proposed methodology can be applied to design multi-objective maintenance of other general aircraft components.

Obj.4 Identify emerging challenges of predictive aircraft maintenance.

Challenges of PdAM were identified in Chapter 7. In particular, I was interested in new challenges associated with introducing aircraft health monitoring (AHM) systems and RUL prognostics. The main difficulty in achieving this objective was the lack of data and experience obtained from practice. To overcome this, I had organized brainstorming sessions with experts in aircraft maintenance. During the brainstorming session, the model of PdAM developed in Chapter 2 was used to facilitate the discussion. The results showed that the reliability of new technologies and decision support systems is the foremost concern. Other challenges are the timely and accurate communication, and users' trust in the new technologies and decision support systems.

8.2. CONCLUSIONS

Based on the findings through this dissertation, the following conclusions are drawn:

1. The first steps to design predictive aircraft maintenance are to identify the interactions of all stakeholders in the aircraft maintenance process, to understand their key performance indicators.

Aircraft maintenance is performed by various stakeholders who exchange information, make decisions, and take actions (Chapter 2). We should clearly identify their interactions to adequately integrate aircraft health monitoring (AHM) systems and Remaining-Useful-Life (RUL) prognostics into the existing aircraft maintenance process. Moreover, the key performance indicators (KPIs) of different stakeholders are often subjected to trade-offs (Chapter 3). A multi-objective analysis of the KPIs is required to define proper objectives of predictive aircraft maintenance (PdAM) and to identify their trade-offs.

2. Predictive aircraft maintenance will be most beneficial for expensive and safety-critical aircraft components.

Throughout the case studies in Chapters 4-6, we have shown that PdAM effectively prevents component failures and unscheduled maintenance tasks. Such benefits are maximized when considering components whose failures are critical or whose maintenance is expensive.

3. Uncertainty in Remaining-Useful-Life prognostics must be considered when planning predictive aircraft maintenance.

In order to plan optimal predictive aircraft maintenance, the uncertainty of RUL prognostics should be taken into account, especially considering trade-offs between too early and too late replacements, as shown in the case studies in Chapters 4-6.

4. Understanding the mechanism of degradation/failure is essential for Remaining-Useful-Life prognostics, especially when considering data-driven approaches.

In this dissertation, two data-driven models are proposed for probabilistic RUL prognostics of two aircraft components: deep convolutional neural networks (CNNs) for turbofan engines (Chapter 4), and Bayesian regression model for landing gear brakes (Chapters 5-6). Each model is selected based on understanding the degradation trend and condition monitoring data. For example, because the degradation trend of landing gear brakes is relatively linear, a Bayesian linear regression model is already effective, and complex models such as CNNs are not worth the effort.

5. In the design of predictive aircraft maintenance strategy, one should apply a multi-objective approach.

Reliability and cost are two major objectives of PdAM, which are often in conflict with each other (Chapter 2). In theory, we can minimize cost while we satisfy a certain level of reliability given as a requirement (constraint). In practice, however, the required level of reliability is hardly known before the optimization, especially when considering new complex systems such as PdAM. Therefore, the design of PdAM should optimize these objectives simultaneously to identify Pareto optimal designs (Chapter 6).

6. Non-technical challenges such as legislation, business models, and people's trust should not be overlooked.

Many of the emerging challenges of PdAM in Chapter 7 are in fact non-technical challenges. Current academic research on PdAM mostly focuses on how to improve its performance, e.g., in its reliability and efficiency. However, the findings in Chapter 7 suggest that it is equally important to discuss how to convince the benefit of PdAM to the relevant stakeholders, regulatory body, business owners, and others involved in PdAM.

8.3. NOVELTY AND SCIENTIFIC CONTRIBUTIONS

This section highlights the novelty and scientific contributions of this dissertation. The main scientific contributions are the frameworks to optimize predictive aircraft maintenance (PdAM), and to integrate Remaining-Useful-Life (RUL) prognostics into maintenance planning. Four main scientific contributions are as follows:

1. This dissertation is the first study to identify all stakeholders in PdAM and model their interactions using stochastic Petri nets. Furthermore, an integrated framework is proposed to assess both the reliability and efficiency of

aircraft maintenance simultaneously. The corresponding contribution led to the following journal publication and conference presentation.

- **J. Lee** and M. Mitici, "An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets," *Reliability Engineering and System Safety*, vol. 202, p. 107052, 2020.

- **J. Lee** and M. Mitici, "Predictive aircraft maintenance: modeling and analysis using stochastic Petri nets," in *Proceedings of the 31st European Safety and Reliability Conference*, pp. 146-153, Angers, France, September 19–23, 2021.

2. This dissertation is the first study to integrate probabilistic RUL prognostics into maintenance planning. The probability distribution of RUL of an aircraft engine is estimated using deep convolutional neural networks and Monte Carlo dropout. The estimated RUL distribution is used for predictive maintenance planning based on a deep reinforcement learning approach. This dissertation provides new methods to quantify the uncertainty of RUL prognostics and to use this information for optimal maintenance planning. The corresponding contribution was submitted as the following publication under review.

- **J. Lee** and M. Mitici, "Deep reinforcement learning for predictive aircraft maintenance using probabilistic Remaining-Useful-Life prognostics," *Reliability Engineering and System Safety*, 2022.

3. This dissertation is the first study to optimize replacement schedules of landing gear brakes of a fleet of aircraft considering RUL prognostics. The proposed approach using integer linear programming provides a basis to optimize predictive maintenance for multiple aircraft and multi-component systems. The corresponding contribution was presented at the following conference and was awarded *Best Paper Award 2nd Prize, European Conference of the Prognostics and Health Management Society* in 2022.

- **J. Lee**, I. de Pater, S. Boekweit, and M. Mitici, "Remaining-Useful-Life prognostics for opportunistic grouping of maintenance of landing gear brakes for a fleet of aircraft," in *Proceedings of the 7th European Conference of the Prognostics and Health Management Society 2022*, pp.278-285, Turin, Italy, July 6–8, 2022.

4. This dissertation proposes a new framework to optimize thresholds and safety margins of PdAM strategies considering conflicting objectives. A novel algorithm using Gaussian process (GP) learning is proposed and illustrated for the maintenance of landing gear brakes. This framework can be used to design new PdAM strategies for other aircraft components. The corresponding contribution led to the following journal publication and conference proceeding, which is awarded *Innovation in Transport Applications, European Safety and Reliability Conference* in 2022.

- **J. Lee** and M. Mitici, "Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling," *Reliability Engineer-*

ing and System Safety, vol. 218, p. 108123, 2022.

- **J. Lee**, M. Mitici, S. Geng, and M. Yang, “Designing reliable, data-driven maintenance for aircraft systems with applications to the aircraft landing gear brakes,” in *Proceedings of the 32nd European Safety and Reliability Conference (ESREL)*, pp.25-32, Dublin, Ireland, August 28 – September 1, 2022.

Besides these optimization frameworks, this dissertation builds the foundation for future research on PdAM as follows:

5. This dissertation is the first study to identify key performance indicators (KPIs) representing various interests of stakeholders in aircraft maintenance, and analyzing the trade-offs of these KPIs. Based on this analysis, objectives for the optimization of PdAM are suggested. The corresponding contribution was presented at the following conference and awarded *Thomas L. Fagan, Jr., RAMS Student Paper Award 1st Place, Reliability and Maintainability Symposium* in 2021.

- **J. Lee** and M. Mitici, “Multi-objective analysis of condition-based aircraft maintenance strategies using discrete event simulation,” in *2021 Annual Reliability and Maintainability Symposium (RAMS)*, pp. 1–6, Orlando, FL, USA, May 24–27, 2021.

6. This dissertation is the first study to identify emerging challenges for predictive aircraft maintenance after the regulatory discussion on the integration of AHM in aircraft maintenance in 2018 ¹. Based on this, future research directions are suggested. This result was submitted as the following publication under review.

- **J. Lee**, M. Mitici, H. A. P. Blom, P. Bieber, and F. Freeman, “Identification of emerging hazards for the data-driven predictive aircraft maintenance process,” submitted to *Safety Science* in 2021 (under review).

8.4. LIMITATIONS AND RECOMMENDATION FOR FUTURE WORK

This dissertation provides novel methodologies to optimize predictive aircraft maintenance (PdAM). Even so, much work remains to be done before we can perform actual aircraft maintenance using PdAM.

IMPROVING RUL PROGNOSTICS AND OPTIMIZATION MODELS

First of all, Remaining-Useful-Life (RUL) prognostics algorithms need to be further improved to obtain more accurate and realistic maintenance planning.

1. Using future flight schedules and operational conditions.

¹International Maintenance Review Board Policy Board (IMRBPB). *Aircraft Health Monitoring (AHM) integration in MSG-3, IP180*. 2018, pp. 1–33.

The RUL prognostics algorithms proposed in this dissertation only use the past condition data. If the future flight schedules and associated operational conditions are known in advance, RUL prognostics and predictive maintenance planning can be more effective. Especially for commercial flights, where flight schedules are already known for a few weeks in advance, such future operational conditions can be integrated into RUL prognostics and PdAM.

2. Predicting RUL of multi-component systems.

The current RUL prognostics algorithms mostly focus on the RUL of a component, but some components compose a multi-component system. For example, an aircraft has 4 landing gear brakes on each side of the wing, and a minimum of 3 brakes is required. The RUL of this multi-brake system can be predicted by simultaneously considering the RUL of individual brakes, system redundancy, and associated regulations. Considering the RUL of multi-component systems and the RUL of individual components together, PdAM can be more flexible and reliable.

3. Considering additional operational requirements, such as spare part management, maintenance crew schedule, etc.

The fleet-level PdAM of this dissertation focused on flight schedule and hangar availability, but there are also other operational requirements such as spare part management and maintenance crew schedule. Some components, such as aircraft engines, are not always stored in the hangar but are ordered when maintenance is scheduled. The logistics often take a significant time, and might delay the maintenance schedule. Also, the availability of certified maintenance personnel is limited. The maintenance crew schedule and the predictive maintenance plans should be obtained considering each other. Although such operational requirements have been studied under current time-based maintenance, their impact of the new predictive maintenance is rarely investigated.

REMAINING NON-TECHNICAL CHALLENGES

This dissertation focuses on the biggest technical challenges of PdAM, especially the reliability and efficiency of RUL prognostics and PdAM decision support systems. However, for the full implementation of PdAM in practice, several challenges remain.

4. Building the trust of all people involved in the new predictive aircraft maintenance approach.

Chapter 7 suggests that building trust is the second biggest challenge in adopting PdAM in practice. Despite all the technical validations and the optimization models proposed for PdAM, it cannot be successfully adopted if the users do not trust this new approach. While the trust in new technologies has been extensively studied in other fields such as self-driving cars, it has not been investigated in-depth for PdAM. For the final approval of PdAM approaches, it is necessary to build trust, not only technically but also emotionally.

5. Adoption of PdAM from the perspectives of legislation and business.

Since aircraft maintenance is strictly regulated under law, legal supports are necessary to implement PdAM. At the same time, it is a business that should make a profit. Thus, further research is necessary that considers the legislation of PdAM and the business models of PdAM. Of course, technical improvements provide the strongest support for the legislation and the business models of PdAM. In this light, this dissertation serves as a starting point for informed discussions on the future of predictive aircraft maintenance.

CURRICULUM VITÆ

Juseong LEE



Juseong Lee was born on March 23rd, 1991, in Suwon, South Korea. In 2010, he started a Bachelor of Science degree in Aerospace Engineering at Korea Advanced Institute of Science and Technology (KAIST). During his Bachelor's program, he conducted research on "Pareto front generation for mixed-discrete multi-objective optimization," which is funded by KAIST Undergraduate Research Participation Program. Between 2011 and 2013, he was away from the university for two years to serve in the Republic of Korea Army. In 2014, he joined an exchange program at Delft University of Technology (TU Delft), where he was motivated to return later. After all, he obtained his B.Sc. degree (*Magna Cum Laude*). In 2018, he obtained his Master of Science degree in Aerospace Engineering at KAIST. His master thesis is "Design of space logistics mission architecture based on time expanded network flow and modular design." Besides his thesis, he participated in a research program funded by Agency for Defense Development of South Korea, and a sounding rocket design project. He was also awarded in a lunar exploration design competition held by NASA.

In 2018, Mr. Lee came back to TU Delft, and started his position as a Ph.D. candidate at Air Transport and Operations (ATO). His research was funded by the European Union's Horizon 2020 ReMAP project (Real-time Condition-based Maintenance for Adaptive Aircraft Maintenance Planning). He has focused on predictive aircraft maintenance to integrate Remaining-Useful-Life prognostics for more reliable and efficient maintenance planning. His contributions have been published in top journals and presented at several international conferences, leading to two honorable best paper awards. Alongside his own research, he was a teaching assistant for a course, "Stochastic Process and Simulation". He assisted in supervising an MSc research entitled as "Fleet level multi-unit maintenance optimization subject to degradation," and coached a Design Synthesis Exercise, "Design a fully electrical aircraft suitable for island hopping operations."

Apart from academic interests, Mr. Lee enjoys all kinds of cooking including Korean, Chinese, Italian, French, and Dutch cuisines. He also loves exploring new cultures, meeting new people, and trying all the other unfamiliar experiences.

LIST OF PUBLICATIONS

JOURNAL PAPERS:

4. **J. Lee** and M. Mitici, "Deep reinforcement learning for predictive aircraft maintenance using probabilistic Remaining-Useful-Life prognostics," *Reliability Engineering and System Safety*, 2022. DOI: [10.1016/j.res.2022.108908](https://doi.org/10.1016/j.res.2022.108908).
3. **J. Lee** and M. Mitici, "Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling," *Reliability Engineering and System Safety*, vol. 218, p. 108123, 2022. DOI: [10.1016/j.res.2021.108123](https://doi.org/10.1016/j.res.2021.108123).
2. **J. Lee**, M. Mitici, H. A. P. Blom, P. Bieber, and F. Freeman, "Identification of emerging hazards for the data-driven predictive aircraft maintenance process," submitted to *Safety Science* in 2021 (under review).
1. **J. Lee** and M. Mitici, "An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets," *Reliability Engineering and System Safety*, vol. 202, p. 107052, 2020. DOI: [10.1016/j.res.2020.107052](https://doi.org/10.1016/j.res.2020.107052).

PEER-REVIEWED CONFERENCE PAPERS:

4. **J. Lee**, M. Mitici, S. Geng, and M. Yang, "Designing reliable, data-driven maintenance for aircraft systems with applications to the aircraft landing gear brakes," in *Proceedings of the 32nd European Safety and Reliability Conference (ESREL)*, pp.25-32, Dublin, Ireland, August 28 – September 1, 2022. DOI: [10.3850/978-981-18-5183-4_R01-04-288-cd](https://doi.org/10.3850/978-981-18-5183-4_R01-04-288-cd).
3. **J. Lee**, I. de Pater, S. Boekweit, and M. Mitici, "Remaining-Useful-Life prognostics for opportunistic grouping of maintenance of landing gear brakes for a fleet of aircraft," in *Proceedings of the 7th European Conference of the Prognostics and Health Management Society 2022*, pp.278-285, Turin, Italy, July 6–8, 2022. DOI: [10.36001/phme.2022.v7i1.3316](https://doi.org/10.36001/phme.2022.v7i1.3316).
2. **J. Lee** and M. Mitici, "Multi-objective analysis of condition-based aircraft maintenance strategies using discrete event simulation," in *2021 Annual Reliability and Maintainability Symposium (RAMS)*, pp. 1–6, Orlando, FL, USA, May 24–27, 2021. DOI: [10.1109/RAMS48097.2021.9605761](https://doi.org/10.1109/RAMS48097.2021.9605761).
1. **J. Lee** and M. Mitici, "Predictive aircraft maintenance: modeling and analysis using stochastic Petri nets," in *Proceedings of the 31st European Safety and Reliability Conference*, pp. 146-153, Angers, France, September 19–23, 2021. DOI: [10.3850/978-981-18-2016-8_050-cd](https://doi.org/10.3850/978-981-18-2016-8_050-cd).

AWARDS:

3. *Innovation in Transport Applications, European Safety and Reliability Conference* in 2022, supported by PayPal.
2. *Best Paper Award 2nd Prize, European Conference of the Prognostics and Health Management Society* in 2022.
1. *Thomas L. Fagan, Jr., RAMS Student Paper Award 1st Place, Reliability and Maintainability Symposium* in 2021.