THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Mixed-Integer Optimization Modeling for the Simultaneous Scheduling of Component Replacement and Repair

Gabrijela Obradović

CHALMERS

Division of Applied Mathematics and Statistics
Department of Mathematical Sciences
Chalmers University of Technology
Gothenburg, Sweden 2023

Mixed-Integer Optimization Modeling for the Simultaneous Scheduling of Component Replacement and Repair
Gabrijela Obradović
Gothenburg 2023
ISBN 978-91-7905-908-8

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 5374
ISSN0346-718X

Department of Mathematical Sciences
Division of Applied Mathematics and Statistics
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone +46 (0)31 772 1000

Typeset with LaTeX
Printed by Chalmers digital print
Gothenburg, Sweden 2023

The only victory that counts
is the one over yourself.

Jesse Owens, athlete

Mixed-Integer Optimization Modeling for the Simultaneous Scheduling of Component Replacement and Repair

Gabrijela Obradović

Division of Applied Mathematics and Statistics
Department of Mathematical Sciences
Chalmers University of Technology

Abstract

Maintenance is a critical aspect of many industries, playing an indispensable role in ensuring the optimal functionality, reliability, and longevity of various assets, equipment, and infrastructure. For a system to remain operational, maintenance of its components is required, and for the industry to optimize its operations, establishment of good maintenance policies and practices is vital.

This thesis concerns the simultaneous scheduling of preventive maintenance for a fleet of aircraft and their common components along with the maintenance workshop, to which the components are sent for repair. The problem arises from an industrial project with the Swedish aerospace and defence company Saab. In the four papers underlying this thesis, we develop mathematical models based on a mixed-binary linear optimization model of a preventive maintenance scheduling problem with so-called interval costs over a finite and discretized time horizon. We extend this scheduling model with the flow of components through the repair workshop, including stocks of spare components as well as of damaged components to be repaired. The components are modeled either individually, aggregated, or as jobs in the workshop, whose scheduling is considered to be preemptive or non-preemptive. Along with the scheduling, we address and analyze two contracting forms between the stakeholders—the aircraft operator and the repair workshop; namely, an availability of repaired components contract and a repair turn–around time contract of components sent to the repair workshop, leading to a bi-objective optimization problem for each of the two contracting forms. To handle the computational complexity of the problems at hand, we use Lagrangean relaxation and subgradient optimization to find lower bounding functions—in the objective space—of the set of non-dominated solutions, complemented with math-heuristics to identify good feasible solutions. Our modeling enables capturing important properties of the results from the contracting forms and it can be utilized for obtaining a lower limit on the optimal performance of a contracted collaboration between the stakeholders.

Keywords: Mixed-Integer Linear Optimization, Multi-Objective Optimization, Mathematical Modeling, Maintenance Optimization, Workshop Scheduling, Contracting Forms, Simultaneous Scheduling

List of publications

This thesis is based on the work represented by the following papers:

I. **Obradović, G.**, Strömberg, A.-B., Lundberg, K. Scheduling of repair and replacement of individual components in operating systems.
Under review for publication in *Journal of Scheduling* (2023)

II. **Obradović, G.**, Strömberg, A.-B., and Lundberg, K. Simultaneous scheduling of replacement and repair of common components in operating systems. Annals of Operations Research, 322:147–165, (2023), doi: 10.1007/s10479- 022-04739-8.

III. **Obradović, G.**, Strömberg, A.-B., Lundberg, K. An enhanced mathematical model for optimal simultaneous preventive maintenance scheduling and workshop planning.
Submitted to *EURO Journal on Decision Processes* (2023)

IV. **Obradović, G.**, Strömberg, A.-B., , Held F., Lundberg, K. Approximating the Pareto frontier for bi-objective preventive maintenance and workshop scheduling. A Lagrangean lower bounding methodology for evaluating contracting forms.
Manuscript (2023)

Additional papers not included in this thesis:

V. **Obradović, G.**, Strömberg, A.-B, Lundberg, K. Simultaneous scheduling of preventive system maintenance and of the maintenance workshop.
Published in *Proceedings of the PLANs forsknings- och tillämpningskonferens* (2020)

VI. **Obradović, G.**, Strömberg, A.-B, Lundberg, K. Replacement and Repair of Common Components in Systems Subject to Operations Planning.
Published in *Proceedings of the 15th Workshop on Models and Algorithms for Planning and Scheduling Problems, MAPSP* (2022)

Author contributions

I.- VI. I worked on model and method development and wrote the manuscript with input from the discussions with the collaborators. Independently, I worked with implementation, creation of data sets, experiment development and computations in papers I.-III,V.-VI. In Paper IV., I did a major part of implementation, creation of data sets and simulation.

Acknowledgements

My sincere gratitude for the supervision, ideas, discussions and sharing her passion for research goes to **Ann-Brith Strömberg**. Working with you for five years have taught me a lot, about optimization and beyond. Thank you, **Kristian Lundberg**, for industrial collaboration, curiosity and ambition. Thanks to the Swedish Governmental Agency for Innovation Systems (VINNOVA; project number 2017-04879), Chalmers University of Technology, and Saab AB for funding the project.

Thank you all at the **Department of Mathematical Sciences**, for making my work possible and contributing to a pleasant working environment. Special thanks to **Marija Cvijović**, for caring, motivating, being an optimist and always putting things into perspective. **Juan Inda**, **Julia Larsson**, **David Lund**, **Malin Mosquera**, **Ioanna Motschan**: I appreciate all the lunches, fikas, encouragement at times, and laughs we had together. Thank you, **Caroline Granfeldt**, for being a very good colleague and a friend. I would also like to thank the rest of the optimization group: **Michael Patriksson**, **Sunney Fotedar**, **Edvin Åblad** and **Quanjiang Yu**.

I would like to mention the **Department of Mathematics and Informatics** at the **University of Novi Sad** and thank everyone who has taught, encouraged and inspired me during my studies.

PhD life is both nice and challenging in many ways. During these five years, I have matured, developed and experienced a lot, professionally and personally. This time would have never been the same without people in my life, and I want to say thank you all. **Wendy**, **Isra**, **Annaleena**, **Marcello**: for being the friends you are, making Gothenburg feel more like home. **Barbara**, for introducing me to climbing and strong yoga (shouts to Eric's **strong yoga** class in Fysiken for the much needed mental and physical practice), but also for being an inspiration in many ways and making my PhD time much more fun. **Endre**, **Anna**, **Elin**, **Lucia**: for all the practice on the climbing walls and fun times outside of the gym. **Martin**, for all the lunches, conversations and proper sauna experience. **Maryam**, **Ali**, **Pamela**, **Diego**, **Irenka**, **Kristoffer**: for all the good times we had over the years.

Even though I moved cities, and then also countries, a few true friendships remained. Thank you: **Ana**, for being a close friend and the person who always gets me for almost fifteen years, and never letting the distance change that; **Vanja**, for being a caring friend, for all the visits and honest conversations about life; **Uroš**, for your energy and sincere joy about every success in my life; **Branislava**, **Nevena**, **Marijana**, for staying in touch and present in each other lives, in one way or the other.

Family has always been there for me. My parents, **Sandra** and **Zoran**, thank you for your love and support, for the faith you always had in me; My sister, **Magdalena**, despite our differences, you are the person in my life no one could ever replace; **Radmila**, **Boban**, **Milina**, for always wholeheartedly welcoming me when I come back home; **Angela** and **Hans**, for being kind, thoughtful and always cheering for us.

Special thanks to you, **Felix**, for being my supportive, caring and loving partner, for all the adventures and trips, for embracing the good times and going together through the challenging ones. Thank you for giving me a safe space to be who I truly am.

Gothenburg, 2023

Contents

1 Introduction

Development of theories studied in mathematics is often motivated by practical applications. While in pure mathematics, the abstract concepts are studied on their own, the problems that are being studied in applied mathematics usually come from different fields (e.g., physics, engineering, medicine, biology, finance, business, computer science and industry). Thus, applied mathematics is a combination of mathematical science and specialized knowledge. The use of mathematical models and methods to find the best alternative when making a decision belongs to the field of *optimization*[1]. Thereby, optimization is the science of making the best possible decision. The term *best* refers to obtaining the best value of the defined objective and *possible* refers to the restrictions/constraints[2] that we most often have. For example, an objective many people have every morning on their way to work is to minimize the travel time. However, there are restrictions such as traffic, speed and roads that have an impact on the minimization of the travel time. Minimizing the travel time subject to traffic, speed and roads constitutes an *optimization problem.* In order to create a mathematical model, the optimization problem, consisting of one (or multiple) optimization objective(s) and the constraints, has to be expressed in terms of mathematical functions and relations. The model is then solved with either existing or problem-tailored methods and solution approaches.

Optimization is an interdisciplinary field in which, in order to achieve good results in a practical application, one often needs skills and competence in mathematics and computer science, but also problem-specific domain knowledge. There are numerous areas of application of optimization, and some of them include production planning [2, 49], transport and logistics [8, 9, 27], telecommunication (e.g., network design [26]), traffic planning (e.g., traffic signal control [12], infrastructure planning [56]), structural design (e.g., electric-vehicle charging stations [42], robot control [61]), timetabling and staff planning (e.g., shifts

[1]The word *optimum* ("optimus" in Latin) means "best, very good".

[2]A *constraint* is a condition of an optimization problem that the solution must satisfy.

in a school [44]), portfolio optimization [38]. Some of various applications within my research group come from energy [34, 73], automotive [75] and aerospace [29, 64] industry.

The application presented in this thesis comes from aerospace industry and concerns maintenance of aircraft components. In order for an aircraft (or any system that operates in a similar fashion) to perform well and remain operational, maintenance is required. Different systems require maintenance at different frequencies. For example, an offshore windmill [65] does not require as frequent maintenance as compared to a commercial airplane [63]. Besides frequencies at which maintenance is performed, we differentiate between types of maintenance as well as costs that come with different maintenance activities. When planning maintenance for any system [70, Ch. 3], the decisions to be made concern when each of its components should be maintained (i.e., replaced, repaired, or serviced) and what kind of maintenance should then be performed, with respect to the operational schedule of the system. A good maintenance plan increases the operational readiness and minimizes the downtime of a system. *Preventive maintenance* (PM) [66], performed in order to avoid failure, can often be planned well in advance, while *corrective maintenance* (CM) [15], performed after failure has occurred in order to restore the system into an operational state, may come on very short notice. While both PM and CM aim at restoring the components in order to put the system back in an operational state, CM is often much more costly than PM, due to a longer system down-time, (possibly) short notice but also due to possible damages to other components caused by the failure. On the other hand, an unexpected but necessary CM action may provide an opportunity for the PM at which the maintenance actions can be rescheduled, starting from the system's current state. Another strategy for planning maintenance activities is so called *opportunistic maintenance* (OM), in which a mathematical model is utilized to decide whether, at a (possibly already planned) maintenance occasion, more than the necessary maintenance activities should be performed [3].

Maintenance planning is sometimes given insufficient attention. Investing in maintenance planning early on can save a lot of costs (in terms of replacements, repairs and renewals). Hidden costs are often even more important to account for (e.g., due to an unexpected failure, the system cannot operate as planned—which is often quite costly) [17]. Effective maintenance planning requires the determination of a measurably good or, ideally, the best plan, which can be achieved by using optimization frameworks and tools. *Maintenance optimization* means deciding which maintenance activities to perform, and when, such that one or several objectives are optimized. Models developed for such tasks are extensively studied in the literature (see the surveys [20, 21, 28, 50]) and have impact on both cost and efficiency of the maintenance actions.

Aims

The main goal of this thesis is modeling of the integrated simultaneous scheduling of the preventive maintenance of the aircraft and of the maintenance workshop. We consider two stakeholders and two types of contracts between them. The model development presented in this thesis was motivated by seeking improvement in two areas: modeling (grasping the problem structure and incorporating a higher degree of specificity) and efficiency (striving for models and solution approaches for obtaining solutions in reasonable computing time). The problems we address are complex, hence the goal is to construct the best mathematical models for these problems as well as an efficient solution approaches.

Outline

The outline of this thesis is as follows. In Chapter 2, we describe the use case given by Saab, which provided the starting point for determining the framework. The mathematical optimization background required to understand the work presented in the thesis is given in Chapter 3. Then, the mathematical modeling of the problem (including decision variables, optimization constraints and objective functions), complexity analysis and methodology are given in Chapter 4. we summarize the appended papers in Chapter 5, and give the main conclusions and future research questions in Chapter 6.

2 Aircraft maintenance

In this chapter, we describe and motivate the industrial problem studied in this thesis, as well as discuss limitations encountered on the way.

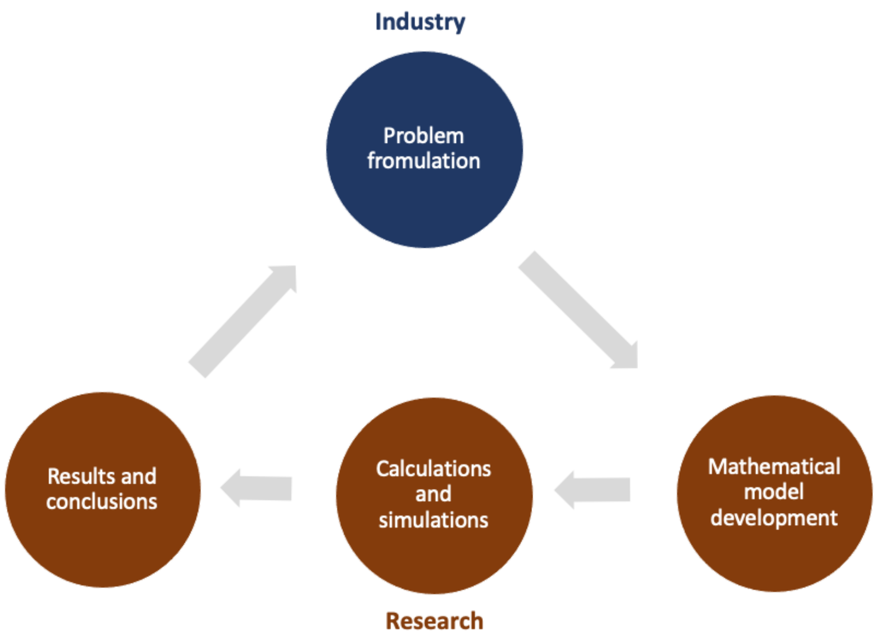

Figure 2.1: Work flow. Collaboration between academia and industry requires identification, understanding and definition of the problem, followed by development of mathematical modeling used to describe it. The next steps are experiment design, choice of data and implementation. After obtaining (any) results, model is likely to be revisited and adjusted couple of times until some significant conclusions and findings can be inferred.

2.1 Problem description

We present an application from the aerospace industry, in collaboration with a Swedish aerospace and defence company Saab. On one side, we consider a system of aircraft that has an operational demand to fulfill, and on the other, the maintenance workshop (Saab) that repairs the components coming from the aircraft and makes them available for usage again. Hence, there are two stakeholders, an *aircraft operator* and a *maintenance workshop* (i.e., maintenance supplier), whose collaboration is normally predefined by a contract. We define and discuss a number of optimization objectives corresponding to two different contract types, so-called *availability* and *turn–around time* contracts. In

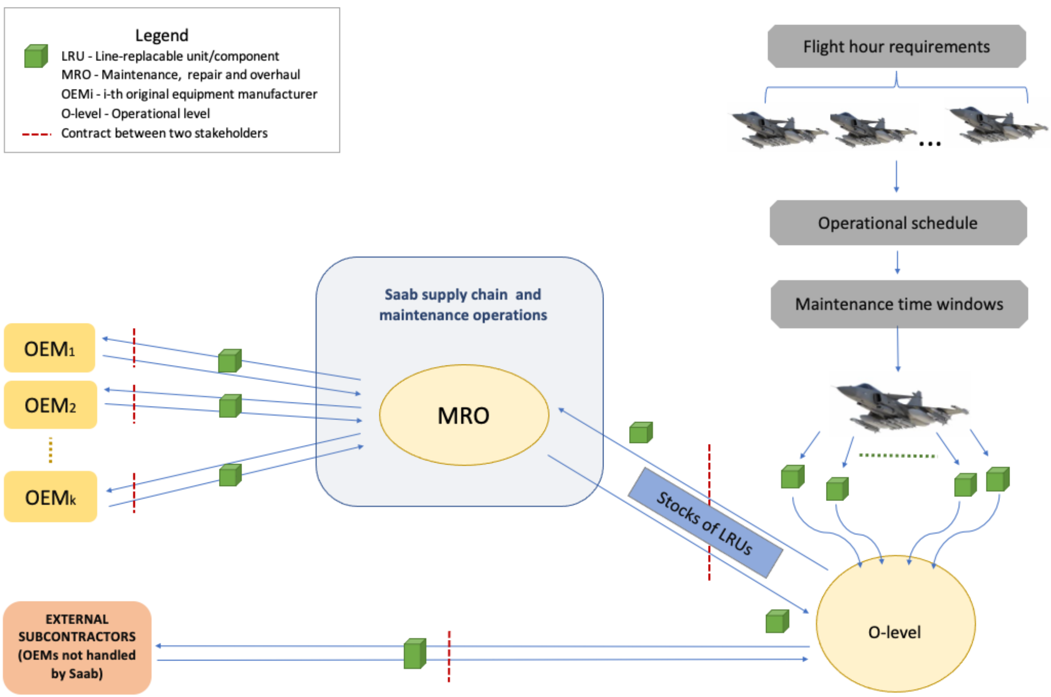

Figure 2.2: Illustration of the problem studied. The problem provided by Saab is one part of a larger supply chain, including external manufacturers and subcontractors, maintenance workshop, operational level, stocks of components and operational scheduling.

Figure 2.2, we illustrate the system–of–systems governing repair and replacement of components from an aircraft. The Swedish Air Force is assigned a flight hour requirement to be distributed among the fleet of aircraft, which defines a *flight assignment* problem [30, 72]. After an operational schedule is made, each aircraft is assigned to a timetable specifying when it is scheduled to fly. Since maintenance can be performed only when the aircraft is grounded, time windows of opportunities for doing maintenance are generated based on the operational timetables. Maintenance scheduling is done on the *operational level* (O–level in Fig. 2.2) where each component to be repaired is replaced with a new (or as good as new) component of the same component type. The component to be repaired goes to the *maintenance workshop* (MRO – Maintenance, Repair and Overhaul in Fig. 2.2), where it is to be scheduled for repair. The maintenance workshop is governed by Saab but there are also *original equipment manufacturers* (OEM_i in Fig. 2.2) and *external subcontractors*. Components can be repaired in the maintenance workshop governed by Saab but they can also be sent further, to one of the OEMs or external subcontractors outside of Saab's supply chain and maintenance operations. Joint activities between any two stakeholders are typically governed by a contract.

2.2 Motivation

The research presented in this thesis was motivated by a specific real world problem. However, the application of this work goes beyond this specific problem. Any system that performs some sort of operations and undergoes maintenance can be considered in our modeling.

Maintenance is a critical aspect of many industries, playing an indispensable role in ensuring the optimal functionality, reliability, and longevity of various assets, equipment, and infrastructure. Some of numerous examples are railway and air traffic, commercial heavy vehicles, manufacturing machines in industry, energy production and automotive industry [6, 11, 24, 53, 57, 68]. Performing maintenance operations in a good fashion is of high importance because it has implications on productivity, efficiency, safety, and cost-effectiveness within industrial operations. Many companies and factories operate in reactive mode, which means that a problem (e.g., a component failure) is addressed and dealt with only once it occurs. Maintenance costs represent, on average, a significant portion of the total operating budget, varying from a few percent in lighter manufacturing to a high percentage in equipment-intensive industries. Frequently, maintenance costs are underestimated due to the fact that the hidden costs are not accounted for. Ineffective maintenance management policies lead to large increases in costs and most importantly, decreased efficiency.

We give a few applications of scheduling optimization from [4, Ch. 1]. Scientist collaborating with United Airlines in 1986 considered their crew scheduling problem. The savings reported form the implementation of the results of the project was 6 million US dollars per year. A planning optimization regarding bus scheduling in Berlin resulted in reduction from the use of 1800 to 1200 buses, without any loss of quality of service. A group of operations research scientist collaborating with the San Francisco police department in 1989 developed a tool based on a heuristic solution of the staff planning and police vehicle allocation problem; it has been reported that it gave a 20% faster planning and savings in order of 11 million US dollars per year. Hence, smart scheduling can provide with significant reduction of cost while giving good, often better solutions for the problem at hand.

Advanced optimization models have been developed for each part of the supply chain of aircraft maintenance—from tactical scheduling of aircraft to missions or maintenance [30, 47, 59, 67], flight assignment [30, 72], to depot level maintenance planning and scheduling [7, 25, 39]. Even though there exists an interdependent relationship between production scheduling and maintenance planning, the two are mostly planned and executed separately, both in literature and in reality. Most of the time, there is lack in communication between the maintenance planning and the production scheduling side [71], resulting in unmet demand and/or supply on either side, implying lower efficiency and higher costs. An integrated optimization problem of non-permutation flow-shop[1] scheduling and maintenance planning with variable processing speed is studied in [36]. A heuristic approach for maintenance scheduling for a military aircraft fleet under limited maintenance capacities was proposed in [74].

Generally speaking, stakeholders collaborate based on a contractual agreement and the level of transparency of their collaboration varies. If the organizations and the information they work with are fully transparent, and the decisions are taken simultaneously for all stakeholders, that is regarded as a tightly integrated collaboration. A systematic review and meta-analysis on the value of integrated planning for production, inventory, and routing decisions was presented in [35], estimating an expected cost savings provided by integration of 11.08% with a 95% confidence interval of $[6.58\%, 15.58\%]$. According to [14], the integrated optimization of production scheduling and maintenance planning in the capital goods industry can reduce up to 63.5% of the total cost by comparing with the existing company's scheduling. The motivation for considering a tight integration of the maintenance planning for the systems and the production scheduling of the maintenance workshop is threefold. First, a tight integration provides a planning tool for the systems in which the maintenance workshop is, in reality, integrated with the operating system. In this case, the stakeholder operating the aircraft is also responsible for and performs maintenance of its components. Secondly, when there is more than one stakeholder, a tightly integrated model formulation will provide an optimistic estimate of the results—in terms of costs

[1] *The Non-Permutation Flow-Shop scheduling problem* (NPFS) [58] is a generalization of the traditional *Permutation Flow-Shop scheduling problem* (PFS) that allows changes in the job order on different machines.

for maintenance, of costs for lateness (under a turn-around time contract), or of the lower limit of items on the stock and/or the average availability (under an availability contract)—that could be obtained in reality and which can be used as a benchmark. Lastly, the integration enables an investigation and a comparison of different types of contracts that can be set up between the stakeholders.

2.3 Limitations

Mathematical modeling is a powerful tool that can be utilized to represent the reality, and the trade-off between model simplicity and inclusion of the nuances of the real world is an integral part of it. Typically, a model is a simplification or an abstraction of reality, tailored to the particular problem at hand, and the level of detail taken into account varies. Assumptions and simplifications are carefully introduced so that the main properties and structures of the problem are kept.

Focus of this work is preventive maintenance modeling, which enables minimization of risks of unexpected failures, but it does not eliminate them. It may still happen that a component breaks or stops functioning unexpectedly, when there is no scheduled preventive maintenance event. Our modeling can address this situation in two ways. First, once an unexpected failure occurs, one can re-plan from that point in time while leaving the part of the schedule in the near future unchanged (to avoid significant disturbances). Since short-term changes in the operational schedules for the systems, as well as in the schedules for the maintenance workshop, are often inconvenient and sometimes not even feasible, the rescheduling should (if possible) be such that the solution remains fixed for a certain number of time steps. Secondly, by keeping up the level of available components, we ensure that once a component needs to be replaced, there will be no long waiting time. Similarly, by ensuring aircraft availability, we may replace an aircraft in need of maintenance with another aircraft, whence the disturbance in the operational schedule would be minimized. However, this is not sufficient in order to account for the possible unexpected events and to fully replace the corrective maintenance planning.

Saab is a defense company, and the data is classified and confidential. Thus, real data could not be used for our research, and results are based on simulated data, which makes it harder to make exact and possibly realistic conclusions about (some parts of) the results. There exists a large number of components circulating in between the aircraft and the maintenance workshop. Instead of modeling all component types, we focus on the most important, the *safety* *critical components*[2] A similar approach is employed in, e.g., nuclear power plant maintenance modeling [19]. The total number of components considered could, in reality, be larger; the length of the planning horizon depends heavily on the length of each time step (e.g., one hour, one half-day, or one day), which is a matter of definition. However, the use of our model in different applications, will result in different instance sizes. For example, rail traffic or commercial airlines instances would have a larger number of systems (i.e., train sets, aircraft). In addition, we do not model the parameters that are judged not to be in the scope of our research (e.g., the cost of maintaining a component of a certain type is fixed and its effect on the solution in the tight integration can thus be neglected). If required, these parameters could be included in the modeling. Nonetheless, the primary research interest in the work presented in this thesis is a general maintenance planning problem with simulated data that may be utilized not only for the specific use case, but for related applications as well.

The choice of the length of a time step is not clearly specified in the thesis. The reason is that this length should be chosen depending on the length of the planning period. Hence, each time step may represent a half-day, a day, or a week, while the planning period may represent one month, one year, or several years.

Stochasticity in the every day operations in the maintenance workshop is hard to model. While we introduce ways for overcoming some level of uncertainty, we can neither eliminate nor fully account for it. Besides this, parallel activities in the workshop that are limited by personnel, test equipment and other facility artifacts, are modeled as parallel repair lines, which indeed leads to a simplification of the actual operations in the maintenance workshop.

[2] A component is called *safety critical* if its failure could lead to a breakdown, possibly with catastrophic consequences (e.g., the engine or the ejection seat).

3 Mathematical modeling toolbox

A general optimization problem can be formulated as

$$\text{minimize} \quad f(\boldsymbol{x}), \tag{3.1a}$$

$$\text{such that} \quad \boldsymbol{x} \in \mathcal{X}, \tag{3.1b}$$

where $f : \mathbb{Z}^{n_1} \times \mathbb{R}^{n_2} \mapsto \mathbb{R}$ is an *objective function*, the *decision variables* are denoted by $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ and $n = n_1 + n_2$. The set $\mathcal{X} \subset \mathbb{Z}^{n_1} \times \mathbb{R}^{n_2}$ defines the *feasible solutions* to the problem and usually has the form $\mathcal{X} := \{\mathbf{x} \in \mathbb{Z}^{n_1} \times \mathbb{R}^{n_2} : g_i(\boldsymbol{x}) \leq b_i,\ i = 1, \ldots, m\}$, where $g_1, \ldots, g_m$ are *functions* and $b_1, \ldots, b_m$ are given *parameters*. Depending on how the functions $g_1, \ldots, g_m$ are specified and which assumptions are made regarding feasible values on the variables, we obtain different problem classes, such as linear optimization (LP), non-linear optimization (NLP), mixed-integer linear programming (MILP) or integer linear optimization (ILP).

This chapter presents a background for the mathematical modeling and optimization methods used within the thesis.

3.1 Mixed-integer linear programming (MILP)

A *mixed-integer linear program* is an optimization problem with affine/linear objective, constraint functions and integral requirements on some of the variables.

Every MILP problem can be expressed as

$$
\begin{aligned}
z^* := \text{minimum} \quad & \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}, && (3.2\text{a})\\
\text{such that} \quad & A\boldsymbol{x} \geq \boldsymbol{b}, && (3.2\text{b})\\
& \boldsymbol{x} \in \mathbb{Z}^{n_1} \times \mathbb{R}^{n_2}, && (3.2\text{c})
\end{aligned}
$$

where $n = n_1 + n_2$ is the dimension of the variable space, m is the number of inequality constraints, A is an $m \times n$ matrix and $\boldsymbol{b}$ and $\boldsymbol{c}$ are vectors. MILP problems are NP-hard (e.g., [18, Section 1.3.3]), which means that the time to solve the model (in the worst case) is exponential as a function of the instance size (i.e., number of variables and constraints).

A classical example of a problem that can be formulated as a MILP is the *travelling salesperson problem* (TSP). Let $G = (\mathcal{V}, \mathcal{A})$ be a directed graph, where the nodes $v \in \mathcal{V}$ represent the cities and arcs $a \in \mathcal{A}$ represent the roads. There is a traveling time cost c_a associated with every arc a. If the traveling cost from city i to city j is equal to the cost from j to i, the problem is called *symmetric* TSP. Otherwise, it is asymmetric. TSP can be modeled as MILP and one of the formulations given by [46], is expressed as to

$$
\begin{aligned}
\text{minimize} \quad & \sum_{a \in \mathcal{A}} c_a x_a, && && (3.3\text{a})\\
\text{such that} \quad & \sum_{a \in \delta^+(i)} x_a = 1, && i \in \mathcal{V}, && (3.3\text{b})\\
& \sum_{a \in \delta^-(i)} x_a = 1, && i \in \mathcal{V}, && (3.3\text{c})\\
& u_i - u_j + (n-1)x_{(ij)} \leq n-2, && (ij) \in \mathcal{A} \text{ s.t. } i, j \neq s, && (3.3\text{d})\\
& u_i \in [1, n-1], && i \in \mathcal{V} \setminus \{s\}, && (3.3\text{e})\\
& \boldsymbol{x} \in \{0,1\}^m. && && (3.3\text{f})
\end{aligned}
$$

The (integer) decision variables u_i denote the order in which the nodes (cities) are being visited while the (binary) decision variables x_a take value 1 if arc a is used, otherwise 0. The constraints (3.3b) and (3.3c) ensure that each node has one entering and one leaving arc, where $\delta^-(i)$ and $\delta^+(i)$ denote the set of arcs entering and leaving node i, respectively. One way to prevent subtours, i.e., to ensure that the solution admits only one connected tour and not multiple disjoint tours, is to use the constraints (3.3d).

3.2 Multi-objective optimization

> "We will say that the members of a collectivity enjoy maximum ophelimity[1] in a certain position when it is impossible to find a way of moving from that position very slightly in such a manner that the ophelimity enjoyed by each of the individuals of that collectivity increases or decreases. That is to say, any small displacement in departing from that position necessarily has the effect of increasing the ophelimity which certain individuals enjoy, and decreasing that which others enjoy, of being agreeable to some and disagreeable to others." *Vilfredo Pareto* (1896)

Multi-objective optimization is a type of vector optimization that has been applied in many fields of science, where optimal decisions need to be taken in the presence of trade-offs between two or more conflicting objectives. In multi-objective mathematical programming[2], there are more than one objective function and in most cases, there is no single optimal solution that optimizes all objective functions simultaneously. Then, it is the *decision maker* who chooses the most preferred solution. Consider the optimization problem to

$$\begin{aligned} &\text{minimize} && \{f_1(\boldsymbol{x}), \ldots, f_K(\boldsymbol{x})\} && (3.4a)\\ &\text{such that} && \boldsymbol{x} \in \mathcal{X} && (3.4b) \end{aligned}$$

where $K \geq 2$ is the number of possibly conflicting objective functions $f_k : \mathbb{R}^m \to \mathbb{R}$, $k = 1, \ldots, K$, that are to be optimized simultaneously. The optimization problem (3.4) is a so-called *multi-objective optimization problem* [22]. If there exists a solution that is optimal with respect to all K objectives, that is a trivial case, since there is no conflict between objectives in that case. We assume that solutions that are optimal with respect to all objectives do not exist in the model (3.4).

Naturally, the question of defining optimality for multi-objective problems arises. For that, we define *Pareto optimality* (see, e.g., [43]).

A point $\boldsymbol{x}^* \in \mathcal{X}$ is *Pareto optimal* in the multi-objective optimization problem (3.4) if and only if there does *not* exist any point $\boldsymbol{x} \in \mathcal{X}$ such that $f_k(\boldsymbol{x}) \leq f_k(\boldsymbol{x}^*)$, $k \in \{1, \ldots, K\}$, and $f_\ell(\boldsymbol{x}) < f_\ell(\boldsymbol{x}^*)$ for at least one $\ell \in \{1, \ldots, K\}$. All Pareto

[1] *Ophelimity* is an economic concept introduced by Vilfredo Pareto as a measure of purely economic satisfaction, so he could use the already well-established term *utility* as a measure of a more broadly based satisfaction encompassing other dimensions as well, such as the ethical, moral, religious, and political.

[2] Also known as *multi-objective programming, multicriteria optimization, multiattribute optimization* or *Pareto optimization.*

optimal points (possibly an infinite number) constitute a *Pareto optimal set* or *Pareto front*, and they all lie on the boundary of the feasible criterion space $\mathcal{X}$. There are many ways of exploring a Pareto front (see [45]). The most common one is to solve single objective problems created from the multi-objective problem through (some sort of) scalarization procedure (e.g., the weighted sum method or the ε-constraint method; see [23]). Since all solutions on the Pareto front are equally good, it is the decision maker who is required to choose one out of the set of all Pareto optimal solutions. In practice, it is not always the case that all solutions on the Pareto front are computed, so the decision maker may choose one of the solutions computed.

3.2.1 Scalarization methods

One way to solve a multi-objective optimization problem is to transform the multi-objective problem to a (parameterized) single objective problem and solve it repeatedly with different parameter values, as illustrated in Fig. 3.1. Scalarization methods fulfill some properties that are desirable when solving a multi-objective problem [37], including that: optimal solutions are (weakly) efficient, solving a scalarized problem is not harder than single objective version of problem (both in theory and in practice), all efficient solutions can be found and scalarization has linear formulation.

A commonly used scalarization methods for computing Pareto optimal points is the *ε-constraint method*, where one of the objectives is optimized while the other objectives are turned into constraints and expressed as to

$$\text{minimize} \quad f_j(\boldsymbol{x}), \tag{3.5a}$$
$$\text{such that} \quad f_k(\boldsymbol{x}) \leq \varepsilon_k, \quad k \in \{1, \ldots, K\} \setminus \{j\} \tag{3.5b}$$
$$\boldsymbol{x} \in \mathcal{X}. \tag{3.5c}$$

By parametric variation in the RHS of the constrained objective functions (ε_k), the efficient solutions of the problem are obtained. Results about the method can be found at [13]. Since the upper bound constraints on objective values, as expressed in (3.5b), are knapsack constraints[3], the problem (3.5) is usually an NP-hard problem [18, Ch. 1.3], which implies that solving each independent scalarized problem (and we might have a fair share of them) may be computationally expensive.

[3] Given a set of items, each with a weight and a value, determine the number of each item to include in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible.

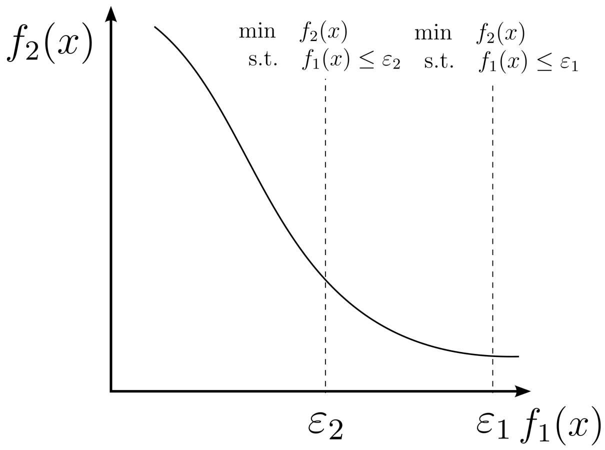

Figure 3.1: ε-constraint method for a (linear) bi-objective optimization problem. One objective is optimized while the other one is constrained by the values of ε. The distribution and number of ε values is a matter of choice. If we consider the range in between the smallest possible ($\varepsilon_{\min}$) and the largest possible ($\varepsilon_1 \geq \varepsilon_{\min}$) value of ε, then we update the $\varepsilon_s := \varepsilon_{s-1} - \frac{1}{n}(\varepsilon_1 - \varepsilon_{\min})$, until $\varepsilon_s \leq \varepsilon_k$, where s is the index of the current ε, and n is the number of discrete epsilon values used.

3.3 Job and machine scheduling

Scheduling is the action of assigning resources (processors, network links or expansion cards) to perform tasks/jobs (threads, processes or data flows). The general *job scheduling problem* involves allocation of n jobs $(j_1, j_2, \ldots, j_n)$ with different processing times to m machines with varying processing power. The objective is to minimize the *makespan*, which is the total duration required to complete all the jobs. In the case of *job-shop scheduling*, each job comprises a set of operations $(o_1, o_2, ..., o_n)$ that must be executed in a predetermined order, taking into account the precedence constraints[4]. Every operation has a designated machine for its processing, and at any given time, only one operation of a job can be processed. If each operation can be processed on any machine, where the machines are considered to be identical, the problem is identified as a *flexible job-shop* problem. *Identical-machines scheduling* is an optimization problem in computer science and operations research. We are given n jobs $(j_1, j_2, \ldots, j_n)$ of varying processing times, which need to be scheduled on m identical machines, such that a certain objective function is optimized (e.g.,

[4]A *precedence constraint* is a relationship between two work orders that indicates when one task begins or ends in relation to another.

minimization of the makespan). This class of problems is a special case of optimal job scheduling. For an overview of scheduling theory see [5, 54].

In scheduling, *preemption* is a technique that allows for the interruption of a running job or operation. One motivation for enabling preemption is the need to switch to a higher-priority task on occasion. Thereby, preemption is particularly useful in situations where jobs have varying priorities or where there is uncertainty in the processing time of jobs. With preemption, a running job can be paused and resumed later, allowing the processor, or machine, to work on a higher-priority job in the meantime [1]. Preemption can improve the performance of scheduling algorithms by allowing them to react quickly to changing conditions and prioritize important jobs. However, preemption can also introduce additional overhead due to the need to save and restore the state of a running job, and can lead to increased complexity in scheduling algorithms. A *non-preemptive* schedule allows no interruptions in job processing. Non-preemption is, however, more suitable for applications in which interruptions in job processing are not allowed. Moreover, non-preemption is a more realistic approach to modeling in some application as interruptions are often not to be planned for.

3.4 Lagrangean relaxation

Generally speaking, a *relaxation* of a problem is an approximation of the problem by a nearby problem that is easier to solve. A relaxation of a problem can be utilized to provide bounds on optimal solutions and define good staring points for heuristic searches for good feasible solutions. Main incentive for utilizing a problem relaxation is that the relaxed problem is typically less computationally demanding as compared to the original, non-relaxed problem. However, solutions to the relaxation are often infeasible in the non-relaxed problem; whence, (heuristic) procedures are used to construct or create a solution that is feasible (and often non-optimal) in the non-relaxed problem.

When we face a large structured problem (e.g., integer programming[5] problems), it can be difficult or impossible to solve the problem in a reasonable time. *Lagrangean relaxation* is a solution strategy used for solving such problems; see [31]. The main idea to is relax some constraints, typically the ones that complicate the problem (i.e., make it more difficult to solve) and take them into account implicitly through the objective in the Lagrangean function. The relaxed constraints do not have to be fulfilled any longer but any violation of

[5]The word "programming" originates from the English word *program* and also means planning - meaning that is more used in the optimization language.

them will be penalized with the set of *Lagrangean multipliers*[6]. Typically, after relaxation of the "complicating" constraints, the original problem separates into smaller and more easily solvable problems. Let us consider a linear optimization problem

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{c}^\top \boldsymbol{x} \qquad , \\ \text{subject to} \quad & \boldsymbol{g}(\boldsymbol{x}) \leq \mathbf{0}^m, \\ & \boldsymbol{x} \in \boldsymbol{\mathcal{X}}, \end{aligned}$$

where $\boldsymbol{c} \in \mathbb{R}^n, \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{g} : \mathbb{R}^n \to \mathbb{R}^m, \mathcal{X} \subset \mathbb{R}^n, m, n \in \mathbb{Z}_+$, and there exists a feasible solution (i.e., $\{\boldsymbol{x} \in \boldsymbol{X} \mid \boldsymbol{g}(\boldsymbol{x}) \leq \mathbf{0}^m\} \neq 0$). The *Lagrangean function* $L : \mathbb{R}^{m+n} \to \mathbb{R}$ is then $L(\mathbf{x}, \boldsymbol{\lambda}) := \mathbf{c}^\mathsf{T}\mathbf{x} + \boldsymbol{\lambda}^\mathsf{T}\mathbf{g}(\mathbf{x})$ and the *Lagrangean dual* problem is defined as

$$\underset{\boldsymbol{\lambda} \geq \mathbf{0}}{\text{maximize}} \qquad h(\boldsymbol{\lambda}),$$

where

$$h(\boldsymbol{\lambda}) := \min_{\boldsymbol{x} \in \boldsymbol{\mathcal{X}}} \quad L(\boldsymbol{x}, \boldsymbol{\lambda})$$

and $h : \mathbb{R}^m \to \mathbb{R}$ is the *Lagrangean dual function.* For some $\boldsymbol{\lambda} \geq \mathbf{0}^m$, the problem of minimizing the Lagrangean function L over $\boldsymbol{x}$ is refered to as a *subproblem.*

The concept of Lagrangean relaxation is based on the duality theory for non-linear problems, known as *Lagrangean duality.* Lagrangean duality holds and is used for non-convex problems (e.g., integer programming problems) and uses the relationship between the primal and the Lagrangean dual problem in order to find the optimal solution.

3.5 Subgradient algorithm

The *subgradient algorithm* is often applied, in combination with Lagrangean duality, to solve optimization problems. The main idea is to relax the complicating constraints and then find optimal (or near-optimal) solutions that will satisfy the relaxed constraints as well.

In order to define and understand the subgradient algorithm, we start by

[6] Dual variables, penalties.

introducing the definition of a subgradient[7]. Furhter, we discuss step length and subgradient direction.

Definition. A vector $\boldsymbol{\gamma} \in \mathbb{R}^n$ is a *subgradient* of a concave function h at $\overline{\boldsymbol{\lambda}} \in \mathbb{R}^n$ if the inequality

$$h(\boldsymbol{\lambda}) \leq h(\overline{\boldsymbol{\lambda}}) + \boldsymbol{\gamma}^{\mathrm{T}}(\boldsymbol{\lambda} - \overline{\boldsymbol{\lambda}})$$

holds for all $\boldsymbol{\lambda} \in \mathbb{R}^n$. The set of subgradients of h at $\overline{\boldsymbol{\lambda}}$ is called the *subdifferential*, denoted by $\partial h(\overline{\boldsymbol{\lambda}})$.

Geometrically, a subgradient is a vector defining a supporting hyperplane to the epigraph of the function h containing the point $\overline{\boldsymbol{\lambda}}$. The subgradient algorithm is defined in Algorithm 1.

Algorithm 1 Subgradient Algorithm

1: Let $k := 0$ and initialize $\boldsymbol{\lambda}^0 \in \Lambda \geq 0$, $h^0_{\text{best}} := h(\boldsymbol{\lambda}^0)$ where Λ is a feasible set for the multipliers $\boldsymbol{\lambda}$;
2: **repeat**
3: Solve the Lagrangean subproblem for $\boldsymbol{\lambda}^k$ and calculate a lower bound $h(\boldsymbol{\lambda}^k)$ on the optimal value;
4: Calculate a subgradient direction $\boldsymbol{\gamma}^k := \boldsymbol{g}(\boldsymbol{x}(\boldsymbol{\lambda}^k)$, step length $\phi^k > 0$ and update $\boldsymbol{\lambda}^{k+1}$:

$$\boldsymbol{\lambda}^{k+\frac{1}{2}} := \boldsymbol{\lambda}^k + \phi^k \boldsymbol{\gamma}^k, \tag{3.6a}$$

$$\boldsymbol{\lambda}^{k+1} := \operatorname{proj}_{\Lambda}(\boldsymbol{\lambda}^{k+\frac{1}{2}}), \tag{3.6b}$$

and update the best bound found so far: $h_{\text{best}} := \max\{h^k_{\text{best}}, h(\boldsymbol{\lambda}^{k+1})\}$;
5: **until** [a termination criteria is fulfilled, let $k := k + 1$].

The step lengths ϕ^k are chosen based on some rule which guarantees convergence. A commonly used step length rule was defined by B. Polyak in [55], and the step length is computed as

$$\phi^k := \frac{\theta^k(h^* - h(\lambda^k))}{||\gamma^k||^2},$$

where θ^k is a scaling parameter for the step length and h^* is a dual upper bound. To ensure theoretical convergence to an optimal solution to the dual problem, it has to hold that $0 < \xi_1 \leq \theta^k \leq 2 - \xi_2 < 2$, $k = 0, 1, 2, \ldots$, where ξ_1 and ξ_2 are positive limits for the scaling parameter θ. One challenge with this step length

[7] In mathematics, the *subderivative*, *subgradient*, and *subdifferential* generalize the derivative to convex functions which are not necessarily differentiable

rule is that h^* is not always known, in which case we may use an upper bound $\overline{h} \geq h^*$ to achieve finite convergence to an ε-optimal solution, where we define ε-optimal as $h(\boldsymbol{\lambda}^k) \geq h^* - \varepsilon$, for any $\varepsilon \geq 0$ (see Theorem 4 in [55]).

There are different ways of updating the value of the scaling parameter θ^k. An adaptive step length update, that has been shown to give fast convergence to an optimal solution in practice, is presented by [10]. The value of the parameter is updated every p number of subgradient iterations. In summary, the best and worst lower bounds found during the last p iterations are compared. If their difference in absolute value is more than 10% of the absolute value of the worst lower bound, implying that too large steps are taken by the algorithm, then the scaling parameter is halved. If their difference is less than 1% of the absolute value of the worst lower bound, the step length can be increased and the scaling parameter is multiplied by $\frac{3}{2}$. If neither of the two cases above holds true, then θ^k is kept unchanged. The percentages and value of the p parameter are adjusted, depending on the algorithm in case, and it is (in practice) no longer required that $\theta^k \in (0, 2)$.

An alternative rule for step length computation is the divergent series step length rule [4, p.181], and it requires that

$$\begin{aligned} \phi^k &> 0, \quad k = 1, 2, \ldots, \\ \lim_{k \to \infty} \phi^k &= 0, \\ \sum_{k=0}^{\infty} \phi^k &= +\infty. \end{aligned}$$

As a result, the subgradient algorithm yields a sequence $\boldsymbol{x}(\boldsymbol{\lambda}^k)$ of solutions to the Lagrangean subproblem. The sequence of solutions typically is not feasible in the original primal problem as the solutions from the sequence will not satisfy the relaxed constraints. Hence, convergence to the optimal primal solutions is not ensured. When that is the case, *ergodic sequences of subproblem solutions* may be utilized. The general idea is creation of approximations of primal solutions by averaging the solutions from the subproblems [76]. This idea was enhanced in [33], where more information is exploited from later subproblem solutions than from earlier ones. The ergodic sequence is shown to converge to an optimal primal solution when the convexity weights are appropriately chosen.

3.6 Complexity classes

Complexity theory is used to determine how long time it takes to solve certain classes of problems. The *algorithmic complexity* provides insights into the relationship between the computational time of an algorithm and the size of the problem the algorithm is used for. We are interested in estimating how the computational time changes as the problem size increases. The concept of *problem complexity*, on the other hand, facilitates the determination of the degree of difficulty associated with solving different problem classes. When

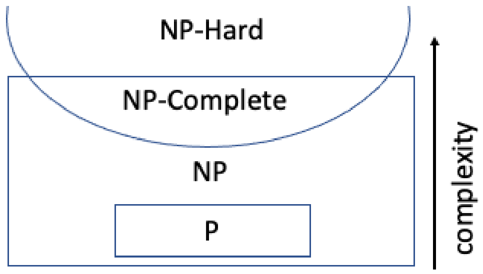

Figure 3.2: Diagram of intersection among classes P, NP, NP-complete and NP-hard problems. The least complex class of problems is P while the most complex one is NP-hard/complete.

we analyze and classify optimization problems with respect to their problem complexity, we study a transformation of the problem, called a *decision problem*. A decision problem is formulated such that the answer to the problem is always either *yes* or *no*. In general, an optimization problem is not harder to solve than its corresponding decision problem.

An algorithm is said to be of *polynomial time* if its running time is upper bounded by a polynomial expression in the size of the input for the algorithm. P is a complexity class that includes the set of all optimization problems whose corresponding decision problems can be solved in polynomial time. That is, given an instance of the problem, the answer yes or no can be decided in polynomial time. These problems are usually refereed to as "easy" problems.

A larger class of problems, including the class P, is *non-deterministic polynomial*, denoted by NP[8]. It is a complexity class that represents the set of all decision problems with the property that for each given solution and corresponding yes answer, there exists a polynomial algorithm that can be used to verify that the yes answer is correct. A decision problem is *NP-hard* if any NP problem can be reduced to it in polynomial time. A decision problem is *NP-complete* if it is in

[8]The *P versus NP problem* is a major unsolved problem in theoretical computer science.

NP and it is NP-hard [18, Ch. 1.3]. These problems are typically difficult to solve.

However, P does not always mean "easy", the same was NP does not always mean "hard". A theoretical polynomial algorithm may have extremely large constant factors or exponents, thus rendering it impractical. Moreover, even if a problem is shown to be NP-complete, there may be effective approaches to solving the problem in practice. Algorithms for some NP-complete problems (e.g., the knapsack problem) have been developed that can solve to optimality many real-world instances in reasonable time.

4 Model and method development

In Chapter 2, we presented the large scale system–of–systems (see Fig. 2.2, Ch. 1) and a subset of it, that is subject of the work presented in this thesis, is illustrated in Figure 4.1. From now on, the MRO is denoted as the maintenance workshop and O-level as the operational level. In this chapter, we take a closer look at each part of the problem, namely aircraft maintenance scheduling, maintenance workshop planning, (repaired and damaged) stocks of components, and operational demand. We present mathematical modeling for each part of the system, as well as their interconnections, and formulate optimization objectives for the respective stakeholders. Component flow can be tracked individually (i.e., each individual component is tracked with its id number) or in an aggregated fashion (i.e., all individual components of the same type are aggregated and only the number of components of a component type is tracked). In the maintenance workshop, we model repair lines and perform repairs such that we respect the capacity constraint and we may include job[1] modeling. Moreover, the workshop model may be preemptive or non-preemptive (that is, interruptions in repair for a specific job can or cannot occur). Whence, in this chapter we refer to three different models and for the sake of simplicity, introduce the following abbreviations:

- Individual component flow (ICF) model (Paper I),
- Aggregated component flow (ACF) model (Paper II) and
- Job flow (JF) model (Paper III, IV),

that will be used from now on. Further, two contracting forms between the stakeholders are defined and different versions of the optimization objectives

[1] Every action taken in the maintenance workshop is considered as a *job*.

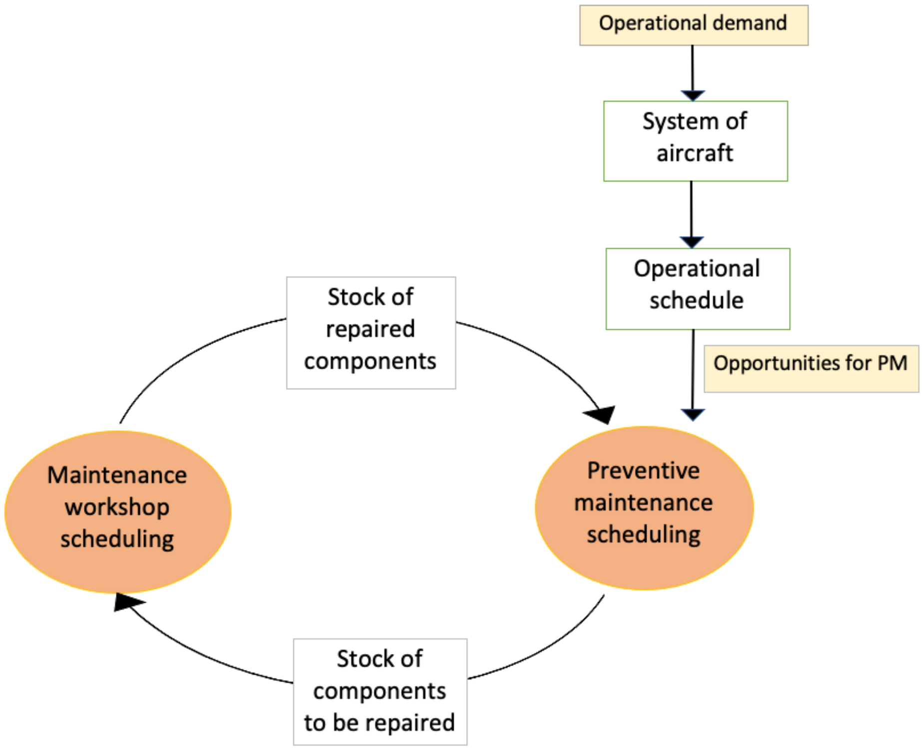

Figure 4.1: Scope of the thesis. Preventive maintenance scheduling for a fleet of aircraft and maintenance workshop scheduling, with operational demand as input to the model and the scheduling of component replacement and repair as output.

corresponding to the two contracts are presented. Lastly, a solution approach for the JF model, for a bi-objective problem that is difficult to solve, is presented.

4.1 Aircraft maintenance scheduling

The model of the maintenance scheduling problem presented is partly based on the *preventive maintenance scheduling problem with interval costs* (PMSPIC) presented in [32]. The PMSPIC considers a system with multiple component types and for which the costs for replacement of components take into account the interval between any two consecutive replacements/maintenance occasions; we generalize this model such that we allow for multiple systems and individual component modeling. The PMSPIC is partly an extension of the opportunistic replacement problem (ORP) studied in [3], described as follows: "The system consists of a set of components. The time between two consecutive replacements of a component may not exceed its assigned maximum replacement interval. To each time point in the planning period corresponds a fixed maintenance set-up cost and replacement costs for each component. The problem is to schedule the component replacements over a finite set of time points in order to minimize the total maintenance cost." Unlike the ORP model, the PMSPIC takes into account the intervals between two replacements (i.e., maintenance occasions) for each component and assigns a cost depending on the length of this maintenance interval.

We consider a fleet of $|\mathcal{K}|$ aircraft with $|\mathcal{I}|$ component types and $|\mathcal{J}_i|$ individual components of each type $i \in \mathcal{I}$. Maintenance can be scheduled at any time step t within the finite and discretized planning horizon $\mathcal{T} := \{0, \ldots, T+1\}$. A maintenance occasion of an aircraft k at time step t generates a maintenance cost. The maintenance interval (i.e., the interval between two maintenance occasions) of a component generates an interval cost, which is non-decreasing with the length of the interval. For each component type, by defining substantially higher costs for scheduling maintenance after—and also close before—the end of its life, unexpected failures are avoided; thereby our approach may stay within the scope of PM scheduling. We model this problem (denoted as GPMSPIC in [51] for the (ICF) and MS-PMSPIC in [52] for the (ACF) model), as a 0-1 mixed-integer linear optimization problem (see [18]); the decision variables are described below.

Decision variables (ICF). To determine the maintenance intervals of the components as well as the maintenance schedules for the aircraft, we define the decision variables

$$x_{st}^{ijk} = \begin{cases} 1, & \text{if individual component } j \text{ of type } i \\ & \text{in aircraft } k \text{ receives PM at times } s \\ & \text{and } t, \text{ but not in-between,} \\ 0, & \text{otherwise,} \end{cases} \qquad \begin{array}{l} j \in \mathcal{J}_i,\ i \in \mathcal{I},\ k \in \mathcal{K}, \\ 0 \leq s < t \leq T+1, \end{array}$$

$$z_t^k = \begin{cases} 1, & \text{if maintenance of aircraft } k \text{ occurs} \\ & \text{at time } t, \\ 0, & \text{otherwise,} \end{cases} \qquad k \in \mathcal{K},\ t \in \mathcal{T}.$$

Decision variables (ACF, JF). In the case of the aggregated component flow the z variables remain the same while the x variables are defined as

$$x_{st}^{ik} = \begin{cases} 1, & \text{if component of type } i \text{ in aircraft } k \\ & \text{receives PM at times } s \text{ and } t, \text{ but} \\ & \text{not in-between,} \\ 0, & \text{otherwise,} \end{cases} \qquad \begin{array}{l} i \in \mathcal{I},\ k \in \mathcal{K}, \\ 0 \leq s < t \leq T+1. \end{array}$$

The variable definition remains the same for the JF model as the job modeling does not belong to this part of the system–of–systems. As we assume that each aircraft $k \in \mathcal{K}$ is equipped with exactly one individual of each component type $\mathcal{I}$, the x variables will be binary with and without the individual component flow.

Constraints (ICF). The feasible set of the maintenance planning is modeled by the following equality and inequality constraints

$$\sum_{j \in \mathcal{J}_i} \sum_{s=0}^{t-1} x_{st}^{ijk} = \sum_{j \in \mathcal{J}_i} \sum_{r=t+1}^{T+1} x_{tr}^{ijk}, \qquad i \in \mathcal{I},\, t \in \mathcal{T},\, k \in \mathcal{K}, \tag{4.1a}$$

$$\sum_{j \in \mathcal{J}_i} \sum_{t=1}^{T+1} x_{0t}^{ijk} = 1, \qquad i \in \mathcal{I},\, k \in \mathcal{K}, \tag{4.1b}$$

$$\sum_{j \in \mathcal{J}_i} \sum_{s=0}^{t-1} x_{st}^{ijk} \leq z_t^k, \qquad i \in \mathcal{I},\, t \in \mathcal{T},\, k \in \mathcal{K}, \tag{4.1c}$$

$$\sum_{k \in \mathcal{K}} \sum_{s=0}^{t-1} x_{st}^{ijk} \leq 1, \qquad j \in \mathcal{J}_i,\, i \in \mathcal{I},\, t \in \mathcal{T}, \tag{4.1d}$$

$$x_{st}^{ijk} = 0, \qquad j \in \mathcal{J}_i,\, k \in \mathcal{K}, \quad \bar{t}_i \leq s + \bar{t}_i < t \leq T+1, i \in \mathcal{I}. \tag{4.1e}$$

For each aircraft k and component type i, a maintenance interval starts at time 0, which is modeled by (4.1b), while the constraints (4.1a) ensure that the same number (i.e., 0 or 1) of maintenance intervals ends and starts at time t. The constraints (4.1c) ensure that if a maintenance interval of component type i in aircraft k ends at time t, then maintenance of aircraft k must occur at time t. The constraints (4.1d) ensure that each component (i, j) is in at most one aircraft k at each time t. To prevent any maintenance interval for component type $i \in \mathcal{I}$ from being longer than $\bar{t}_i \leq T$, the constraints (4.1e) are defined. That, in effect, prevents from having to perform corrective maintenance (to an extent).

Constraints (ACF, JF). In the case of the aggregated component flow, constraints (4.1a)–(4.1e) become

$$\sum_{s=0}^{t-1} x_{st}^{ik} = \sum_{r=t+1}^{T+1} x_{tr}^{ik}, \qquad i \in \mathcal{I},\ t \in \mathcal{T},\ k \in \mathcal{K}, \tag{4.1f}$$

$$\sum_{t=1}^{T+1} x_{0t}^{ik} = 1, \qquad i \in \mathcal{I},\ k \in \mathcal{K}, \tag{4.1g}$$

$$\sum_{s=0}^{t-1} x_{st}^{ik} \leq z_t^k, \qquad i \in \mathcal{I},\ t \in \mathcal{T},\ k \in \mathcal{K}, \tag{4.1h}$$

$$x_{st}^{ik} = 0, \qquad \bar{t}_i \leq s + \bar{t}_i < t \leq T+1,\ i \in \mathcal{I},\ k \in \mathcal{K}. \tag{4.1i}$$

and their interpretation is similar as to the interpretation of (4.1a)–(4.1e). As for the variables, the constraints remain the same in the JF model.

4.2 Maintenance workshop scheduling

Components that should be maintained are sent to the maintenance workshop, which contains a number (L) of (identical) parallel repair lines for component repair. Each repair line has a repair capacity of one unit, while each component repair requires one unit of this capacity per time step during a prespecified (component type-specific) number of time steps. When a component arrives at the workshop, it is available for repair and (in the case of a turn–around time contract) assigned a due date, at which the repair should be finished, and the component returned back to the aircraft operator. This problem is identified as an *identical parallel machines scheduling problem* (IPMSP; Brucker and Knust (2012)), where a machine is equivalent to a repair line. For a survey of parallel machine scheduling problems, see [48]. In the classical deterministic IPMSP, there is a number of independent jobs to be processed on a range of identical machines. Each job has to be carried out on one of the machines during a fixed processing time, without preemption[2]. A component that finishes repair prior to (after) its due date generates a non-positive (non-negative) penalty cost, which applies only in the case of a turn–around time contract (see Section 4.5). A solution to the maintenance workshop scheduling problem specifies at which time each component arriving at the workshop should start maintenance. In case of the JF model, the solution also specifies on which machine the repair is performed while that information is lost with ICF and ACF models.

Decision variables and constraints (ICF). For each individual component j of each type i and for each time step t, we define

$$u_t^{ij} = \begin{cases} 1, & \text{if component } (i,j) \text{ starts repair at time } t, \\ 0, & \text{otherwise,} \end{cases} \qquad i \in \mathcal{I},\ j \in \mathcal{J}_i,\ t \in \mathcal{T}.$$

The number of active parallel repair lines at each time step t is defined by the

[2] If preemption (i.e., job splitting) is allowed, the processing of any operation may be interrupted and resumed at a later time.

non-negative integer variable ℓ_t, that should fulfill the constraints

$$0 \leq \ell_t = \ell_{t-1} + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}_i} \left(u_t^{ij} - u_{t-p^i}^{ij} \right) \leq L, \qquad t \in \mathcal{T}, \tag{4.2a}$$

where ℓ_0 and u_t^{ij}, $t \leq 0$, are initial (fixed) values that constitute input to the model.

Decision variables and constraints (ACF). In the case of the aggregated component flow, we define for each $i \in \mathcal{I}$ and $t \in \mathcal{T}$ the variables

$u_t^i \in \mathbb{Z}_+$: the number of components of type i starting maintenance at time t;

and ℓ_t becomes

$$0 \leq \ell_t = \ell_{t-1} + \sum_{i \in \mathcal{I}} \left(u_t^i - u_{t-p^i}^i \right) \leq L, \qquad t \in \mathcal{T}. \tag{4.2b}$$

As before, ℓ_0 and u_t^i, $t \leq 0$, are initial (fixed) values that constitute input to the model.

Both models (4.2a) and (4.2b) are preemptive models, which means that interruptions of repairs in the maintenance workshop are allowed. We next define a non-preemptive model of IPMSP combined with modeling of jobs in the maintenance workshop.

Decision variables (JF). For each job $n \in \mathcal{N}_i$ and each $i \in \mathcal{I}$, $l \in \mathcal{L}$, and $t \in \mathcal{T}$, the following binary variables are defined as

$$u_t^{inl} = \begin{cases} 1, & \text{if a component of type } i \text{ starts maintenance at time } t \text{ as job } n \text{ in machine } l, \\ 0, & \text{otherwise.} \end{cases}$$

Constraints (non-preemptive JF). For each $t \in \mathcal{T}$, the number $\ell_t \in \mathbb{Z}_+$ of active parallel machines at time t, and model the constraints

$$\sum_{i \in \mathcal{I}} \sum_{n \in \mathcal{N}_i} \sum_{s=t-p^i+1}^{t} u_s^{inl} \leq 1, \qquad t \in \mathcal{T},\ l \in \mathcal{L}, \tag{4.2c}$$

$$\sum_{l \in \mathcal{L}} \sum_{t \in \mathcal{T}} u_t^{inl} \leq 1, \qquad n \in \mathcal{N}_i,\ i \in \mathcal{I}, \tag{4.2d}$$

$$\sum_{l \in \mathcal{L}} \sum_{i \in \mathcal{I}} \sum_{n \in \mathcal{N}_i} \sum_{s=t-p^i+1}^{t} u_s^{inl} = \ell_t, \qquad t \in \mathcal{T}. \tag{4.2e}$$

The constraints (4.2c) state that each machine $l \in \mathcal{L}$ can process at most one job at each time step $t \in \mathcal{T}$, and that any job $n \in \mathcal{N}_i$ that starts processing in a machine l at a certain time step t, will occupy the machine until it is finished, at p^i time steps later, i.e., the scheduling is non-preemptive. The constraints (4.2d) make sure that each job $n \in \mathcal{N}_i$ is assigned to component type i at most once over all repair lines $l \in \mathcal{L}$ and all time steps $t \in \mathcal{T}$. In our study, we also vary the number of parallel machines (L), to enable decision support for capacity investments in the maintenance workshop. The constraints (4.2e) define the loading ℓ_t (i.e., the number of repair lines occupied) of the maintenance workshop at time step t. The constraints (4.2c) and (4.2e) together imply that the workshop loading ℓ_t cannot exceed the maximal number, L, of repair lines in the workshop. Naturally, non-preemption may be disabled in the JF model, in which case, the workshop model would be modeled similarly to (4.2b)

$$0 \leq \ell_t = \ell_{t-1} + \sum_{i \in \mathcal{I}} \sum_{n \in \mathcal{N}_i} \sum_{l \in \mathcal{L}} \left(u_t^{inl} - u_{t-p^i}^{inl} \right) \leq L, \qquad t \in \mathcal{T}, \tag{4.2f}$$

To model the interface between the variables defined for the two respective problems presented in Sections 4.1 and 4.2, I next introduce the dynamics of the stock of components.

4.3 Stocks of components modeling

When an individual component is taken out of an aircraft it is sent—with no time delay—to the stock of damaged components, where it stays until it is scheduled for repair. The transport time between the stock of damaged components and the maintenance workshop δ_a^i is prespecified. Upon being repaired, the component goes to the stock of repaired, so called *as good as new* components, again with a prespecified transport time between the workshop and stock of repaired components δ_b^i, where it is kept until its scheduled time for placement into an(other) aircraft. We assume that all transport times are represented by non-negative integers. Further, we introduce the variables and the constraints for the ICF, ACF and JF model.

Decision variables (ICF). To model the flow of components, we define the following binary variables

$$a_t^{ij}\ (b_t^{ij}) = \begin{cases} 1, & \text{if individual component } j \text{ of type } i \text{ is on the stock of} \\ & \text{damaged (repaired) components at time } t \in \mathcal{T} \cup \{0\}, \\ 0, & \text{otherwise,} \end{cases}$$

$$\alpha_t^{ij}\ (\beta_t^{ij}) = \begin{cases} 1, & \text{if individual component } j \text{ of type } i \text{ is taken out of (placed} \\ & \text{in) one of the aircraft } k \in \mathcal{K} \text{ at time } t \in \mathcal{T}, \\ 0, & \text{otherwise.} \end{cases}$$

Decision variables (ACF). In the case of the aggregated component flow, the variables are transformed to

$a_t^i\ (b_t^i)$: the number of individuals of component type i on the stock of damaged (repaired) components at time $t \in \mathcal{T} \cup \{0\}$;

$\alpha_t^i\ (\beta_t^i)$: the number of individuals of component of type i taken out of (placed in) any of the systems $k \in \mathcal{K}$ at time $t \in \mathcal{T}$,

for all $i \in \mathcal{I}$.

Decision variables (JF). Further, in the case of job flow modeling, the only modification (as compared to ACF) is the definition of the α variables to

$$\alpha_t^{ink} = \begin{cases} 1, & \text{if an individual of component type } i \text{ is taken out of a} \\ & \text{system } k \in \mathcal{K} \text{ at time } t \in \mathcal{T} \text{ and allocated to job } n \in \mathcal{N}_i; \\ 0, & \text{otherwise,} \end{cases}$$

for all $i \in \mathcal{I}$. The variables a_t^i, b_t^i and β_t^i remain the same as for the ACF.

Constraints (ICF). The stock of damaged components is then modeled by the constraints

$$\alpha_t^{ij} = \sum_{k \in \mathcal{K}} \sum_{s=0}^{t-1} x_{st}^{ijk}, \qquad j \in \mathcal{J}_i,\ i \in \mathcal{I},\ t \in \mathcal{T}, \tag{4.3a}$$

$$a_t^{ij} = a_{t-1}^{ij} + \alpha_t^{ij} - u_{t+\delta_a^i}^{ij} \in \{0,1\}, \quad t \in \{1-\delta_a^i, \ldots, T+1\}, j \in \mathcal{J}_i, i \in \mathcal{I}. \tag{4.3b}$$

The constraints (4.3a) connect the variables from the maintenance scheduling with the stock of damaged components: if a component (i,j) is taken out of any of the aircraft $k \in \mathcal{K}$ at time t, α_t^{ij} will take the value 1; otherwise α_t^{ij} takes the value 0. The constraints (4.3b) provide the state of component (i,j) at time t: whether it is on the stock of damaged components (i.e., $a_t^{ij} = 1$) or not (i.e., $a_t^{ij} = 0$). The state of a component at time t depends on its state in the previous time step $t-1$, whether it is taken out of any system k and placed on the stock at time step t, and whether it is starting maintenance at time step $t + \delta_a^i$.

The stock of repaired components is modeled analogously as

$$\beta_t^{ij} = \sum_{k \in \mathcal{K}} \sum_{r=t+1}^{T+1} x_{tr}^{ijk}, \qquad j \in \mathcal{J}_i,\ i \in \mathcal{I},\ t \in \mathcal{T} \tag{4.3c}$$

$$b_t^{ij} = b_{t-1}^{ij} - \beta_t^{ij} + u_{t-\delta_b^i-p^i}^{ij} \in \{0,1\}, \qquad j \in \mathcal{J}_i,\ i \in \mathcal{I},\ t \in \mathcal{T} \tag{4.3d}$$

$$\sum_{j \in \mathcal{J}_i} b_t^{ij} \geq \underline{b}^i, \qquad i \in \mathcal{I},\ t \in \mathcal{T}. \tag{4.3e}$$

The constraints (4.3c) represent the connection between the stock of repaired components and the maintenance scheduling. If component (i, j) is placed into any aircraft k at time t, β_t^{ij} will take the value 1; otherwise β_t^{ij} takes the value 0. In (4.3d) the individual states of the components at time t are updated: a component is either on the stock (i.e., $b_t^{ij} = 1$) or it is not (i.e., $b_t^{ij} = 0$). A component's state on the stock of repaired components is affected by its state in the previous time step $t-1$, whether it is placed in some system k at time t, and whether it will arrive to the stock at time t after being repaired (i.e., if $u_{t-\delta_b^i-p^i}^{ij} = 1$, which means that component (i, j) started maintenance at time $t - \delta_b^i - p^i$ and will arrive to the stock of repaired components at time t). The variables b_0^{ij}, β_0^{ij}, and u_t^{ij}, $t \in \{1 - \delta_b^i - p^i, \ldots, 0\}$, comprise (fixed) input data. Then, in (4.3e) it is expressed that the sum of the variables b_t^{ij} over the individual components, i.e., the stock level of repaired components per component type i at time t, may not be below the lower stock limit $\underline{b}^i$.

Constraints (ACF). The stock of damaged components is then modeled by the constraints

$$\alpha_t^i = \sum_{k \in \mathcal{K}} \sum_{s=0}^{t-1} x_{st}^{ik}, \qquad i \in \mathcal{I},\ t \in \mathcal{T}, \tag{4.3f}$$

$$a_t^i = a_{t-1}^i + \alpha_t^i - u_{t+\delta_a^i}^i \geq 0, \qquad i \in \mathcal{I},\ t \in \{1 - \delta_a^i, \ldots, T+1\}, \tag{4.3g}$$

while the stock of repaired components is modeled analogously, as

$$\beta_t^i = \sum_{k \in \mathcal{K}} \sum_{r=t+1}^{T+1} x_{tr}^{ik}, \qquad i \in \mathcal{I},\ t \in \mathcal{T}, \tag{4.3h}$$

$$b_t^i = b_{t-1}^i - \beta_t^i + u_{t-\delta_b^i - p^i}^i \geq \underline{b}^i, \qquad i \in \mathcal{I},\ t \in \mathcal{T} \cup \{T+1\}. \tag{4.3i}$$

The interpretation of the stock constraints with aggregated component flow remains similar to (4.3a)–(4.3e). The main difference is that now a_t^i and b_t^i correspond to the stock levels of component type i at time step t while in (4.3a)–(4.3e), a_t^{ij} and b_t^{ij} represent the $\{0,1\}$ state of an individual component (i,j) on the stock(s) (a component is either on a stock or not).

Constraints (JF). The model of the stock of damaged components becomes

$$\sum_{n \in \mathcal{N}_i} \alpha_t^{ink} = \sum_{s=0}^{t-1} x_{st}^{ik}, \qquad k \in \mathcal{K},\ t \in \mathcal{T}, \tag{4.3j}$$

$$\sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} \alpha_t^{ink} \leq 1, \qquad n \in \mathcal{N}_i, \tag{4.3k}$$

$$a_t^i - a_{t-1}^i = \sum_{n \in \mathcal{N}_i} \left(\sum_{k \in \mathcal{K}} \alpha_t^{ink} - \sum_{l \in \mathcal{L}} u_{t+\delta_a^i}^{inl} \right), \qquad t \in \{1-\delta_a^i, \ldots, T+1\}, \tag{4.3l}$$

$$a_t^i \geq 0, \qquad t \in \{1-\delta_a^i, \ldots, T+1\}, \tag{4.3m}$$

for all $i \in \mathcal{I}$, while the stock of repaired components is modeled as

$$\beta_t^i = \sum_{k \in \mathcal{K}} \sum_{r=t+1}^{T+1} x_{tr}^{ik}, \qquad t \in \mathcal{T}, \tag{4.3n}$$

$$b_t^i = b_{t-1}^i - \beta_t^i + \sum_{n \in \mathcal{N}_i} \sum_{l \in \mathcal{L}} u_{t-\delta_b^i - p^i}^{inl}, \qquad t \in \mathcal{T} \cup \{T+1\}, \tag{4.3o}$$

$$b_t^i \geq \underline{b}^i, \qquad t \in \mathcal{T}, \tag{4.3p}$$

for all $i \in \mathcal{I}$. Constraints (4.3k) ensure that a pair (i,n) can occur at most once during the planning period (i.e., a job can occur at most once). The interpretation of the remaining constraints in (4.3j)–(4.3p) remains similar as in the ACF model.

4.4 Operational demand

The system of aircraft considered possesses an operational demand, represented by a flight schedule that should be fulfilled. The flight schedule defines time intervals during which the aircraft is either operating or grounded. Thereby, the starting point for our modeling is precisely the operational demand (as illustrated in Figure 4.1).

For the maintenance planning problem, the flight schedule (i.e., operational schedule) is represented in terms of time intervals when the system is either operating – at which times maintenance cannot be performed – or accessible for maintenance. In other words, PM may not be scheduled while a system is operating. In the case of railway systems [41], each train is assigned time slots when it should operate (i.e., perform transport of goods or passengers); hence, PM may be scheduled only in-between those time slots. In the case of offshore wind turbine maintenance [60], the operational demand is fulfilled by wind energy production, while maintenance work can be done only during time periods of not too harsh weather conditions. When planning any PM occasion, the (predicted or planned) operational schedule for the systems provide time windows during which maintenance may be performed. As input to the integrated aircraft maintenance scheduling and maintenance workshop scheduling model (presented in Sections 4.1,4.2, 4.3), for all $t \in \mathcal{T}$ and all $k \in \mathcal{K}$ we thus use the parameters

$$\overline{z}_t^k = \begin{cases} 1, & \text{if PM is allowed to be scheduled for system } k \text{ at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

Further, we include the following constraints (for all three models ICF, ACF, JF) —such that the time windows for PM are respected—in our model:

$$z_t^k \leq \overline{z}_t^k, \qquad t \in \mathcal{T},\, k \in \mathcal{K}. \tag{4.4a}$$

An efficient way of generating the operational schedules (e.g., timetables) for the systems considered is presented in [30], in which the availability of a fleet of aircraft is maximized subject to requirements on the transport missions and maintenance of the aircraft and their components. An alternative way is presented in [16], where the maximal number of aircraft in maintenance at any given time during the planning period is minimized.

Another way of modeling (4.4a) may be to use slightly softer constraints to model the opportunities for performing maintenance, as follows:

$$\sum_{k \in \mathcal{K}} z_t^k \leq M, \qquad t \in \mathcal{T}, \tag{4.4b}$$

where (4.4b) limits the number of maintenance occasions for each time step t to at most M aircraft at a time. The main benefit of using this approach is that it gives more freedom to the model to choose the optimal maintenance schedules.

However, solving the integrated system–of–systems while fulfilling the operational demand may be done in a reversed, bottom-up way. After solving the integrated system–of–systems, flight schedule may be generates such that the PM occasions are input to the flight assignment problem. The flight assignment problem, subject to a given operational demand, is then solved. The approach in this thesis is, however, top-down.

4.5 Choice of optimization objectives

Up until this section, we have defined the feasibility problems for the ICF, ACF and JF models. We now discuss and define objective functions, that, together will the feasibility problems, define optimization problems. The definition of the different objectives for the ICF model is described below. The objectives are easily modified for the ACF and JF models, according to the variable and constraint definition in Sections 4.1–4.4.

Minimizing costs for maintenance set-up and intervals. Each maintenance occasion yields a set-up maintenance cost (d_t) for the aircraft operator. It can be either the cost of having an aircraft grounded/unavailable for flight operations, or the cost of performing any maintenance activity, or both. Besides the set-up cost, there is an interval cost (c^i_{st}) for every component which is determined based on the length of the interval between two consecutive maintenance occasions. We assume that the interval cost is non-decreasing with an increasing length of the interval. Furthermore, the longer the length of the maintenance interval is the more expensive it becomes to do maintenance. Using this cost structure enables prevention of (too) long maintenance intervals which could lead to over usage of a component and thereby, to component failure.

From the aircraft operators' point of view, the objective is to *minimize the total costs for maintenance*, which includes both set-up and interval costs, and it is modeled as to

$$\text{minimize} \quad \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} d_t z^k_t + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}_i} \sum_{t=1}^{T+1} \sum_{s=0}^{t-1} c^i_{st} x^{ijk}_{st}. \tag{4.5}$$

Minimizing the risk for lack of spare parts. To ensure that the operational schedule is undisturbed, or that the disturbance is minimal, it is crucial to have enough spare components available. When an unexpected failure occurs, whether there is disturbance in planned operations or not is determined by the availability of a "as good as new" component that will replace the one that has a failure. Thus, availability of components plays a crucial role when it comes to the stability of the system–of–systems. We give three formulations of an availability contract.

1. *Maximize a weighted average of the number of components (repaired or new) available*, which is modeled as to

$$\text{maximize} \qquad \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{I}} w^i \sum_{j\in\mathcal{J}_i} b_t^{ij}, \tag{4.6a}$$

   where $w^i > 0$ is an objective weight assigned to component type $i \in \mathcal{I}$.

2. The risk for lack of spare components may be minimized by *maximizing a weighted average of the lower limits on the numbers of available components of each type*, subject to a lower bound on the availability of each component type, i.e., to

$$\text{maximize} \qquad \sum_{i\in\mathcal{I}} w^i e^i, \tag{4.6b}$$

$$\sum_{j\in\mathcal{J}_i} b_t^{ij} \geq e^i \geq \underline{b}^i, \qquad i \in \mathcal{I},\, t \in \mathcal{T}, \tag{4.6c}$$

   where, for each component type $i \in \mathcal{I}$, $w^i > 0$ denotes the weight assigned while the lower limit on the number of available components is denoted by e^i and $\underline{b}^i \geq 0$ is the lower limit on the level of available components of type $i \in \mathcal{I}$ (as defined in Section 4.3).

3. Whenever the stock level for component type i goes below a predefined threshold $\underline{\underline{b}}^i$, there is a non-negative penalty c_i^{AV} for every unit y_t^i of number of components at time step t that go below $\underline{\underline{b}}^i$. That is,

$$\text{minimize} \qquad \sum_{i\in\mathcal{I}} c_i^{\mathrm{AV}} \sum_{t\in\mathcal{T}} y_t^i, \tag{4.6d}$$

$$y_t^i \geq \underline{\underline{b}}^i - \sum_{j\in\mathcal{J}_i} b_t^{ij}, \qquad i \in \mathcal{I},\ t \in \mathcal{T}, \tag{4.6e}$$

$$y_t^i \geq 0, \qquad i \in \mathcal{I},\ t \in \mathcal{T}, \tag{4.6f}$$

$$b_t^{ij} \in \{0,1\}, \qquad i \in \mathcal{I}, j \in \mathcal{J}_i \tag{4.6g}$$

   where the inequalities (4.6e)–(4.6f) define y_t^i as the measure of how much the stock level $\sum_{j\in\mathcal{J}_i} b_t^{ij}$ for component type i at time t falls below the lower limit $\underline{\underline{b}}^i$ on the stock of available components.

Minimizing the risk for exceeding the contracted turn–around times for component repair. The *turn–around time* of an individual component (i, j) is defined as the time from when the component is taken out of an aircraft in $\mathcal{K}$ until it has become repaired and is available for usage again in an(other) aircraft. The total turn–around time, v_{tat}^{ij}, for component individual (i, j), $j \in \mathcal{J}_i$, $i \in \mathcal{I}$, over the planning period $\mathcal{T}$, is thus computed as

$$v_{\text{tat}}^{ij} = \sum_{t=0}^{T+1} \left(\left(t + p^i + \delta_b^i\right) u_t^{ij} - t\alpha_t^{ij} \right), \tag{4.7a}$$

where the term $(p^i + \delta_b^i)u_0^{ij}$ is positive if component (i, j) is initially on the stock of damaged components, and the equalities $u_{T+1}^{ij} = a_0^{ij} - u_0^{ij} + \sum_{t\in\mathcal{T}}(\alpha_t^{ij} - u_t^{ij})$ and $\alpha_{T+1}^{ij} = 0$ hold.[3] The shortest possible turn–around time for a component of type i equals $\delta_a^i + p^i + \delta_b^i$, i.e., the sum of the repair time in the maintenance workshop and the time required for the transportation to and from the workshop. Letting $c_{\text{delay}}^{ij} > 0$ and $c_{\text{early}}^{ij} \in (0, c_{\text{delay}}^{ij}]$ denote the penalty for late and early[4], respectively, delivery of a repaired component, this objective is then expressed as to

$$\text{minimize} \quad \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} \left(c_{\text{delay}}^{ij} v_{\text{delay}}^{ij} - c_{\text{early}}^{ij} v_{\text{early}}^{ij} \right), \tag{4.7b}$$

where v_{delay}^{ij} (v_{early}^{ij}) denotes the total delay (earliness) for component (i, j) over the planning period. These variables are due to the constraints

$$v_{\text{early}}^{ij} \leq v_{\text{tat}}^{ij} - q_{\text{due}}^{ij} \left(a_0^{ij} + \sum_{t=1}^{T+1} \alpha_t^{ij} \right) \leq v_{\text{delay}}^{ij}, \tag{4.7c}$$

$$v_{\text{early}}^{ij} \leq 0 \leq v_{\text{delay}}^{ij}, \tag{4.7d}$$

where $q_{\text{due}}^{ij} > 0$ denotes the contracted due date for component (i, j), $j \in \mathcal{J}_i$, $i \in \mathcal{I}$. Due to the construction of (4.7c)–(4.7d) either v_{early}^{ij} or v_{delay}^{ij} (or both) will attain value zero when the objective (4.7b) is optimized (a component will be either early, or late, or on time; in the latter case $v_{\text{early}}^{ij} = v_{\text{delay}}^{ij} = 0$ hold). Therefore, for each component (i, j) the objective (4.7b) *minimizes the penalty for total lateness or earliness.* Typically, v_{delay}^{ij} would be penalized and optimized for, and v_{early}^{ij} could be removed from the model formulation, leading

[3] Note that the use of the variables a_0^{ij} and u_{T+1}^{ij} leads to a possible underestimate of v_{tat}^{ij}, as we possibly shorten the v_{tat}^{ij} for components which were initialized on the stock of components to be repaired at $t = 0$ and components which did not finish repair until $t = T + 1$. It follows that v_{early}^{ij} (v_{delay}^{ij}), as defined in (4.7c)–(4.7d), will possibly be under(over)estimated.

[4] A penalty for earliness will, by construction, be a reward.

to the following reformulation of (4.7b)–(4.7d):

$$\text{minimize} \quad \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}_i} c^{ij}_{\text{delay}} v^{ij}_{\text{delay}}, \tag{4.7e}$$

$$v^{ij}_{\text{tat}} - q^{ij}_{\text{due}}\left(a^{ij}_0 + \sum_{t=1}^{T+1}\alpha^{ij}_t\right) \leq v^{ij}_{\text{delay}}, \tag{4.7f}$$

$$v^{ij}_{\text{delay}} \geq 0. \tag{4.7g}$$

4.5.1 Other objectives that could be utilized

There are more objectives that could be discussed and included in the multi-objective setting, if relevant and desired. Two such examples are given below.

Minimizing investment costs for repair lines in the workshop. We assume that each repair line in the maintenance workshop comes with an investment cost $c_{\text{inv}} > 0$. One of the objectives on the maintenance workshop side would be to minimize the investment costs for repair lines in the workshop. The workshop capacity costs are then addressed as to

$$\text{minimize} \quad c_{\text{inv}} L, \tag{4.8}$$

where the parameter L would then be regarded as a decision variable, which takes the role of an upper limit, as expressed in the constraints (4.2a). This objective is relevant when investigating the optimal workshop capacity.

Minimizing the costs of performing repairs. Each maintenance activity associated with a component (i, j) normally has a repair cost c^{ij}_{repair}, which may depend on the component's processing time p^{ij}, be non-decreasing with an increasing value of p^{ij} and assigned to the u^{ij}_t variables. The objective is to

$$\text{minimize} \quad \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}_i} c^{ij}_{\text{repair}}(p^{ij}) u^{ij}_t, \tag{4.9}$$

and it is relevant if we want to optimize the number of maintenance activities. Within the application presented in this thesis, components have to be repaired regardless of the price of repair, and it is not possible to merge two or more repairs together to minimize the costs (opportunistic replacement planning, see e.g. [3]); thus, we neglect this objective at the current stage. Moreover, (4.9) would lead to a so-called zero-sum game [5], which would not have a great impact on the bi-objective analysis presented in our current work.

4.6 Contracting forms and the bi-objective problem definition

The collaboration between the stakeholders is regulated with a contractual agreement. In reality, there are many ways to define a contract. Based on industrial input, we define and study two contract types: an *availability* and a *turn–around time* contract. Both are some measures of component flow over time: availability represents the levels of available components on the stock of repaired components and turn–around time measures the lateness/earliness of component delivery after being repaired. Neither of the two contract types is uniquely defined. Aircraft operator has the same objective in both contracts,

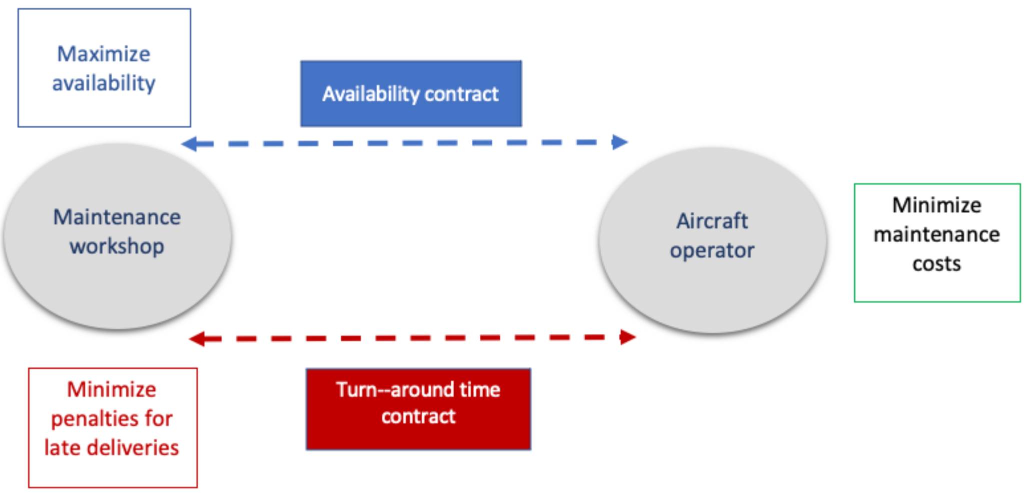

Figure 4.2: Contracting forms between the stakeholders. One of the two contracts, availability or turn–around time, is employed. While the aircraft operator has the same objective in the case of both contracts, the choice of the optimization objective for the maintenance workshop depends on the contract choice.

and that is to minimize maintenance costs. Depending on the contract type, maintenance workshop has two different objectives: maximizing availability (or, minimizing availability penalties, depending on the choice of the contract

[5]In game theory and economic theory, a *zero-sum game* is a mathematical representation of a situation in which an advantage that is won by one of two sides is lost by the other. See e.g. [69].

	model	constraints defining the feasible set for the bi-objective problem	preemption in the workshop	contract; optimization objectives
1.	ICF	(4.1a)–(4.1e), (4.2a), (4.3a)–(4.3d), (4.4a)	preemptive	AV; (4.5), (4.6a)
2.	ICF	(4.1a)–(4.1e), (4.2a), (4.3a)–(4.3e), (4.4a),(4.7c)–(4.7d)	preemptive	TAT; (4.5), (4.7b)
3.	ACF	(4.1f)–(4.1i), (4.2b) (4.3f)–(4.3i), (4.4a), (4.6e)–(4.6g)	preemptive	AV; (4.5),(4.6d)
4.	JF	(4.1f)–(4.1i), (4.2c)–(4.2e) (4.3j)–(4.3o),(4.4a), (4.6e)–(4.6g)	non-preemptive	AV; (4.5), (4.6d)
5.	JF	(4.1f)–(4.1i), (4.2c)–(4.2e) (4.3j)–(4.3p),(4.4a), (4.7f)–(4.7g)	non-preemptive	TAT; (4.5), (4.7e)
6.	JF	(4.1f)–(4.1i), (4.2f) (4.3j)–(4.3o),(4.4a)	preemptive	AV; (4.5), (4.6d)
7.	JF	(4.1f)–(4.1i), (4.2f) (4.3j)–(4.3p),(4.4a), (4.7f)–(4.7g)	preemptive	TAT; (4.5), (4.7e)

Table 4.1: Summary of the bi-objective optimization problems studied. For each model ICF, ACF and JF, preemptive or non-preemptive scheduling in the maintenance workshop, and an availability (AV) or turn–around time (TAT) contract, we define a bi-objective optimization problem constituted by a set of constraints (i.e., a feasible set) and two optimization objective functions.

formulation) when the contract in place is availability and minimizing penalties for late (and early) deliveries in the case of a turn–around time contract. This is illustrated in Figure 4.2. The two bi-objective optimization problems, *the availability bi-objective problem* and *the turn–around time bi-objective problem*, together with their feasible sets and optimization objectives, are summarized in Table 4.1.

4.7 Problem complexity

After introducing a mathematical model, a relevant question is, to which complexity class it belongs to. Without loss of generality, we choose the JF model for the following theorem and definition 3. for the availability objective function (4.6). The same result can be proven for the ICF and ACF models.

Theorem 1 (**Complexity of the availability bi-objective problem**)**.** The complete JF model of the system–of–systems (4.1)–(4.4), with either of the objective functions (4.5) or (4.6d), binary requirements on the variables x_{st}^{ik}, z_t^k,

u_t^{inl}, and α_t^{in}, and non-negativity and integer requirements on the variables y_t^i, a_t^i, b_t^i, β_t^i, and ℓ_t, for all relevant values of the indices, is NP-hard.

Proof. Consider the JF constraints (4.1)–(4.4), with the relevant binary, non-negativity, and integer requirements on the variables. Assume that the capacity of the maintenance workshop equals the total number of jobs, i.e., that $L = \sum_{i\in\mathcal{I}} N_i$ holds. Moreover, assume that for each component type $i \in \mathcal{I}$, the number of individual components fulfills $J_i \geq K \left\lceil \frac{T+1}{1+\delta_a^i+p^i+\delta_b^i} \right\rceil$ (where 1 represents the shortest time a component spends in a system k), that $\bar{b}_0^i = J_i - K$, and that $\underline{\underline{b}}^i = 0$. Then, each repair job can always be instantly performed in the workshop and there will always be a (repaired) component in stock for replacement.

The problem (4.1)–(4.4), with the objective to minimize (4.5), is hence reduced to the minimization of (4.5) subject to (4.1g)–(4.1e), which separates into one instance of the PMSPIC for each of the systems $k \in \mathcal{K}$. As stated in Definition 1 in [52] and the reasoning thereafter, the PMSPIC is NP-hard; see also [32]. Therefore, there exists an instance of the problem of minimizing (4.5) which is NP-hard.

Now, consider the problem (4.1)–(4.4), (4.6e), (4.6f), with the objective to minimize (4.6d). Since $\underline{\underline{b}}^i = 0$, the problem separates over minimizing (4.6d) subject to $y_t^i \geq 0$ for all relevant i and t, and minimizing a zero objective subject to (4.1)–(4.4), and $b_t^i \geq 0$ for all relevant i and t. Minimization of (4.6d) subject to $y_t^i \geq 0$ for all relevant i and t results in $y_t^i = 0$, for all relevant i and t (for all non-negative cost coefficients) and there are no constraints involving y (or any of other) variables. For the former problem, setting all y variables to zero will be an optimal solution, such that the optimal value of (4.6d) will equal zero (that is, $C^{\mathrm{AV}}(y^*) = 0$ for any optimal solution $(x^*, z^*, u^*, \ldots, y^*)$ since $y^* = 0$ is optimal whenever $\underline{\underline{b}}^i = 0$). The latter problem is reduced to the problem of minimizing (4.5) subject to (4.1)–(4.4), with the costs $c_{st}^i = 0$ and $d_t = 0$, for all relevant indices. Hence, there exists an instance of the problem of minimizing (4.6d) which is NP-hard.

We conclude that the availability bi-objective problem is NP-hard. □

In a similar fashion, it can be proved that the turn–around time bi-objective problem, for any of the ICF, ACF or JF models, is NP-hard. The proof is given in Paper IV.

The complete problem may be reduced (at most) to an NP-hard problem (PMSPIC, [32]), regardless of the contract type, model selection (ICF, ACF,

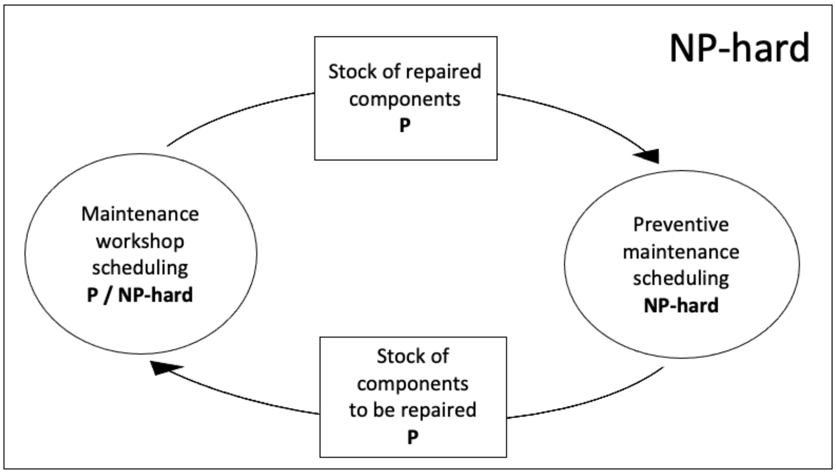

Figure 4.3: Complexity of isolated parts within the system–of–systems. Each part of the system–of–systems has its own complexity, which has an effect on the complexity of the integrated system–of–systems.

JF) and enabling preemption or not. If we consider each part of the whole system–of–systems independently, as in Figure 4.3, as isolated systems: the stock dynamics equations can be formulated as a network flow model, where each equality constraint is a node balancing constraint. The problem is an LP and thus can be solved in polynomial time; The preemptive IPMSP with a (weighted) sum objective is polynomially solvable [40, Ch. 8.0], whereas its version with a minimax (i.e., makespan) objective is NP-hard [7, Ch. 2.1]. Finding an optimal non-preemptive schedule on parallel machines with a makespan objective is NP-hard even for the case of two identical machines [62]. If we have a non-preemptive IPMSP with an objective that is neither a (weighted) sum nor a makespan, we cannot with certainty conclude its computational complexity.

4.8 Bounding the Pareto front of a turn–around time bi-objective optimization problem

The turn–around time contract is first defined for the ICF model (Paper I). However, due to it becoming computationally intractable for larger instances, either a new formulation of the turn–around time model or an efficient way of solving the existing one is required. It appears to be challenging to define a formulation of the turn–around time without the individual tracking of components (as in Paper I, using ICF) or tracking of jobs (as in Paper III, using JF). The second one proves to be more efficient when optimizing for not being late with component delivery upon repair, hence we choose the JF model formulation. To solve the problem more efficiently, we Lagrangean relax the complicating constraints defining the delay penalties.

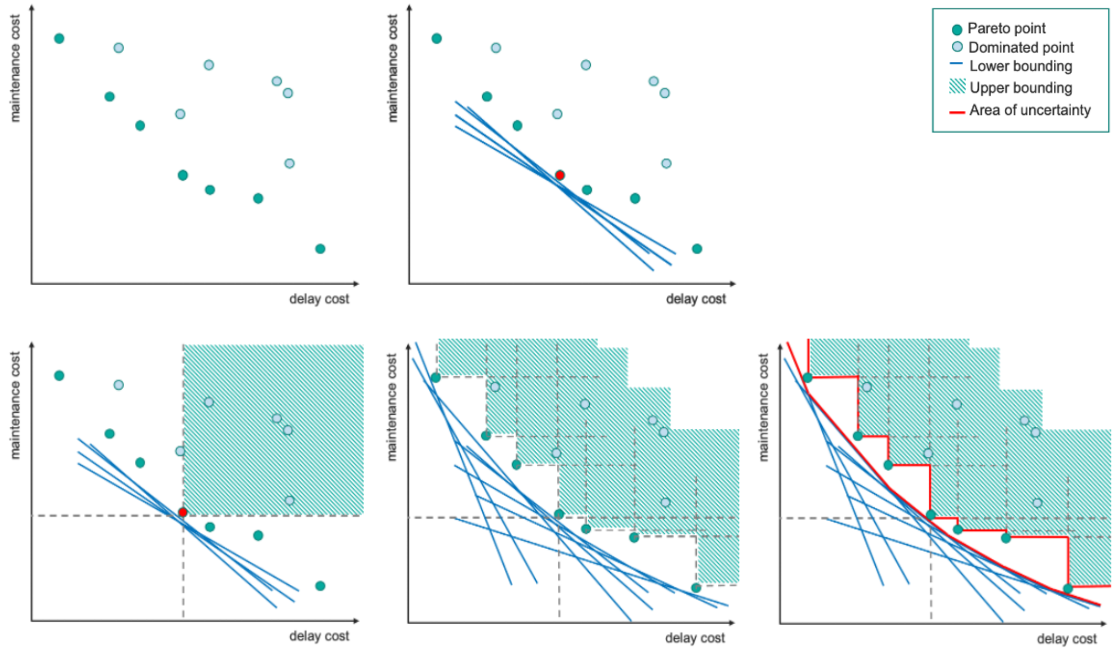

Figure 4.4: A simplified illustration of the bounding of the Pareto front. The delay cost is the objective transformed into an ϵ-constraint. For each value of ϵ, using the subgradient algorithm, a number of linear lower bounds on the optimal solution of the ϵ-constraint scalarized problem reformulation is obtained. In combination with the pair of upper bounds on the two optimization objectives, we attain the area of uncertainty in which the Pareto points may be found.

Summary of the method. Let us consider (either of) the bi-objective optimization problem(s) 5. (or 7.) from Table 4.1. The ϵ-constraint scalarization of the problem (see Section 3.2.1) gives a scalarized, single objective optimization problem, in which one of the two objectives is constrained in an ϵ-constraint. Next, the complicating constraints ((4.7f) defined for the JF model) are Lagrangian relaxed (see Section 3.4) and the remaining constraints define the Lagrangean subproblem. For each value of ϵ, the subgradient algorithm (see Section 3.5) is performed and the dual variables (i.e., penalties for the Lagrangean relaxed constraint) are updated in each subgradient iteration. For each pair of values of ϵ and of the dual variable, a linear lower bound is obtained, as the Lagrangean dual function is a linear function of ϵ. In addition, the Lagrangean dual function is a lower bound on the optimal primal solution. The linear lower bound is a valid bound for all points on the Pareto front for the bi-objective problem. Further, a pair of upper bounds (for the two objectives in the bi-objective problem) is derived. This idea is illustrated in Figure 4.4 and a complete mathematical derivation is presented in Paper IV.

5 Summaries of the appended papers

The foundation for the models formulated in Papers I–III is a mixed-binary linear optimization (MBLP) model of a preventive maintenance scheduling problem with so-called interval costs over a finite and discretized time horizon. We extend this scheduling model with the individual (ICF) or aggregated (ACF) flow of components, or with a job flow (JF) through the repair workshop. Next, we include stocks of spare components, both those components that need repair and the repaired ones. The resulting scheduling model is then utilized in the optimization of (one of the) two main contracts, namely maximizing the availability of repaired (or new) components, and minimizing the deviation from the contracted turn–around times for the components in the maintenance loop. Each of these objectives are combined—in a bi-objective setting—with the objective to minimize the costs for maintenance of the operating systems. While Papers I–III primarily address the modeling, Paper IV concerns method development.

paper	model	preemption in the workshop	availability contract	turn–around time contract
I	ICF	+	+	+
II	ACF	+	+	-
III	JF	-	+	-
IV	JF	+	+	+

Table 5.1: Summary of the main characteristics of the appended papers.

In this chapter, we summarize the respective papers, whose main properties and differences are highlighted in Table 5.1. The papers were developed as enumerated and each of the papers I–III motivated the development of the next one.

5.1 Paper I: Scheduling of repair and replacement of individual components in operating systems

We develop a mixed-binary linear optimization (MBLP) model for simultaneous preventive maintenance scheduling and (preemptive) workshop planning, in which the components are tracked individually. To ensure that the operational schedule is undisturbed, or at least that the disturbance is minimal, it is crucial to have enough spare components available. The availability objective is defined as maximization of a weighted average of the number of repaired (or new) components available. The turn–around time objective minimizes the total penalty for late and early component delivery. Each of these objectives, together with a minimization of the maintenance costs, lead to two bi-objective problems, one corresponding to an availability contract and the other corresponding to a turn–around time contract.

Our model assumes deterministic processing times in the maintenance workshop. Uncertainty, such as unexpected events, may, however, affect the processing times. Unexpected change in processing time(s) is incorporated in our model by a re-solution of the scheduling problem for the new processing time(s), from the point in time at which the processing time(s) changed (i.e., the time step when the component at hand started repair in the workshop). The re-solution is performed whenever an unexpected event is revealed. The complete schedule employed is then composed by the computed part-schedules, as defined by each consecutive pair of time steps at which a(ny) longer processing time(s) were detected.

We analyze the two contracting forms by studying and comparing the Pareto fronts resulting from different parameter settings, regarding minimum allowed stock levels and investments in repair capacity of the workshop. Our specific results concern the effect on the levels of the stocks of components. We conclude that our bi-objective mixed-binary linear optimization model is able to capture important properties of the results from the contracting forms.

Main results and contribution:

1. Formulation of an integrated preventive maintenance scheduling and maintenance workshop planning, including the modeling of stock dy-

namics. The model includes individual component flow and preemptive scheduling in the maintenance workshop, as well as a rescheduling algorithm in case of unexpected events.

2. Definition of the two contracting forms between the stakeholders, an availability and a turn–around time, leading to two bi-objective optimization problems.

3. The turn–around time contract, which introduces non-binary coefficients in the constraint matrix, becomes computationally intractable for larger instance sizes. An availability contract is computationally more tractable as it allows removal of individual components in the model, thereby more suitable for larger instances and real world problems.

This paper is submitted to the Journal of Scheduling and is under review for publication. Initial ideas were presented on The First EUROYoung Workshop, Seville (2019) and some later ideas on the Swedish Operations Research Conference, Nyköping (2019) and PLANs forsknings- och tillämpningskonferens, KTH Södertälje (2020).

5.2 Paper II: Simultaneous scheduling of replacement and repair of common components in operating systems

The key bottleneck in Paper I is the intractability of the suggested model, primarily for a turn–around type contract between the stakeholders, but also for the availability contract when the size of the instance that is to be solved increases. Ergo, the question is how to preserve the main model formulation but improve its efficiency.

The model formulated in this paper preserves the structure of the model developed in Paper I; whence a preventive maintenance scheduling problem, integrated with the maintenance workshop planning via the stocks of spare components. Unlike in Paper I, the individual component flow is removed and instead, we only consider component types. That leads to a lot of variables becoming integer rather than binary. Hence, the mixed-binary linear optimization model becomes a mixed-integer linear problem (MILP).

The stakeholders' collaboration is based upon an availability contract. A new definition of the availability contract is presented, in which the lower limit on the number of available components for a component type is maximized. Together with minimizing the maintenance (interval and set-up) costs, the bi-objective problem corresponding to the availability contract is defined. The resulting model is then utilized in the optimization of the availability contract.

Removal of individuality proves to be advantageous as we are able to keep the main model properties and at the same time solve the problem more efficiently. The trade-off, however, appears once we want to define a component repair turn–around time, for which the aggregation over individual components is not sufficient.

Main results and contribution:

1. An aggregation of the individual components for each component type leads to an efficient framework for the availability type of contract.
2. The resulting solutions can be used to find a lower limit for an optimal performance of a collaboration between the stakeholders. Our results concern the effect of our modeling on the levels of the stocks of components over time, in particular minimizing the risk for lack of spare components. The model can be used for analyzing the change in the solution when some parameters (e.g., the maintenance workshop capacity) are varied.
3. This modeling provides a planning tool when the maintenance workshop and the system operator are integrated, and a decision support, regardless of the integration.

This paper has been published in Annals of Operations Research [52] (2023). The results were presented on the MAPSP (Models and Algorithms for Planning and Scheduling Problems) conference in Oropa, Italy (2022) as well as on a SANU (Serbian Academy of Sciences and Arts) seminar in Belgrade, Serbia (2022).

5.3 Paper III: An enhanced mathematical model for optimal simultaneous preventive maintenance scheduling and workshop planning

In Paper I, we formulate a model with individual flow of components and in Paper II, a model with an aggregated flow of components through the whole system–of–systems. The first one showed to be computationally intractable for a turn–around contract type while the second one appeared to be reasonably fast for the availability contract type. The next question is if we can formulate an efficient model that allows modeling of a component turn–around time. In addition, the modeling of the maintenance workshop in Papers I–II is rather simple. We assumed preemptive flow of components through the workshop, which means that job interruptions are allowed. While preemption can sometimes be advantageous, we are also interested in formulating a non-preemptive model for the maintenance workshop, where a job that has started must be completed with no interruptions (neither in terms of time nor in terms of repair line), which we do in this paper.

As a compromise in between the individual and the aggregated flow of components through the system–of–systems, we introduce the flow of jobs through the maintenance workshop, facilitating the preservation of the main model features but also enabling the computation of turn–around times. The availability objective is now formulated as minimization of penalties for going below a predefined lower limit on the stock of repaired components. We further analyze the computational complexity of the resulting bi-objective MILP and prove that it belongs to the class of NP-hard problems (see Section 3.6), hence it is a computationally demanding problem.

We analyze the uncertainty intervals of the availability penalty reduction for an increased workshop capacity, and possible reductions of the availability penalty for an increased maintenance workshop capacity. In addition, utilization of components in the systems is computed as the averaged wasted component life over the total number of replacements over the planning horizon, indicating that there is a trade-off between decreasing the risk of component failure and increasing its utilization.

Main results and contribution:

1. An availability bi-objective mathematical model that enables non-preemptive job scheduling in the maintenance workshop, which brings this model closer to a real world application.
2. A model that is solvable in reasonable time and allows for computation of component turn–around times.
3. Our results measure the interplay between the workshop capacity and the level of component availability, as well as the corresponding cost trade-off between the stakeholders.

This paper is submitted to EURO Journal on Decision Processes (2023).

5.4 Paper IV: Approximating the Pareto frontier for bi-objective preventive maintenance and workshop scheduling. A Lagrangean lower bounding methodology for evaluating contracting forms

One of the results from Paper III is that, with the JF model, we have the means to compute the turn–around times without tracking the individual components. Instead, the turn–around is computed for each job/repair in the workshop. The next step is to formulate a model for the turn–around time contract between the stakeholders, that, as before, consists of the minimization of the maintenance cost as one objective and minimization of penalties for late deliveries as the other objective. Even though the JF model is more efficient than an ICF, optimizing for turn–around time (i.e., minimizing the penalties for late deliveries of repaired components) still leads to long computing times.

This paper develops a solution approach for solving a computationally heavy bi-objective optimization problem, which corresponds to the turn–around time contract and JF model. In order to manage the computations in reasonable time, we use Lagrangean relaxation (see Section 3.4) and subgradient optimization (see Section 3.5) to find lower bounding functions—in the objective space—of the set of non-dominated solutions, complemented with math-heuristics to identify good feasible solutions (see Section 4.8). The main bottleneck in our method is the individual tracking of jobs that is necessary for computing the turn–around times. For that reason, we suggest aggregation over jobs in some parts of the model, and a heuristic for splitting the aggregated variables back into individual jobs upon solving the subproblem. As a result, we obtain an approximation of the Pareto frontier for bi-objective preventive maintenance and workshop scheduling.

The suggested method may be utilized for further evaluation and comparison of contracting forms between the stakeholders. At the current stage, when it comes to the costs the stakeholders face, our results indicate that an availability contract performs better than a turn–around time contract. While penalties do not necessarily represent the actual costs, they constitute a measure of contract violation. Our results indicate that delay penalties are higher as compared to the availability penalties.

Main results and contribution:

1. Lagrangean relaxation and subgradient algorithm development for a bi-objective optimization problem. The problem is defined for a turn–around time contract, but the framework may be utilized for different problems/optimization objectives as well.
2. Pareto front bounding for a turn–around time bi-objective problem, that is normally very computationally expensive to compute.
3. Framework for the comparison between the two contract types.

This paper is a manuscript. The main idea of this work was presented on the Northern Lead Day conference in Gothenburg, Sweden (2023).

6 Conclusions and future perspectives

Mathematical models are often created subject to limitations and simplifications, hence they cannot be seen as complete representations of reality. Rather than striving to describe the reality in its fullest, a good objective is to approximate it by focusing on important problem-specific aspects. A representative model enables systematic analyses and understanding of complex systems, prediction, insight deduction and cost-effective testing of theories and hypotheses, before committing to expensive experimentation or development.

This thesis summarizes the development of mathematical modeling and solution approaches for simultaneous scheduling of component replacement and repair. On our trajectory towards a comprehensive model, we made certain modifications, assumptions and improvements, which resulted in Papers I–III. In Paper I, we show the viability of the integrated preventive maintenance and workshop scheduling problem, define the contracting forms between the stakeholders and the corresponding bi-objective optimization problems. The turn–around time contract becomes computationally intractable for larger instance sizes, indicating that an availability contract is more computationally tractable. Removing the individuality of components improves the computing times, leading to Paper II. The resulting solutions may be used to find a lower limit for an optimal performance of a collaboration between the stakeholders, regulated by an availability contract. An availability bi-objective mathematical model that enables non-preemptive job scheduling in the maintenance workshop, with a new definition of the availability contract, is presented in Paper III. Modeling of jobs allows for computation of turn–around times, but at the same time, is less computationally demanding as compared to the individual tracking of components. The bi-objective turn–around time optimization problem is still difficult to solve, hence the modeling developed in Paper III requires solution approaches that make it practically useful. The methodology, that is based on

Lagrangean lower bounding, heuristically computed upper bounds, and problem relaxation techniques, is developed in Paper IV. As a result, the Pareto front for a turn–around time bi-objective optimization problem, which is in practice unsolvable, is approximated. In light of this, a framework for comparison of the contracts between the stakeholders is obtained.

Our work indicates that an availability contract type leads to lower maintenance costs and may be a better choice as compared to a turn–around time contract type. Evidently, this result is dependent on the data utilized. Nevertheless, while the turn–around time contract only focuses on repairing and returning components within a contracted due date, an availability contract reduces the risk of not fulfilling the operational schedule—which may come very costly—by ensuring component availability on the stock. Moreover, the availability models are computationally easier to handle, thus more appealing for practical usage.

Inspiration for this work came from an industrial problem in aerospace. However, the application of the models and methods presented goes beyond the original scope, as they may be applied for any system–of–systems with a similar problem structure. The resulting scheduling modeling provides decision support for the stakeholders, and may be used as a planning tool when the stakeholders operate in an integrated fashion. Our modeling provides a framework for aiding and making decisions.

Future research

There are a few directions in which this work may be extended. An extension important for the intended application of this work is to introduce corrective maintenance modeling. At the current stage, the means to handle unexpected failures are to reduce the risk for such failures by not allowing too large maintenance intervals, and to reschedule the maintenance plan whenever an unexpected event occurs. Short-term changes in the operational schedules, as well as in the schedules for the maintenance workshop, are often inconvenient and sometimes not even feasible. Thus, if possible, the rescheduling should be such that the solution remains fixed for a certain number of time steps.

The total number of components to be considered can be quite large and, as discussed in Chapter 1, it is challenging to model them all. Since only a subset of the total number of components is safety critical, they are the ones that are of main interest and constitute the driving force of the whole system–of–systems. Another extension could be to cluster the non-safety critical components in a way such that they can be incorporated into the modeling. Additionally, we may differentiate between missions that are supposed to be flown, and aircraft configurations required to perform them, as that may have an impact on maintenance policies, wear of and demand for components.

To model the whole system–of–systems presented in Figure 2.2, we may expand to more than one maintenance workshop, include external subcontractors and a contract regulating Saab's collaboration with each one of them. Instead of having operational schedules as inputs to the model, the problem of aircraft operations scheduling may be included as well.

For the purpose of further evaluating the contracts governing the collaboration between the stakeholders, the modeling may be expanded with additional indicators and optimization objectives.

As previously discussed, the problem at hand is a complex one. Thus, methods and solution approaches for improving the computational speed are needed (e.g., problem relaxation, decomposition, algorithm development), especially if the above suggested model extensions would be incorporated into the modeling.

Bibliography

[1] C. M. Able, A. H. Baydush, C. Nguyen, J. Gersh, A. Ndlovu, I. Rebo, J. Booth, M. Perez, B. Sintay, and M. T. Munley. A model for preemptive maintenance of medical linear accelerators—predictive maintenance. *Radiation Oncology*, 11, 2016. doi: 10.1186/s13014-016-0602-1.

[2] A. Aliano Filho, W. A. Oliveira, and T. Melo. Multi-objective optimization for integrated sugarcane cultivation and harvesting planning. *European Journal of Operational Research*, 309:330–344, 2023. doi: 10.1016/j.ejor.2022.12.029.

[3] T. Almgren, N. Andréasson, M. Patriksson, A.-B. Strömberg, A. Wojciechowski, and M. Önnheim. The opportunistic replacement problem: Theoretical analyses and numerical tests. *Mathematical Methods of Operations Research*, 76(3):289–319, 2012. doi: 10.1007/s00186-012-0400-y.

[4] N. Andréasson, M. Patriksson, and A. Evgrafov. *An Introduction to Continuous Optimization: Foundations and Fundamental Algorithms.* Dover Books on Mathematics. Dover Publications, 3rd edition, 2020. ISBN 978-0486802879.

[5] K. R. Baker. *Introduction to Sequencing and Scheduling.* Wiley, 1974.

[6] L. Boliang, J. Wu, R. Lin, J. Wang, H. Wang, and X. Zhang. Optimization of high-level preventive maintenance scheduling for high-speed trains. *Reliability Engineering & System Safety*, 183:261–275, 2019. doi: 10.1016/j.ress.2018.11.028.

[7] P. Brucker and S. Knust. *Complex Scheduling.* GOR-Publications. Springer-Verlag, Berlin, 2nd edition, 2012.

[8] M. Bussieck, T. Winter, and U. Zimmermann. Discrete optimization in public rail transport. *Mathematical Programming, Series B*, 79(1–3): 415–444, 1997. doi: 10.1007/BF02614327.

[9] A. M. Campbell and T. V. Woensel. Special issue on recent advances in urban transport and logistics through optimization and analytics. *Transportation Science*, 53(1):1–5, 2019. doi: 10.1287/trsc.2018.0882.

[10] A. Caprara, M. Fischetti, and P. Toth. A heuristic method for the set covering problem. *Operations Research*, 47(5):730–743, 1999. doi: 10.1287/opre.47.5.730.

[11] C. R. Cassady and E. Kutanoglu. Integrating preventive maintenance planning and production scheduling for a single machine. *IEEE Transactions on Reliability*, 54(2):304–309, 2005. doi: 10.1109/TR.2005.845967.

[12] S. A. Celtek, A. Durdu, and M. E. M. Alı. Real-time traffic signal control with swarm optimization methods. *Measurement*, 166:108206, 2020. doi: 10.1016/j.measurement.2020.108206.

[13] V. Chankong and Y. Y. Haimes. Multiobjective decision making. Theory and methodology. North-Holland Series in System Science and Engineering, 8. New York- Amsterdam: North-Holland Publishing Co. XVII, 1983.

[14] S. Chansombat, P. Pongcharoen, and C. Hicks. A mixed-integer linear programming model for integrated production and preventive maintenance scheduling in the capital goods industry. *International Journal of Production Research*, 57(1):61–82, 2019. doi: 10.1080/00207543.2018.1459923.

[15] Y. Chen, P. Cowling, F. Polack, S. Remde, and P. Mourdjis. Dynamic optimisation of preventative and corrective maintenance schedules for a large scale urban drainage system. *European Journal of Operational Research*, 257(2):494–510, 2017. doi: 10.1016/j.ejor.2016.07.027.

[16] P. Cho. Optimal scheduling of fighter aircraft maintenance. Master's thesis, Massachusetts Institute of Technology, 2011. URL `http://hdl.handle.net/1721.1/67773`.

[17] R. Cigolini, L. Fedele, M. Garetti, and M. Macchi. Recent advances in maintenance and facility management. *Production Planning & Control*, 19 (4):279–286, 2008. doi: 10.1080/09537280802034034.

[18] M. Conforti, G. Cornejols, and G. Zambelli. *Integer programming*. Number 271 in Graduate Texts in Mathematics. Springer International Publishing, Switzerland, 2014.

[19] J. A. Day and L. L. George. Opportunistic replacement of fusion power system parts. *Conference: Opportunistic replacement of fusion power system parts*, 1981. URL `https://www.osti.gov/biblio/5559083`.

[20] B. de Jonge and P. A. Scarf. A review on maintenance optimization. *European Journal of Operational Research*, 285(3):805–824, 2020. doi: 10.1016/j.ejor.2019.09.047.

[21] R. Dekker, R. Wildeman, and F. van der Duyn Schouten. A review of multi-component maintenance models with economic dependence. *Mathematical Methods of Operations Research*, 45:411–435, 1997. doi: 10.1007/BF01194788.

[22] M. Ehrgott. *Multicriteria Optimization*. Springer, Berlin, 2nd edition, 2005. doi: 10.1007/3-540-27659-9.

[23] M. Ehrgott. A discussion of scalarization techniques for multiple objective integer programming. *Annals of Operations Research*, 147:343–360, 2006. doi: 10.1007/s10479-006-0074-z.

[24] B. Einabadi, M. Mahmoodjanloo, A. Baboli, and E. Rother. Dynamic predictive and preventive maintenance planning with failure risk and opportunistic grouping considerations: A case study in the automotive industry. *Journal of Manufacturing Systems*, 69:292–310, 2023. doi: 10.1016/j.jmsy.2023.06.012.

[25] M. Erkoc and K. Ertogral. Overhaul planning and exchange scheduling for maintenance services with rotable inventory and limited processing capacity. *Computers & Industrial Engineering*, 98:30–39, 2016. doi: 10.1016/j.cie.2016.05.021.

[26] R. Z. Farahani, E. Miandoabchi, W. Szeto, and H. Rashidi. A review of urban transportation network design problems. *European Journal of Operational Research*, 229(2):281–302, 2013. doi: 10.1016/j.ejor.2013.01.001.

[27] E. Fatnassi, J. Chaouachi, and W. Klibi. Planning and operating a shared goods and passengers on-demand rapid transit system for sustainable city-logistics. *Transportation Research Part B: Methodological*, 81:440–460, 2015. doi: 10.1016/j.trb.2015.07.016.

[28] N. Firdaus, H. Ab-Samat, and B. T. Prasetyo. Maintenance strategies and energy efficiency: a review. *Journal of Quality in Maintenance Engineering*, 29:640–665, 2023. doi: 10.1108/JQME-06-2021-0046.

[29] S. Fotedar, A. Strömberg, T. Almgren, and S. Cedergren. A criterion space decomposition approach to generalized tri-objective tactical resource allocation. *Computational Management Science*, 20(17), 2023. doi: 10.1007/s10287-023-00442-6.

[30] A. Gavranis and G. Kozanidis. An exact solution algorithm for maximizing the fleet availability of a unit of aircraft subject to flight and maintenance requirements. *European Journal of Operational Research*, 242:631–643, 2015. doi: 10.1016/j.ejor.2014.10.016.

[31] M. Guignard. Lagrangean relaxation. *Top*, 11:151–200, 2003. doi: 10.1007/BF02579036.

[32] E. Gustavsson, M. Patriksson, A.-B. Strömberg, A. Wojciechowski, and M. Önnheim. The preventive maintenance scheduling problem with interval costs. *Computers & Industrial Engineering*, 76:390–400, 2014. doi: 10.1016/j.cie.2014.02.009.

[33] E. Gustavsson, M. Patriksson, and A.-B. Strömberg. Primal convergence from dual subgradient methods for convex optimization. *Mathematical Programming*, 150:365–390, 2015. doi: 10.1007/s10107-014-0772-2.

[34] L. Göransson, C. Granfeldt, and A.-B. Strömberg. Management of wind power variations in electricity system investment models. *Operations Research Forum*, 2(25), 2021. doi: 10.1007/s43069-021-00065-0.

[35] D. Hrabec, L. M. Hvattum, and A. Hoff. The value of integrated planning for production, inventory, and routing decisions: A systematic review and meta-analysis. *International Journal of Production Economics*, 248:108468, 2022. doi: 10.1016/j.ijpe.2022.108468.

[36] J. Jiang, Y. An, Y. Dong, J. Hu, Y. Li, and Z. Zhao. Integrated optimization of non-permutation flow shop scheduling and maintenance planning with variable processing speed. *Reliability Engineering & System Safety*, 234: 109143, 2023. doi: 10.1016/j.ress.2023.109143.

[37] R. Kasimbeyli, Z. K. Ozturk, N. Kasimbeyli, G. D. Yalcin, and B. Icmen Erdem. Comparison of some scalarization methods in multiobjective optimization. *Bulletin of the Malaysian Mathematical Sciences Society*, 42: 1875—1905, 2019. doi: 10.1007/s40840-017-0579-4.

[38] K. Kobayashi, Y. Takano, and K. Nakata. Cardinality-constrained distributionally robust portfolio optimization. *European Journal of Operational Research*, 309(3):1173–1182, 2023. doi: 10.1016/j.ejor.2023.01.037.

[39] J. Kurz. Capacity planning for a maintenance service provider with advanced information. *European Journal of Operational Research*, 251(2): 466–477, 2016. doi: 10.1016/j.ejor.2015.11.029.

[40] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys. Chapter 9: Sequencing and scheduling: Algorithms and complexity. In

Logistics of Production and Inventory, volume 4 of *Handbooks in Operations Research and Management Science*, pages 445–522. Elsevier, 1993.

[41] T. Lidén. Coordinating maintenance windows and train traffic: a case study. *Public Transport*, 12:261–298, 2020. doi: 10.1007/s12469-020-00232-2.

[42] Z. Liu, F. Wen, and G. Ledwich. Optimal planning of electric-vehicle charging stations in distribution systems. *IEEE Transactions on Power Delivery*, 28(1):102–110, 2013. doi: 10.1109/TPWRD.2012.2223489.

[43] D. T. Luc. Pareto optimality. In A. Chinchuluun, P. M. Pardalos, A. Migdalas, and L. Pitsoulis, editors, *Pareto Optimality, Game Theory And Equilibria*, pages 481–515. Springer, New York, USA, 2008. ISBN 978-0-387-77247-9. doi: 10.1007/978-0-387-77247-$9_1$8.

[44] X. Luo, Y. Sun, X. Liu, Y. Gao, H. Sun, and Y. Liu. Course timetable optimization for a university teaching building considering the building energy efficiency and time-varying thermal perception of students. *Building and Environment*, 219:109175, 2022. doi: 10.1016/j.buildenv.2022.109175.

[45] R. Marler and J. Arora. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26:369–395, 2004. doi: 10.1007/s00158-003-0368-6.

[46] C. Miller, A. Tucker, and R. Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM*, 7:326–329, 1960. doi: 10.1145/321043.321046.

[47] S. Mohammad Hadian, H. Farughi, and H. Rasay. Development of a simulation-based optimization approach to integrate the decisions of maintenance planning and safety stock determination in deteriorating manufacturing systems. *Computers & Industrial Engineering*, 178:109132, 2023. doi: 10.1016/j.cie.2023.109132.

[48] E. Mokotoff. Parallel machine scheduling problems: A survey. *Asia-Pacific Journal of Operational Research*, 18:193–242, 2001.

[49] J. Mula, R. Poler, J. García-Sabater, and F. Lario. Models for production planning under uncertainty: A review. *International Journal of Production Economics*, 103(1):271–285, 2006. doi: 10.1016/j.ijpe.2005.09.001.

[50] R. P. Nicolai and R. Dekker. Optimal maintenance of multi-component systems: A review. In *Complex System Maintenance Handbook*, pages 263–286. Springer London, London, GB, 2008. ISBN 978-1-84800-011-7. doi: 10.1007/978-1-84800-011-$7_1$1.

[51] G. Obradović. *Mathematical Modeling, Optimization and Scheduling of Aircraft's Components Maintenance and of the Maintenance Workshop.* Chalmers University of Technology, Licentitate thesis, 2021. URL `https://research.chalmers.se/publication/524023/file/524023_Fulltext.pdf`.

[52] G. Obradović, A.-B. Strömberg, and K. Lundberg. Simultaneous scheduling of replacement and repair of common components in operating systems. *Annals of Operations Research*, 322:147–165, 2023. doi: 10.1007/s10479-022-04739-8.

[53] N. Papakostas, P. Papachatzakis, V. Kanthakis, D. Mourtzis, and G. Chryssolouris. An approach to operational aircraft maintenance planning. *Decision Support Systems*, 48:604–612, 2010. doi: 10.1016/j.dss.2009.11.010.

[54] M. L. Pinedo. *Scheduling: Theory, Algorithms, and Systems.* Springer, 5th edition, 2016.

[55] B. Polyak. Minimization of unsmooth functionals. *USSR Computational Mathematics and Mathematical Physics*, 9(3):14–29, 1969. doi: 10.1016/0041-5553(69)90061-5.

[56] I. Riepin, M. Schmidt, L. Baringo, and F. Müsgens. Adaptive robust optimization for European strategic gas infrastructure planning. *Applied Energy*, 324:119686, 2022. doi: 10.1016/j.apenergy.2022.119686.

[57] E. Robert, C. Berenguer, K. Bouvard, H. Tedie, and R. Lesobre. Joint dynamic scheduling of missions and maintenance for a commercial heavy vehicle: value of on-line information. *IFAC PapersOnLine*, 51(24):837–842, 2018. doi: 10.1016/j.ifacol.2018.09.672. 10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS 2018.

[58] D. A. Rossit, F. Tohmé, and M. Frutos. The non-permutation flow-shop scheduling problem: A literature review. *Omega*, 77:143–153, 2018. doi: 10.1016/j.omega.2017.05.010.

[59] F. Schulze Spüntrup, G. D. Ave, L. Imsland, and I. Harjunkoski. Integration of maintenance scheduling and planning for large-scale asset fleets. *Optimization and Engineering*, 23:1255–1287, 2022. doi: 10.1007/s11081-021-09647-7.

[60] M. Shafiee, M. Patriksson, and A.-B. Strömberg. An optimal number-dependent preventive maintenance strategy for offshore wind turbine blades considering logistics. *Advances in Operations Research*, 2013:205847, 2013. doi: 10.1155/2013/205847.

[61] L. Shi, H. Yao, M. Shan, Q. Gao, and X. Jin. Robust control of a space robot based on an optimized adaptive variable structure control method. *Aerospace Science and Technology*, 120:107267, 2022. doi: 10.1016/j.ast.2021.107267.

[62] A. J. Soper and V. A. Strusevich. Preemptive and non-preemptive scheduling on two unrelated parallel machines. *Journal of Scheduling*, 25:659–674, 2022. doi: 10.1007/s10951-022-00753-7.

[63] C. Sriram and A. Haghani. An optimization model for aircraft maintenance scheduling and re-assignment. *Transportation Research Part A: Policy and Practice*, 37(1):29–48, 2003. doi: 10.1016/S0965-8564(02)00004-6.

[64] K. Thörnblad, A.-B. Strömberg, M. Patriksson, and T. Almgren. Scheduling optimisation of a real flexible job shop including fixture availability and preventive maintenance. *European Journal of Industrial Engineering*, 9(1): 126–145, 2015. doi: 10.1504/EJIE.2015.067451.

[65] Z. Tian, T. Jin, B. Wu, and F. Ding. Condition based maintenance optimization for wind power generation systems under continuous monitoring. *Renewable Energy*, 36(5):1502–1509, 2011. doi: 10.1016/j.renene.2010.10.028.

[66] S. Tzvetkova and B. Klaassens. Preventive maintenance for industrial application. *IFAC Proceedings Volumes*, 34(29):3–8, 2001. doi: 10.1016/S1474-6670(17)32784-2. 6th IFAC Symposium on Cost Oriented Automation (LCA 2001), Berlin, Germany, 8–9 October 2001.

[67] M. Verhoeff and W. J. C. Verhagen. Component maintenance planning optimization in defense aviation. *Aerospace*, 10(3), 2023. doi: 10.3390/aerospace10030255.

[68] M. Verhoeff, W. J. C. Verhagen, and R. Curran. Maximizing operational readiness in military aviation by optimizing flight and maintenance planning. *Transportation Research Procedia*, 10:941–950, 2015. doi: 10.1016/j.trpro.2015.09.048.

[69] J. von Neumann and O. Morgenstern. *Theory of games and economic behavior*. Princeton: Princeton University Press, 60-anniversary edition, 2007.

[70] H. Wang and H. Pham, editors. *Reliability and Optimal Maintenance*. Springer Series in Reliability Engineering, Springer, 2006. doi: 10.1007/b138077.

[71] L. Weinstein and C.-H. Chung. Integrating maintenance and production decisions in a hierarchical production planning environment. *Computers & Operations Research*, 26(10):1059–1074, 1999. doi: 10.1016/S0305-0548(99)00022-2.

[72] D. Wu and H. Xia. A multi-processor system real-time scheduling algorithm for flight assignment problem. *Lecture Notes in Electrical Engineering*, 126: 507–514, 2012. doi: 10.1007/978-3-642-25766-$7_6$8.

[73] Q. Yu, M. Patriksson, and S. Sagitov. Optimal scheduling of the next preventive maintenance activity for a wind farm. *Wind Energy Science*, 6 (3):949–959, 2021. doi: 10.5194/wes-6-949-2021.

[74] Q. Zhang, Y. Liu, T. Xiahou, and H.-Z. Huang. A heuristic maintenance scheduling framework for a military aircraft fleet under limited maintenance capacities. *Reliability Engineering & System Safety*, 235:109239, 2023. doi: 10.1016/j.ress.2023.109239.

[75] E. Åblad, D. Spensieri, R. Bohlin, J. S. Carlson, and A.-B. Strömberg. Spatial–temporal load balancing and coordination of multi-robot stations. *IEEE Transactions on Automation Science and Engineering*, 2022. doi: 10.1109/TASE.2022.3214567.

[76] M. Önnheim, E. Gustavsson, A.-B. Strömberg, M. Patriksson, and T. Larsson. Ergodic, primal convergence in dual subgradient schemes for convex programming, II: the case of inconsistent primal problems. *Mathematical Programming*, 163:57–84, 2017. doi: 10.1007/s10107-016-1055-x.