MSc Thesis Report

A practical maintenance task packaging model applicable to aircraft maintenance

M.M.D. Witteman

Delft University of Technology

TU Delft Delft University of Technology

Challenge the future

MSc Thesis Report

A practical maintenance task packaging model applicable to aircraft maintenance

by

M.M.D. Witteman

To obtain the degree of Master of Science,
at the Delft University of Technology,
to be defended publicly on Thursday April 18, 2019 at 14:00.

Student number:	4294289	
Student:	M. M. D. Witteman	
Thesis committee:	Prof. Dr. R. Curran,	TU Delft, committee chairman
	Dr. ir. C.C. de Visser,	TU Delft, committee
	Dr. ir. B. F. Lopes dos Santos,	TU Delft, supervisor
	Mr. Q. Deng,	TU Delft, daily supervisor

An electronic version of this thesis is available at http://repository.tudelft.nl. The author can be reached at the following e-mail address: max.witteman@gmail.com.

TU Delft Delft University of Technology

Abstract

This research concerns the problem of scheduling aircraft maintenance tasks, that must be carried in multiple maintenance checks to keep the aircraft airworthy. During this research the allocation of maintenance tasks to their maintenance opportunities is referred to as the task allocation problem. It is a complex combinatorial problem that is solved daily by aircraft operators.

We propose a novel two-stage framework capable of solving the task allocation problem, for an entire fleet. A Mixed-Integer Linear Programming (MILP) formulation was developed for both stages of the framework. In the first stage an exact method is used to pre-determine the workforce to allocate per day to each aircraft under maintenance. In the second stage, the task allocation problem is solved independently for each aircraft, allocating task at the work shift level and per workforce skill. Moreover, an approximation algorithm is proposed, using bin packaging problem solution techniques, for both stages of the framework.

Both stages were tested and validated using data of 45 aircraft from a European airline. The computational performance of the approximation algorithm is benchmarked using a MILP formulation, solved with a commercial solver. Results indicate that the solution quality of the proposed algorithm remains excellent and runs up to 26 times faster than the exact method.

This research is the first to present an optimization model to solve the maintenance task allocation problem, being solved at the work shift level. Furthermore, it presents a bin-packaging approximation algorithm that can be solved within minutes for the 45 aircraft fleet and a planning horizon of 4 years. The framework was validated in practice and the results give insights on the potential benefit of using such framework to solve this daily recurrent problem.

Acknowledgements

As with any research teamwork is key to achieve success. Firstly, I would like to thank the people that were directly involved in my research. I would like to thank my daily supervisor, Qichen Deng, who supported me throughout the last ten months. His tremendous support and expertise helped me a lot with my research. I admire his effort in helping me even in times where both of us experienced difficult moments. I would also like to thank my other supervisor, Bruno Santos, who steered me in the right direction whenever he thought it was needed. Bruno expertise on optimization models helped me a lot with understanding and implementing my own models. This research would not have been so complete and practical without the support of the airline. The airline provided me with data, validated my models and answered any questions I had. I want to thank them as well as my supervisors for this great opportunity.

Secondly, I would like to thank the people that were not directly linked with my research. I would like to thank Neil Yorke-Smith, who I contacted during my initial phase of my research. His expertise on optimization algorithms helped me a lot with understanding the possibilities and expected outcomes of my exact models. I admire that he wanted to meet several times to discuss some of my concerns. I would also like to thank the experts that were involved in answering questions related to their publications: Anthony Muchiri, Karsai Gabor, Maclein Pereira, and Albert Steiner. Their input gave me additional insight in possible solutions for solving the task allocation problem.

I would also like to thank my friends and fellow students Mamdouh, Ashwyn and Joey for supporting me throughout my studies. My girlfriend, Angie, supported me a lot during my studies and helped me escape the grind of studying during the weekends. For which I would like to thank her. The past years of my study would not have been possible without the support of my parents both mentally and financially, for which I am immensely grateful.

Max Witteman
Delft, March 2019

Contents

1
Introduction

1.1. Introduction

Aircraft maintenance is essential for aircraft fleet management since it usually accounts for a considerable part of the overall operational cost and sets constraints on the planning of flight operations. Estimations reported by the industry of aeronautics indicate that maintenance activities correspond from 10% to 20% of an operator's direct operating cost [1]. This is a substantial part which needs to be taken seriously in order to enhance operational efficiency [2].

Before the introduction of the Boeing 777 most maintenance programs were developed around the various letter checks (i.e. A/B/C/D checks). This traditional approach had a primary intention to ensure that aircraft remained airworthy and on schedule [3]. Nowadays modern maintenance schedules are developed based on the latest Maintenance Steering Group (MSG-3) approach [4]. The MSG-3 approach can be classified as a top-down approach, whereby failure analyses is focused at the systems level. The primary objective of this approach is to preserve system function, thereby not every item of equipment is equally important. Since MSG-3 revision 2, which coincided with the introduction of the B777, letter checks were no longer prescribed. Maintenance tasks are now controlled individually, which allows operators to execute maintenance on their systems or equipment at the most appropriate time [3, 5].

The scheduling of aircraft maintenance is the allocation of maintenance tasks, that must be carried out on specific aircraft, at a given time in order to keep the aircraft airworthy. This is a complex combinatorial problem that needs to be solved daily by aircraft operators. The large amount of constraints and the complexity of cost makes the maintenance scheduling problem a time consuming job, especially for large fleets [2]. During this research the problem of allocating tasks to maintenance opportunities is referred to as the task allocation problem.

Current approaches in aircraft maintenance scheduling is to utilize the maintenance opportunities efficiently by appropriate grouping of maintenance tasks. This act of grouping maintenance tasks is called task packaging. The packaging of tasks allows an efficient use of maintenance opportunities but leads to waste of aircraft component life when tasks are performed before its due date. This research was initiated by an airline that expressed his interest in designing a framework that is capable of solving the task allocation problem down to shift level. Currently, the airline uses a manual based approach to allocate the maintenance tasks to their maintenance opportunity for the entire fleet.

This thesis report describes a two stage model framework capable of solving the task allocation problem at the shift level. The models will take into account all the routine maintenance tasks required during A-check and C-check maintenance opportunities. The framework included a non-routine factor to account for expected non-routine maintenance tasks. Each of the models were tested during a case study using data of 45 aircraft from a European airline.

1.2. Outline report

The first chapters of this research are dedicated to describing the main literature found and to present the research framework. Thereafter, the thesis is separated into three main parts. The first part explains the model formulation for each of the presented models. The second part presents the results of the case study using data from a European airline. The last part consists of the conclusion and recommendation chapters.

This report starts with a literature review on general maintenance planning strategies and packaging strategies (Chapter 2). In addition similar well-known problems in literature were examined including techniques to solve them. Based on these findings the fundamental knowledge was obtained to set-up a research framework in chapter 3. In chapter 4 the available data was presented and the proposed model framework for this research was introduced.

The first part of this report consists of the model formulation. In chapter 5 the task allocation model for the tactical level is introduced. This includes the mathematical formulation, objective function, constraints and model flow of both the exact method and the approximation algorithm. Chapter 6 discusses the formulation of the task allocation model at the operational levels for both the exact method and approximation algorithm.

In the second part of this report a case study is conducted. Results of applying the airlines data to each of the exact models is presented in chapter 7. In chapter 8 the exact methods were verified and validated. After the validation of the exact models the approximation algorithms were verified by comparison with the obtained solutions from the exact methods. A sensitivity analysis was conducted, in chapter 9, to see how some of the key parameters and assumptions associated with the case study affect the results of the models.

The last part of this report consists of conclusions and recommendations chapters. In chapter 10 the research questions were answered. Finally, recommendations for improving the results, obtained throughout this research, are given in chapter 11.

2

Literature review

Prior to this research a literature study was conducted to seek the main elements of the task allocation problem. This chapter will summarize the main findings obtained during the literature study. For the full literature study the reader is referred to Witteman [6].

2.1. Introduction to aircraft maintenance

Modern maintenance schedules are developed based on the latest Maintenance Steering Group (MSG-3) approach [4]. The MSG-3 approach can be classified as a top-down approach, whereby failure analyses is focused at the systems level. The primary objective of this approach is to preserve system function, thereby not every item of equipment is equally important. It is important to know if the failure affects the safe operation of the aircraft. Hence, MSG-3 assigns failures to two basic categories: safety or economics [5]. The MSG-3 analysis results in an original maintenance program, the tasks selected are published in a document called the Maintenance Review Board Report (MRBR). This report outlines the initial minimum scheduled maintenance/inspection requirements [3, 7]. It is used by the operator, after approval by the appropriate authority, to create their own approved maintenance program [3, 5]. In addition to the MRBR the aircraft manufactures publish their own document, Maintenance Planning Document (MPD), for maintenance planning. This document contains all the tasks described in the MRBR and additional tasks suggested by the airframe manufacturer [5]. Before the introduction of the Boeing 777 most maintenance programs were developed around the various letter checks (i.e. A/B/C/D checks). This traditional approach had a primary intention to ensure that aircraft remained airworthy and on schedule [3]. Typical maintenance intervals for the heavy checks are listed in table 2.1.

	B747-400	**A300B4**
A check	Every 600 FH	Divided in 4 parts (A1,A2,A3,A4) Every 385 FH or 11 weeks
B check	In two parts (B1,B2) Every 1200 FH	None
C check	In two parts (C1, C2) Every 5000 FH or 18 months	In two parts (C1, C2) Every 3000 FH or 18 months
D check	First check between 25k-27.5k FH Subsequent every 25k FH or 6 years	Every 12k FH or 4 years

Table 2.1: Maintenance check interval data obtained from Kinnison et al. [5].

Since MSG-3 revision 2, which coincided with the introduction of the B777, letter checks were no longer prescribed. Maintenance tasks are now controlled individually, this allows operators to execute maintenance on their systems or equipment at the most appropriate time [3, 5]. The benefits of this new approach are illustrated in the paper of Ozkol and Senturk [8], in which they compared the single task-orientated maintenance concept with the block check approach. Results of a case study show that the proposed approach could potentially save millions of dollars during the lifespan of a single aircraft.

2.1.1. Maintenance planning requirements

The adequate planning of aircraft maintenance activities entails a critical role in achieving airline objectives. The ultimate objective of aircraft maintenance is to deliver airworthy aircraft in time to meet the flight schedule [9]. One of the requirements for an aircraft, to be airworthy, is to perform the maintenance in accordance with the approved maintenance program [3]. The maintenance tasks in the maintenance program should be accomplished within the given interval. The intervals for these tasks fall within different categories: operational units (i.e. flight hours, flight cycles), calendar units (i.e. hour(s), day(s), month(s), year(s)) and other codes (i.e. national requirement, vendor requirement). In order to better address the organisational needs it is often required to change the intervals of specific tasks [5], which may be done only if initial requirements of the MPD are not exceeded [10]. Several tasks could claim the same limited resources. Therefore, it is necessary to find a way to serialize these concurrent tasks according to their priorities. These limited resources may include but are not limited to equipment and tools, staff with different qualifications and working space with limited access capabilities [11]. Other factors that may influence the maintenance planning are wrong estimation of required man-hours per task, and fluctuations in man power availability [9].
Long-term maintenance plans are affected by daily operations that lead to unscheduled maintenance required for the safe operation of aircraft. Unscheduled maintenance can lead to flight delays or cancellations if the item is listed in the minimum equipment list [12]. Which lead to complications in scheduling thereby negatively influencing the overall maintenance objectives (i.e cost, time quality or safety). This causes the need for additional resources and materials [13]. A diversity of papers claim that unscheduled maintenance can be as high as 50 per cent of the total workload [13–15]. These serious consequences should be managed in order to achieve effective and efficient maintenance [16]. The stochastic nature of the unscheduled maintenance activities makes it difficult to forecast and manage them [17]. Grey [18] described the problem of planning non-routine unplanned work. According to the paper traditional approaches of dealing with this problem are additional margins for deterministic tasks or the allocation of additional buffer time at the end of the project schedule. The paper describes seven net buffer sizing techniques, which improve the duration and the stability of the project. The paper of Rosales, Chen and Yang [17] proposed an approach to estimate the non-routine rate (i.e. relationship between number of unscheduled and scheduled activities) of an aircraft. The approach was based on the Evidential Reasoning (ER) rule that analysed the historical data of different operational variables related to the aircraft. The ER rule can be classified as a general probabilistic reasoning process which combines different pieces of independent evidence each having their own corresponding weights and reliabilities [19]. In the paper the variables age, flight hours per year, flight cycles per year, and prior non-routine rate distribution are used as the four pieces of evidence whereon the ER is applied.

2.1.2. Packaging strategies

There are thousands of parts and components on an aircraft that need preventive maintenance [8, 20]. Each time an aircraft is on the ground is regarded as an opportunity for maintenance [12]. In order to use maintenance opportunities efficiently is it important to appropriately group the various maintenance tasks [12, 21, 22]. Due to the immense competition in the aviation industry, valuable information about cost optimization in aircraft maintenance is not shared among companies. Therefore, as expected, literature on the specific topic of clustering aircraft maintenance tasks has been proven to be very limited. Just two papers were especially written to cover this topic. The paper of Muchiri and Klaas [23] developed a model to group aircraft maintenance tasks into manageable packages. Anthony Muchiri, one of the authors of the paper, said that it would be a good idea to give tasks, having the same access panels, a unique parameter. This parameter can later be used in the consideration of grouping tasks together (Anthony Muchiri, personnel communication, June 30, 2018). The other paper, discussing this topic, is the more recent paper written by Pereira and Babu [4] which grouped aircraft maintenance tasks based on the requirement of using the same access panel. Maclein Pereira, one of the authors of the paper, said that the purpose of the paper was to develop a simple tool that would allow maintenance engineering to combine tasks in such a way that it would eliminate repetitions. He started by selecting a particular system (e.g. ATA 29 hydraulic system) and created a database of all the tasks for that system. Then, when different tasks were put into the tool, they would get grouped based on access panels to be opened. He also added a feature where you segregate tasks zone wise and threshold wise (Maclein Pereira, personnel communication, June 3, 2018). These two papers form the basis for the factors influencing the clustering of aicraft maintenance tasks. Additional papers mentioned in this subsection were not especially written for this topic; however, they did indicate factors for clustering of aircraft maintenance tasks.

Various papers mention that opening and closing of access panels should be avoided as much as possible. This time consuming task is necessary to perform a maintenance task on most aircraft systems and components. In most circumstances it is possible to maintain more than one system or component by operating the same access panel [4]. Grouping these maintenance tasks together leads to various advantages such as: reduction in time [4] and reduction in wear and tear on the panels and fasteners [4, 23]. When looking at the content of the MPD, in this case for the A320, it becomes evident that there are more tasks which require the same preparation. Examples of such preparations are aircraft jacking and main landing gear removal, flaps extended and slats fully extended (figure 2.1), removal of the seats and electrical power off.

A318/A319/A320/A321 MAINTENANCE PLANNING DOCUMENT

TASK NUMBER	ZONE	DESCRIPTION	THRESHOLD INTERVAL	SOURCE	REFERENCE	MEN	M/H	APPLICABILITY
275446-02-1	530 630	FLAP TRACKS AF DI DETAILED INSPECTION OF TRACKS AND ROLLERS (AS FAR AS VISIBLE) PREP. : FLAPS EXTENDED; SLATS FULLY EXTENDED;	I: 22000 FH	MRB 6,9	275446-200-001 MRB REFERENCE : 27.50.00/09	1 1 *	0.30 0.30 0.04	ALL
275446-03-1	530 630	FLAP TRACKS AF DI DETAILED INSPECTION OF SPHERICAL BEARING INSTALLATIONS AT FLAP TRACKS 2, 3 AND 4 IN ACCORDANCE WITH SB 57-1027 PREP. : FLAPS EXTENDED AT 20 DEG.;	I: 6000 FH	ISB	275400-210-003 ISB 57-1027	1 1 *	0.15 0.15 0.02	A320 PRE 21586 (57-1026) PRE 21999
275449-01-1	575 675	ACTUATOR ASSEMBLY AF CHK CHECK FLAP ROTARY ACTUATOR ASSEMBLY AT TRACK 2 FOR EVIDENCE OF WATER INGRESS. PREP. : FLAPS EXTENDED; ACCESS: 573AB 673AB	I: 2400 FH	ISB	275400-220-002 ISB 27-1067 ISB 27-1076	1 1 * * *	0.50 0.50 0.02 0.02 0.02	A320 PRE 24337 (27-1077) OR A321 PRE 24337 (27-1091)
275449-02-1	575 675	ACTUATOR ASSEMBLY AF CHK CHECK FLAP ROTARY ACTUATOR ASSEMBLIES AT TRACKS 1, 3 AND 4 FOR EVIDENCE OF WATER INGRESS. PREP. : FLAPS EXTENDED; ACCESS: 575BB 575CB 575FB 575GB 675BB 675CB 675FB 675GB	I: 40 MO	ISB	275400-220-001 ISB 27-1067 ISB 27-1076	2 2 * * *	1.00 1.00 0.02 0.08 0.08	A320 PRE 24337 (27-1077) OR A321 PRE 24337 (27-1091)

Figure 2.1: Example of tasks requiring the same preparation obtained from [10].

Another important factor is the maintenance task due date, indicated by flight hours, flight cycles or time [5, 23]. The paper of Li *et al.* [21] used the basic properties of maintenance tasks which includes ATA code, maintenance interval, zone, check and man-hours to determine which tasks should be combined into a work package. The properties ATA code, maintenance interval, zone and check were utilised to measure any similarities between the maintenance tasks. Each of them was given a different weight to quantify for the differences in roles and influence. Based on engineering experience the weighting factor for the maintenance interval was set at 0.8, much higher than the others. The importance of maintenance intervals is also described by Muchiri and Klaas [23], where the maintenance interval is solely used to cluster task.

Estimations reported by the industry of aeronautics indicate that maintenance activities correspond from 10% to 20% of an operator's direct operating cost [1]. Therefore, cost should also be a major driver in the decision of grouping tasks. In Muchiri and Klaas [23] the cost was used as an performance indicator from which conclusions were drawn. The results of the optimisation method in the paper of Hölzel *et al.* [12] indicate that the best results were obtained when sorting the tasks by cost in descending order. This way the optimiser tried to allocate the most expensive tasks to maintenance opportunities closest to the Remaining Useful Life (RUL) of the component. A paper from the railway industry written by Sriskandarajah, Jardine and Chan [24] clearly explained the effect of scheduling a task earlier than its due date. In the paper an example was given of a CO-check, for train cars, which was maintained one month earlier than its due date. According to the paper this resulted in an additional 7961 man hours for the entire life time of the train car.

It is important to understand the nature of all the tasks. For instance when considering a painting job on an aircraft, it is absolutely forbidden to simultaneously execute tasks that require electricity [11].

Therefore, a throughout understanding of the properties of each task is essential information required for clustering tasks.

2.1.3. Discussion

Main factors found in this section, for clustering aircraft maintenance tasks, are tasks which require the same preparation (e.g. access panels), task interval due dates, and cost associated with a specific task. It was also mentioned that some tasks may not be clustered together because of safety reasons. Therefore, it seems that a thoroughly understanding of the properties of each tasks is required for this research. At the current state of this research an excel sheet is provided with the following task information: MPD task number, a task description, threshold intervals, and estimation of the required man-hours. Given the MPD tasks numbers it is possible to find the tasks which require the same preparation. However, this will most likely be a manual process, which will be a burden to do for each of the approximately 2500 different tasks. Since cost information per tasks is classified such data will not be available for this research. However, it is possible to make a distinction between task cost based on the required man-hours. The paper of Muchiri and Smit [25] confirms this by writing that the total man-hour cost is a good indicator of other cost related to maintenance tasks (i.e tasks requiring more man-hours are more expensive in terms of cost associated with executing that specific tasks). Anthony Muchiri, one of the authors of the paper, said this statement is based on experience and data collected over the years. He worked at an airline where it was practice to take nominal hours suggested by the maintenance manual, and multiply them by a certain factor. The outcome multiplied by the labour rate, served as a good indicator for the total cost of maintenance. This was proven by taking the actual maintenance cost for each fleet type, and dividing it by the total maintenance hours (i.e. maintenance downtime). The difference was quite small for each fleet type, hence the concept was proven. Important to note is that the total cost referred to here covered labour and materials. The difference between the calculated value and the actual value could be attributed to sever non-routine maintenance (e.g. external damage, and not normal wear and tear) (Anthony Muchiri, personnel communication, June 30, 2018). The last main factor, task interval due date, is present in the excel sheet and can therefore be used. Additional practical knowledge could be valuable in deciding which tasks to cluster, however this information is not available. Nevertheless, this will have no impact on the potential of the to be developed model. As explained non-routine maintenance can have a major impact on a long-term maintenance schedule. Therefore it is of utmost importance to take these unexpected maintenance into account. Due to the stochastic nature of these events it becomes hard to plan them, current techniques that try to estimate non-routine maintenance are based on historical data. As this research has no access to such data additional margins are the only acceptable choice. Any required assumptions following from this section are stated in 3.4.

2.2. Preventive maintenance scheduling in the aviation sector

A very interesting report is the one written by Vanbuskirk *et al.* [26], which presents a two stage system that support chiefs with allocating the aircraft to missions and scheduling the maintenance activities on aircraft. The first stage assigns the planes to forecasted flight operations by using a custom-built, multi-level greedy search algorithm. The second stage schedules all preventive maintenance activities and is implemented as a constraint satisfaction problem. That is solved with a search engine. The system was tested with approximately 17 jets, and results indicate that a planning horizon of 3 months was solved within 19 minutes and 58 seconds. Whereby 3750 maintenance actions were scheduled for 17 Harrier aircraft. Karsai Gabor, one of the authors, said that the goal was to schedule the activities, given various constraints: calendar-based actions had to be done within a time window, usage based actions had to be done when the usage clock on a part/subsystem reached a certain value, personnel had to be available to do the job, mechanics can only do jobs that they are qualified for, maintenance jobs had to be inspected by a quality/safety inspector, et cetera. No optimization was done since the support for the flight schedule was the top priority. The model was solved with Mozart/Oz programming language which has a built-in support for constrained-orientated programming (Karsai Gabor, personnel communication, June 27, 2018).
The paper of Steiner [2] presents a heuristic method for the scheduling of preventive aircraft maintenance. The objective of the research was to minimize the overall number of maintenance actions and uniformly distribute the capacity and flying hours over time. Even under heavy constraints the pro-

posed algorithms have shown to work reliable, fast and with good optimization results. According to the author the time to compute a new maintenance plan is within the range of 5 to 15 minutes. Albert Steiner, the author of the paper, said that the algorithms used in the paper were not based on any known approaches. An aspect was to split the overall process into sub-processes that could be handled computationally at the same time. Determining the optimal position of the maintenance actions was the least difficult one, whereas the balancing step was the most challenging one. Steiner said that if he would need to develop this kind of tool again, he would surely consider combining some of the core ideas of his approach with state-of-the-art operations research methods (e.g. the ones mentioned in this report), which he thinks would be possible. The number of tasks scheduled per fleet were in the range of around 50 to 500, with a time horizon of five-year (Albert Steiner, personnel communication, July 22, 2018).

Kleeman and Lamont [27] investigated the use of a Genetic Algorithm (GA) to solve the maintenance scheduling problem of aircraft engines. The paper had two specific objectives, namely, to minimize the time needed to return an engine and to minimize the cost. The research confirms that the maintenance scheduling problem of aircraft engines can be effectively solved with the proposed algorithm.

Safaei, Banjevic and Jardine [28] dealt with the issue of maximizing the availability of a fleet of military aircraft. Pre- or after-flight inspections on military aircraft determine if components require maintenance at the repair shop. The objective is to schedule the required maintenance jobs in such a way that sufficient military planes are available for the next mission. In this problem the primary constraint or resource is the workforce. The problem was formulated as a Mixed-Integer Linear Programming (MILP) problem and was solved by using a branch-and-bound method.

Heavy aircraft maintenance facilities must ensure that critical equipment remains available by scheduling preventive maintenance tasks. Often the workforce in these facilities are high-paid, meaning that worker idle time should be as low as possible. The problem consists of two conflicting objectives, namely, the optimal required workforce needed and the minimization of the completion time. Quan, Greenwood, Liu and Hu [29] used evolutionary algorithms to solve this Multi-Objective Problem (MOP) of preventive maintenance scheduling. Many other papers [30–32] use the weighted-sum to transform the MOP to a single-objective function. However, it remains difficult to assign meaningful weights to the MOP, which requires knowledge of the behaviour of each objective function [29]. It is worth mentioning that there is another popular strategy called the Pareto-optimal approach [33, 34]. This approach uses dominance to pick better solutions. Quan *et al.* [29] distinguish itself by using preferences among solutions to guide the search. Advantage of this approach is that it is possible to target specific subsets of the Pareto front. Results of executing the proposed method on a large problem indicate that found solutions are better aligned with the manager's expectations.

The planning of aircraft maintenance can be seen as a variation of the bin packaging problem, in which maintenance opportunities represent bins that need to be filled with their respective preventive maintenance tasks. Since the bin packaging problem is a well known NP-hard problem [35–37], an exact algorithm is not able to solve it in polynomial time. However, it is possible to solve NP-hard problems heuristically in polynomial time with the use of branch-and-bound algorithms [38]. The paper of Hölzel *et al.* [12] describe the program flow of the Aircraft Maintenance Planning (AIRMAP) optimizer module which is based on a heuristic approach with a branch-and-bound algorithm. The end results of the AIRMAP module is a maintenance schedule with the lowest overall cost. By controlling the variables of RUL, task duration and tasks costs it is possible to simulate alternative aircraft maintenance strategies.

2.3. Related problems in literature

Prior to this thesis research a literature study was conducted to find relevant related problems to this research. Similar problems found in literature are the job shop scheduling problem, bin packaging problem, master bay plan problem, knapsack problem, resource-constrained project scheduling problem, and the resource allocation problem. The problems covered in this section are the job shop scheduling problem, bin packaging problem and knapsack problem. The remaining problems can be found in the full literature study [6].

2.3.1. Job shop scheduling problem

Scheduling a set of preventive maintenance task with a known workforce can be seen as a variation of the Job Shop Scheduling Problem (JSSP) [39, 40]. The most basic version of the JSSP is formulated as the case whereby a set of n jobs need to be scheduled on a set of m machines while minimizing the makespan (i.e. total amount of time it takes to complete all tasks). Each of the jobs consists of a specific set of operations, which need to be addressed in a specific order [41]. An extension of the classical JSSP is called the Flexible Job Shop Problem (FJSP) which takes into account the production flexibility [40]. Dissimilar to the JSSP each operation in the FJSP can be processed on any of the machines. The FJSP cover the machine assignment problem and the operation sequencing problem [41, 42]. Various papers indicate that the JSSP is a non-deterministic polynomial-time (NP) hardness problem (i.e. the problem cannot be solved within a deterministic polynomial time) [29, 43, 44]. Since the FJSP is a more complex version of the JSSP it is clearly NP-hard [40, 42]. Solving approaches for the JSSP fall in two categories exact methods or approximation techniques [45]. Most of the FJSP concentrate on a single objective. However, recently more attention has been given to the multi-objective variant [42]. Unlike most single objective problems, multi-objective problems usually have a set of optimal solutions called Pareto optimal solutions [46]. In this research the scheduling of jobs on maintenance opportunities can be seen as a variation of the JSSP where the maintenance opportunities represent machines. Also the aspect of multi-objective FJSPs could be of interest for this research since the problem at hand is also multi-objective. This concludes that the JSSP and its extension, FJSP, are interesting problems which could be useful for this research.

Exact algorithms

Since the FJSP is NP-hard the only reason for using exact algorithms is to obtain solutions for small or medium-sized problems. These solutions can then be used for comparison with approximation techniques [40, 47]. Approximation techniques for searching optimal solutions can be applied to exact algorithms, nevertheless they deliver limited computational potential for NP-hard problems [40].
In literature the exact algorithms commonly proposed, for solving the multi-objective FJSP, are brand-and-bound algorithms [48] and mixed-integer programming formulations solved by using a solver [40]. Fattahi *et al.* [49] introduce a Mixed Integer Linear Programming (MILP) formulation for the FJSP. This mathematical model was used to find the optimal solution for small-sized problems. The problem was coded by lingo software which solves the model by using the branch-and-bound method. In order to solve real sized problems they used two heuristic approaches which were based on simulated annealing and tabu search. Özgüven *et al.* [50] developed a MILP for the FJSP which was solved by using CPLEX, and compared it with the model of Fattahi *et al.* The results of the sample test problems indicated that the model of Özgüven *et al.* was superior compared to the model of Fattahi *et al.* Another MILP model was introduced in the paper of Mati and Xie [47] which used the optimization engine XPRESS to solve the model. In the three previous mentioned papers the models had the same objective function, namely, minimizing the completion time of all tasks. Besides these single-objective optimization problems also literature was found on solving the multi-objective FJSP. In the paper written by Khalife *et al.* [51] three objectives are combined into a single objective function by using the weighted sum method. The MILP model is then solved by using the branch-and-bound algorithm. Thörnblad *et al.* [52] also used a weighted sum method to combine their three objectives into a single objective function. The proposed model was solely defined in time-index variables. Although the model was able to produce optimal schedules for real applications within reasonable time, the problem size (i.e. maximal number of jobs used was 35) is still too small for this research problem. Another special kind of optimization technique is goal programming (i.e extension of linear programming) which was used in [53] to solve the multi-objective optimization by minimizing total cost, minimizing total lead time and maximizing the total quality.

Approximation techniques

Approximation techniques can yield optimal or near-optimal solutions for NP-hard problems within a reasonable amount of time [48]. Evolutionary Algorithms (EA) are mentioned in literature to be effective in solving a wide variety of NP-hard optimization problems [29]. There are two major paradigms of EA, Genetic Algorithms (GA) and Evolution Strategies (ES) [39]. According to Kalyanmoy Deb, a highly cited researcher specialized in multi-objective optimization and EA, an EA is a perfect choice for solving multi-objective optimization problems [54]. For solving complex real-world problems it is recommended to start, if possible, with a customized initialization to save computational time [54].
Available literature shows promising results of using evolutionary algorithms in scheduling preventive maintenance tasks. In [39] a study has been conducted about applying EA to preventive maintenance

scheduling. The results support practical utility in solving large-scale, complex preventive maintenance problems by using evolutionary algorithms. Li *et al.* [42] proposed a collaborative evolutionary algorithm based on Pareto optimality to solve the FJSP. The FJSP considered was multi-objective, namely, optimizing the makespan and the balance of the workload of each machine. The paper claims that the used approach (i.e. Pareto optimality) is better than the weighted-sum method in terms of finding more and better Pareto solutions.
Many of the papers found use a combination of various approximation techniques to find near-optimal solutions. Among them is the work of Meeran and Morshed [55] which present a solution method that combine GA and Tabu Search (TS). The rationale behind using this combination was to use the global parallel search capabilities of the GA and the local optimum avoidance capabilities of the TS. Xia and Wu [56] proposed an approach which used Particle Swarm Optimization (PSO) to assign operations to machines and a Simulated Annealing (SA) algorithm was used to schedule the operations on the machines. The results, obtained from a computational study, indicate that the proposed algorithm is an effective approach for solving multi-objective FJSPs.

2.3.2. Bin packaging problem

The most basic version of the Bin Packaging Problem (BPP) is formulated as the case whereby given a finite set of C bins and a set of n items, which sizes are smaller than C, pack the items into a minimum number of bins whereby the sum of the sizes in each bin does not exceed its respective capacity C [57]. Various literature indicates that the BPP, like all other related problems in this section, is NP-hard [35–37]. Therefore, only approximation algorithms are suitable for solving large-scaled problems. In the context of this research bins can be seen as maintenance opportunities which need to be filled with their respective items (i.e. preventive maintenance tasks) [12]. This concludes that the BPP is similar to this research objective; therefore, the solution techniques used for the BPP are examined.

On-line algorithms

In the BPP on-line algorithms, pack the items in the order they are detected which means that the complete list of items is not known prior to packaging [57, 58]. This is somewhat conflicting to our research objective; however, it must be noted that unscheduled maintenance is also not known prior to packaging.
A classical approach for the BPP is to pack the bins each at a time according to the Next-Fit (NF) approach. In this approach the first item is packaged into a bin, where after each successive item is put into the same bin till it is completely full. Whenever the bin is full or it is not possible to put the item in the current bin, the bin is closed and the item is allocated to a new empty bin. The above mentioned procedure is executed till all items are allocated or all bins are full. A major disadvantage of the NF approach is that it closes bins that could potentially hold later identified items [57].
In the Worst-Fit (WF) approach if the current item fits into the bin with the lowest nonzero level it is placed in there. Otherwise, the item is placed into a new empty bin [59]. A modification of the WF approach is called the Almost Worst-Fit (AWF), hereby the current item is placed in the bin with the second lowest content. Only if the current item fits nowhere else it is placed in the bin with the smallest content [57].
Two other classic approaches are the First Fit (FF) and Best-Fit (BF) approaches. In the FF approach the current item is assigned to the leftmost nonzero bin capable of fitting the item. If such a bin is not available the item will be put into a new empty bin [59]. The BF approach puts the item in the fullest bin that has the required capacity left for the current item. As with the other approaches if no capacity is available a new bin will be opened [36].

Off-line algorithms

In the BPP the off-line algorithms know the full list of items to be packed prior to the packaging process [57]. This complies with our research wherein all preventive maintenance tasks are known beforehand. The off-line algorithms mostly sort the specific list of items and then use a well-known classical on-line algorithm.
A sorted list packed according to the NF approach is called the Next-Fit Decreasing (NFD) algorithm. Baker and Coffman [60] proved that the Asymptotic Performance Ratio (APR) (i.e. the ratio between the amount of bins used by an optimal algorithm and the proposed algorithm) of the NFD is slightly

better than the FF and BD approaches. Sorted list packed according to the FF or BD approaches are called respectively First-Fit Decreasing (FFD) and Best-Fit Decreasing (BFD). Both the FFD and BFD approaches have a much better APR compared to their respective on-line algorithms [61–63].
Various other attempts were made to improve the APR. The paper of Garey and Johnsen [64] presents the Modified First-Fit Decreasing (MFFD) algorithm. The MFFD algorithm first reorders the list of items according to their size, just as has been done with the FFD. In addition to this each item is given a specific type, which depends on the size of the item. For example group A is defined as items with sizes that fall within $(\frac{1}{2},1]$ of the bin size. The MFFD starts with assigning these A items to the available bins. Then the algorithm proceeds through the A-bins from left to right and if any unpacked B item (i.e. item with the size $(\frac{1}{3},\frac{1}{2}]$) fits within a specific bin it is placed there. This procedure is repeated for the other item categories. In the last step of the algorithm the FFD approach is utilized to pack any remaining items which did not fit in the available bins. The paper proves that the MFFD algorithm guarantees an APR of $\frac{71}{60}$.
Dokeroglu and Cosar [37] proposed eight different parallel hybrid Grouping Genetic Algorithms (GGAs) for solving the one-dimensonal BBP. According to the paper these algorithms take advantages of the parallel computation techniques and obtain solutions for large-scale BPP instances. A special property of the algorithms is that the initial population is partly produced by the BFD and FFD heuristics.

2.4. Mixed-integer linear programming

Mixed-Integer Linear Programming (MILP) is an exact method capable of finding an optimal solution to an optimization problem when solved by a solver (e.g. CPLEX, Gurobi). Formulating a problem into a MILP is mostly straightforward. It is, however, important to take care in constructing the MILP since certain formulation attributes can have an impact on the effectiveness of a LP-based solver [65]. Almost all MILP solvers are based on branch-and-bound algorithms [66]. Main ingredients of MILP solvers include, preprocessing phase, cutting plane generation, sophisticated branching strategies, primal heuristics, and parallel implementation [67]. There are multiple free MILP solvers available (e.g. BLIS, GLPK, lp_solve) as well as more powerful solvers that require a commercial license (e.g. CPLEX, Gurobi, Lindo). Since most of the problems covered in this literature review are NP-hard the MILP solvers, especially the non-commercial solvers, will most likely only be capable of solving small to medium sized problems. In the literature study [6] conducted, prior to this research, MILP formulations were solved by commercial/non-commercial solvers in the following papers [40, 47, 49, 50, 68, 69]. Table 2.2 shows an overview of other solution techniques found during the literature study including their definition, main limitations and relevant papers.

2.5. Discussion literature review

Based on the conducted literature review it is recommended to first formulate the research problem into a MILP formulation. After which a commercial solver (e.g. CPLEX, Gurobi) can be used to solve the problem. If the problem is solved quickly, by a commercial solver, it is recommended to further test the model capabilities in more restricted cases. In the end the model can be further improved by decomposing the problem (e.g. clustering of tasks) or using special techniques (e.g. row generation, column generation). The solution of the exact method can be used as reference solution for the to be designed approximation algorithm. Most of the found literature, during this literature review, used a hybrid algorithm consisting of features from different approximation algorithms. It is recommended to try and formulate the task allocation problem as one of the found related problems (e.g. bin packaging problem). After which a combination of known solution techniques, for the related problem, can be used to solve the task allocation problem. Since long-term maintenance plans are affected by daily operations, that lead to unscheduled maintenance required for the safe operation of aircraft, it is required to be able to reoptimize the maintenance task schedule at an operational level (i.e. a short-term time horizon of a couple of maintenance opportunities). Therefore, a model framework is required that enables an optimization at the long-term horizon (i.e. tactical level) and an optimization at the short-term horizon (i.e. operational level). This concludes the literature review, which may be seen as preparation for obtaining the task packaging model applicable to aircraft maintenance.

Algorithm	Definition	Main limitations	Relevant papers
Genetic algorithm	Algorithm based on concepts of evolutionary biology. Concepts of selection, mutation, and cross over are used to generate new populations.	Optimum parameter settings unknown [70], premature convergence [71], for complex problems a great computational effort required [72], no guarantee that the solution found is optimal.	[24, 27, 37, 55, 73, 74].
Mixed-integer linear programming	An exact method capable of finding an optimal solution by using a solver.	High amount of computational effort required for solving large-scale problems, commercial solvers have the best performance.	[40, 47, 49, 50, 68, 69].
Tabu search	A metaheuristic that continues the local search, whenever it reaches a local optima, by allowing moves that do not improve the solution.	Large number of iterations, a lot of parameters to be determined, no guarantee that the solution found is optimal.	[42, 55, 69, 75–77].
Ant colony optimization	An algorithm that is based on the behaviour of real ants. The problem is formulated as searching for the minimum cost path in a graph.	Time to convergence is uncertain, research is experimental [78], no guarantee that the solution found is optimal.	[46, 76, 79–81].
Evolutionary algorithms	Algorithm that imitate biological principles of adaptive selection inspired by nature.	Same limitations as the genetic algorithm.	[29, 39, 42, 82, 83].
Branch-and-bound algorithm	An algorithm that iteratively builds a search tree of subproblems and is capable of finding the exact solution.	Large-sized problems create a very large tree, solving large-scaled problems require a large amount of computational effort.	[12, 28, 48, 51].
Simulated annealing	A randomized local search algorithm based on the annealing process of solids.	May require to much computational time [84], tailoring work required for the different classes of constraints and fine-tuning the parameters of SA can be rather delicate [85], and there is no guarantee that the found solution is optimal.	[51, 56, 86].
Particle swarm optimization	An algorithm that mimics the behaviour of flying birds and the way they exchange information.	The algorithm is not able to work out the problems of scattering and optimization [72], and there is no guarantee that a global optimal solution is found.	[56].
Constrained orientated programming	Basic idea is that the user states the constraints upon which a solver is used to find a feasible solution [87].	Requires languages that support constraint programming or have specific libraries.	[26].
Goal programming	Some sort of linear programming.	Difficult to set the appropriate weights [88], no guarantee that the obtained solution is Pareto optimal [89].	[53].

Table 2.2: Overview of the found solution techniques during the conducted literature study obtained from Witteman [6].

3
Research definition

3.1. Problem statement

Current approaches in aircraft maintenance scheduling is to utilize the maintenance opportunities efficiently by appropriate grouping of maintenance tasks. This act of grouping maintenance tasks is called task packaging. The packaging of tasks allows an efficient use of maintenance opportunities but leads to waste of aircraft component life when tasks are performed before its due date. This approach is an inefficient and time-consuming process that needs to be improved in order to enhance operational efficiency. Long-term maintenance plans are affected by deviations from flight plans or unforeseen events. This causes the need of frequently modifying or redoing the maintenance schedule [26]. The problem described is twofold. Firstly, the creation of a maintenance schedule is time consuming and may need to be redone multiple times depending on deviations from flights plans or unforeseen events. Secondly, current approaches are not capable to deliver a maintenance schedule for restricted cases (i.e. cases whereby available man-hours significantly constrain the problem). The goal should be to develop a maintenance tasks packaging model that automatically generates aircraft maintenance schedules which should be be better as the ones made by experienced schedulers.

3.2. Research framework

The objective of this research is as follows:
To develop a maintenance tasks packaging model for all the tasks required to ensure ongoing airworthiness of the aircraft by developing a framework capable of delivering optimized schedules for the allocation of maintenance tasks to the available maintenance opportunities.

In order to achieve the research objective, two main research questions were stated. The main research questions are further split into sub questions, which together will provide the answer for each corresponding main research question.

1. How can the modelling framework be designed?
   (a) What are the main functions required for the model?
   (b) What solution technique(s) will be used to solve the model?
   (c) What are the results of simulating the model with data from an airline?
2. Is the final framework capable of delivering better quality schedules compared to the current approach?
   (a) Do the results of simulating the framework deliver the required features?
   (b) Is the final framework acceptable in terms of computational time, optimization and assumptions?

3.3. Model requirements

Achieving the research objective is only possible if the model requirements are known. In collaboration with the European airliner, supporting this research, the following model requirements were set:

- All inputs to the models are provided in Excel files.
- The computational time of the tactical level models should be less than an overnight (i.e. approximately 12 hours). The computational time of the operational level models should be less than 10 minutes.
- In order to not impose financial borders it is desired to deliver a model that runs on free software.
- The output of the model needs to be verified and validated.

These requirements will be incorporated in each of the models, which are programmed in Python 3.7 and were solved on an Intel Core i5 1.6 GHz laptop with 4GB ram.

3.4. Scope of the research

Narrowing the scope of this research is important for creating good research questions. The main focus should be on the development of the tasks packaging model. In order to keep the main focus, assumptions are necessary for problems not directly related to this research objective. The assumptions mentioned in this section are valid for this research. However, it must be mentioned that during this research some of the assumptions were eliminated. Overall, a model that have fewer assumptions is more practicable for companies. Limitations for this research also lead to additional assumptions.

This research is subject to the following **limitations/challenges**:

- To not impose financial borders for companies it is preferred to not use commercial solvers in the final model. Therefore, if possible only free software is used, such as the high-level programming language Python. However, it must be noted that commercial software, such as CPLEX or Gurobi, can be used during this research for assessing the quality of the final model.
- Due to the high competition in the aviation industry, it is very unlikely that any cost related information about tasks can be obtained.

This research is subject to the following **assumptions**:

- Workforce
  - It is assumed that estimated man-hours per task, as given in the provided excel sheet, is a realistic representation of the actual time required to finish the task.
  - It is assumed that engineers produce eight effective hours per workday.
- Available data
  - It is assumed that total man-hours costs is a good indicator of other cost related to maintenance tasks (i.e. tasks requiring more man-hours are more expensive than tasks with less man-hours) [25].
  - It is assumed that different tasks can be executed simultaneously on the same aircraft.
- Maintenance
  - In order to obtain a feasible maintenance schedule it is necessary to take into account non-routine maintenance. The non-routine rates are provided by the airline and are assumed to be a realistic representation of the expected non-routine rates during maintenance.
  - It is assumed that there are sufficient aircraft spare parts, maintenance tools and equipment available.
- Location
  - It is assumed that maintenance ties up only one single hangar. Therefore maintenance is always executed at the same location.

3.5. Relevance of the study

The findings of this study redound to the benefit of maintenance organizations and researchers considering that not much information is available on such a specific topic. This research is the first to present an optimization model to solve the maintenance task packaging problem, being addressed at the work shift level. Furthermore, it presents a bin-packaging approximation algorithm that can be solved within minutes for the 45 aircraft fleet and a planning horizon of 4 years. The framework was validated in practice and the results give insights on the potential benefit of using such framework to solve this daily recurrent problem.

4
Research strategy

4.1. Data

The input data for the model is obtained from an airline which worked in close cooperation with this research. The used data consists of five different data files: tasks information, maintenance opportunity dates, amount of technicians, monthly utilization of each aircraft and non-routine rates. These files will all be thoroughly explained in the upcoming subsections.

4.1.1. Task excel sheet

The main input for this research is the tasks information given for each aircraft in the fleet. These data files will be more or less the same for each of the different aircraft, and therefore the data from one aircraft is analyzed. The task excel file contains of a detailed task description for each of the A-check or C-check tasks present in the excel file. The main information per task is its interval limit, last executed date, required man-hours, skill types required, and a task reference number. When looking at the data it becomes evident that a task can have up to three different limits, namely, a Flight-Cycles (FC) limit (figure 4.1), a Flight-Hours (FH) limit (figure 4.2), and a Calendar-based (CAL) limit (figure 4.3). The different colours in the figures indicate if there are multiple intervals for the specific task, as indicated by the legend of the respective figure. Depending on the planning horizon considered some tasks need to be rescheduled multiple times and others do not need to be scheduled at all.

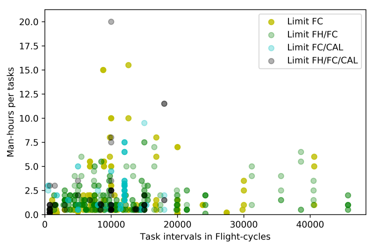

Figure 4.1: Visualization of tasks having at least a flight-cycle limit.

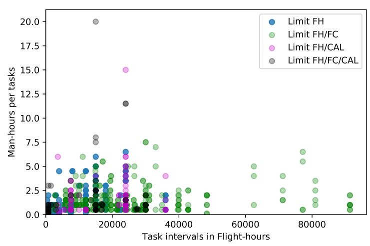

Figure 4.2: Visualization of tasks having at least a flight-hour limit.

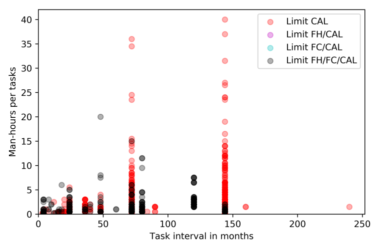

Figure 4.3: Visualization of tasks having at least a calendar-based limit.

4.1.2. Maintenance opportunity dates

The maintenance dates are generated by a dynamic programming based approach as described in the paper of Deng [90]. The approach aims at minimizing the total unused flight hours of a fleet and considers aircraft type, status and operational constraints. The result is an optimized A- and C-check schedule for each aircraft. The excel file used in this research contains the optimized A- and C-check schedule for the entire fleet. An A-check maintenance opportunity will take one entire day, while C-check maintenance opportunities take multiple days. The maintenance schedule is generated in such a way that concurrent scheduling of C-checks are limited to three and A-checks to two. The available man-hours, on the overlapping days, will need to be shared among the different aircraft. Figure 4.4

shows a visualization of the generated maintenance schedule for the fleet during the considered time horizon.

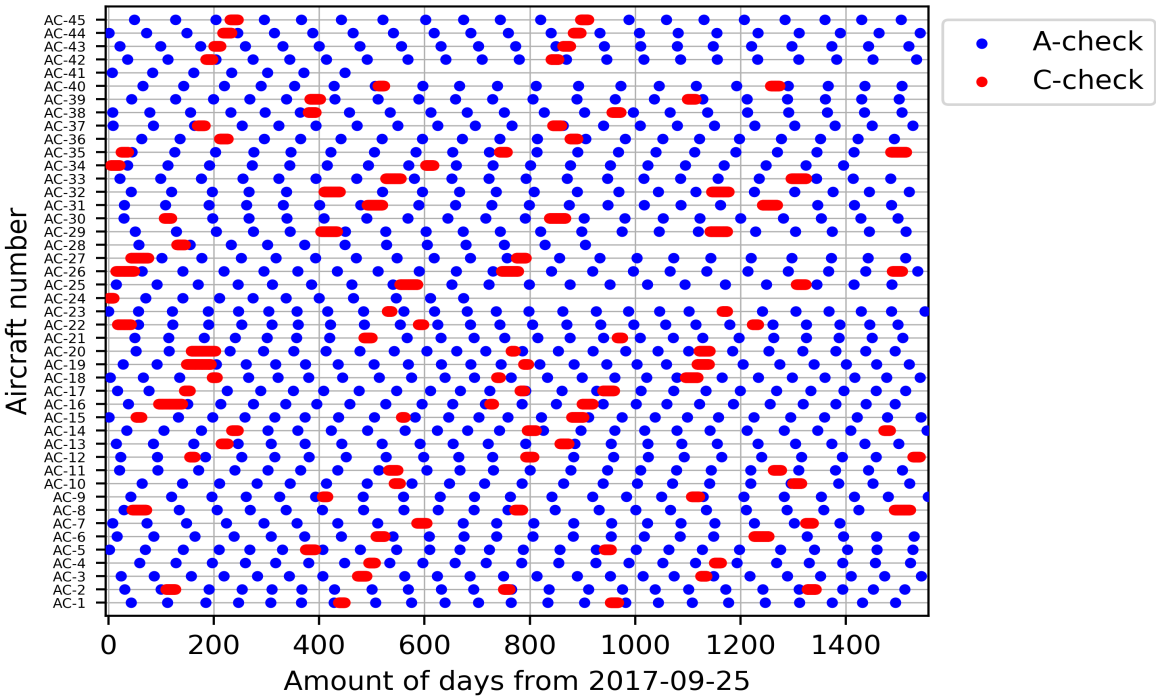

Figure 4.4: Scheduled maintenance opportunities fleet-wide.

4.1.3. Number of technicians

During this research historical data about the daily available technicians is used. The data comprises of the amount of daily working technicians for the following different skill types: GR1, GR2, GR4, ESHS, ICH, PINT, MAP, and NDT. By processing the data it is possible to estimate the available man-hours per skill type during each of the maintenance days. The data makes a distinction between Light Maintenance (LM) and Heavy Maintenance (HM) personnel. HM personnel is available to work on C-check maintenance opportunities and LM personnel is available to work at the A-check maintenance opportunities. Technicians will only work on weekdays, which means that no man-hours are available on Saturdays and Sundays. It is worth mentioning that official holidays are also classified as weekend days. The visualization of the average available man-hours per working day for LM is given in figure 4.5 and for HM in figure 4.6. Comparing these figures it becomes evident that heavy maintenance has

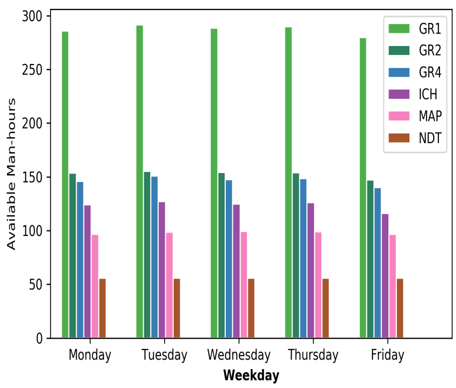

Figure 4.5: Daily available man-hours at LM.

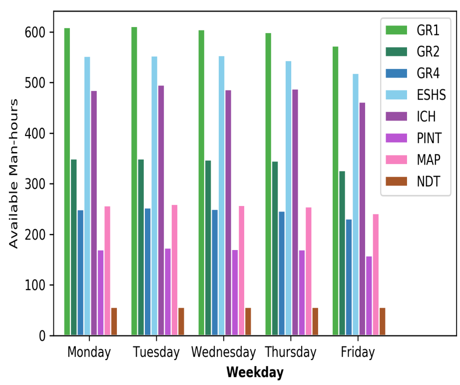

Figure 4.6: Daily available man-hours at HM.

two additional skill-types namely: ESHS and PINT. This means that any task requiring the skill-types ESHS or PINT need to be allocated to C-check maintenance opportunities.

4.1.4. Daily utilization aircraft

In order to calculate the due-dates of the given tasks, it is essential to have information about daily utilization of the aircraft. The data contains an average daily flight-hours and flight-cycles utilization for each month per aircraft. The average monthly utilization of aircraft 1 are visualized in figure 4.7. From this figure it becomes evident that there is a small seasonal pattern which needs to be taken into account during the simulation. A related point to consider is that the average utilization differs for each of the considered aircraft. These values will be used during the simulation to determine the due-date of tasks having a flight-hours or flight-cycles limit.

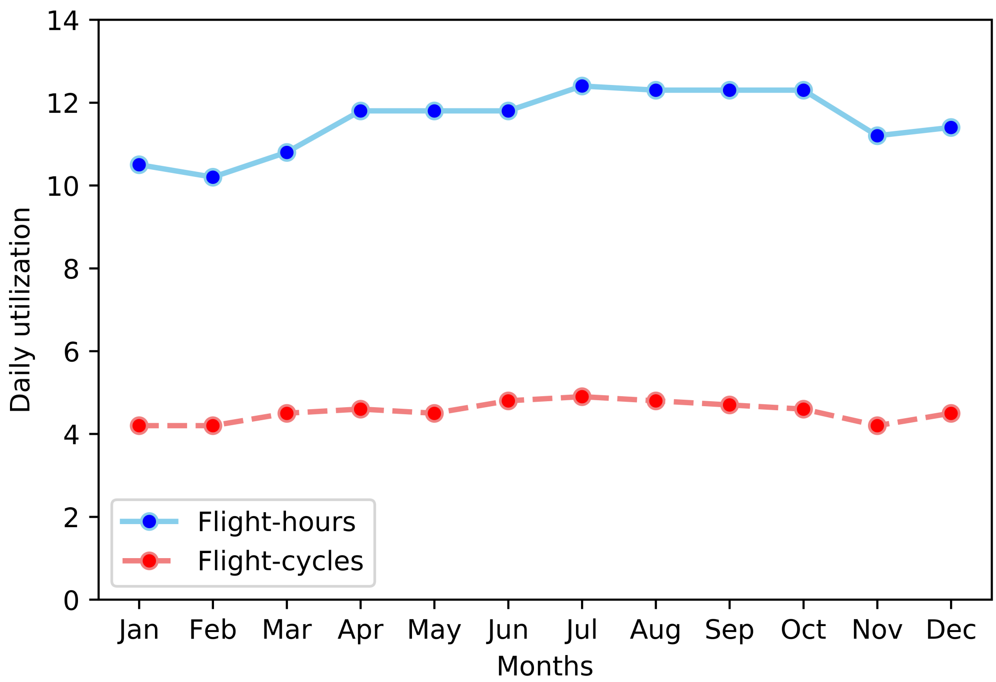

Figure 4.7: Average daily flight-hours and flight-cycles per month for aircraft 1.

4.1.5. Non-routine rates

In the literature review it was already noted that non-routine maintenance can have a major impact on a long-term maintenance schedule. Therefore, the airline provided non-routine rates, as given in table 4.1, that need to be incorporated into the models. The ratio columns indicate the non-routine rates that should be multiplied with the required man-hours of the task. After which the calculated non-routine man-hours should be added to the required man-hours of the considered task.

4.2. Model framework - high level perspective

This section presents the high level perspective of the model. The model consists of a two-staged approach capable of solving the task allocation problem for an entire fleet. The overall objective of the model is to utilise the maintenance opportunities efficiently by appropriately allocating maintenance tasks. The allocation of tasks should be done in such a way that remaining useful life of the tasks are as low as possible. Since long-term maintenance plans are affected by deviations from flight plans or unforeseen events. It becomes essential to have a model capable of delivering good results within a reasonable amount of time. Given the size of the problem it is chosen to have a two-stage approach. Consisting of an optimization on the tactical level and the operational level. The high level model framework for this research is presented in figure 4.8.

A-check tasks				C-check tasks			
Skill	Block	Skill	Ratio	Skill	Block	Skill	Ratio
GR1	INSP	GR1	0.18	GR1	INSP	GR1	0.65
GR1	INSP	GR2	0.01	GR1	INSP	ESHS	0.41
GR1	INSP	GR4	0.01	GR1	INSP	ICH	0.37
GR2	INSP	GR1	0.02	GR1	INSP	MAP	0.26
GR2	INSP	GR2	0.28	GR1	INSP	PINT	0.31
GR2	INSP	GR4	0.01	GR2	INSP	GR2	1.38
GR4	INSP	GR4	0.61	GR2	INSP	ICH	1.95
ICH	INSP	ICH	2.39	GR2	INSP	ESHS	0.6
ICH	INSP	MAP	0.05	GR2	INSP	MAP	0.86
				GR2	INSP	PINT	0.54
				GR4	INSP	GR4	0.83
				ICH	INSP	ICH	20.51
				ICH	INSP	ESHS	3.20
				ESHS	INSP	ESHS	1.62

Table 4.1: Non-routine rates per task category.

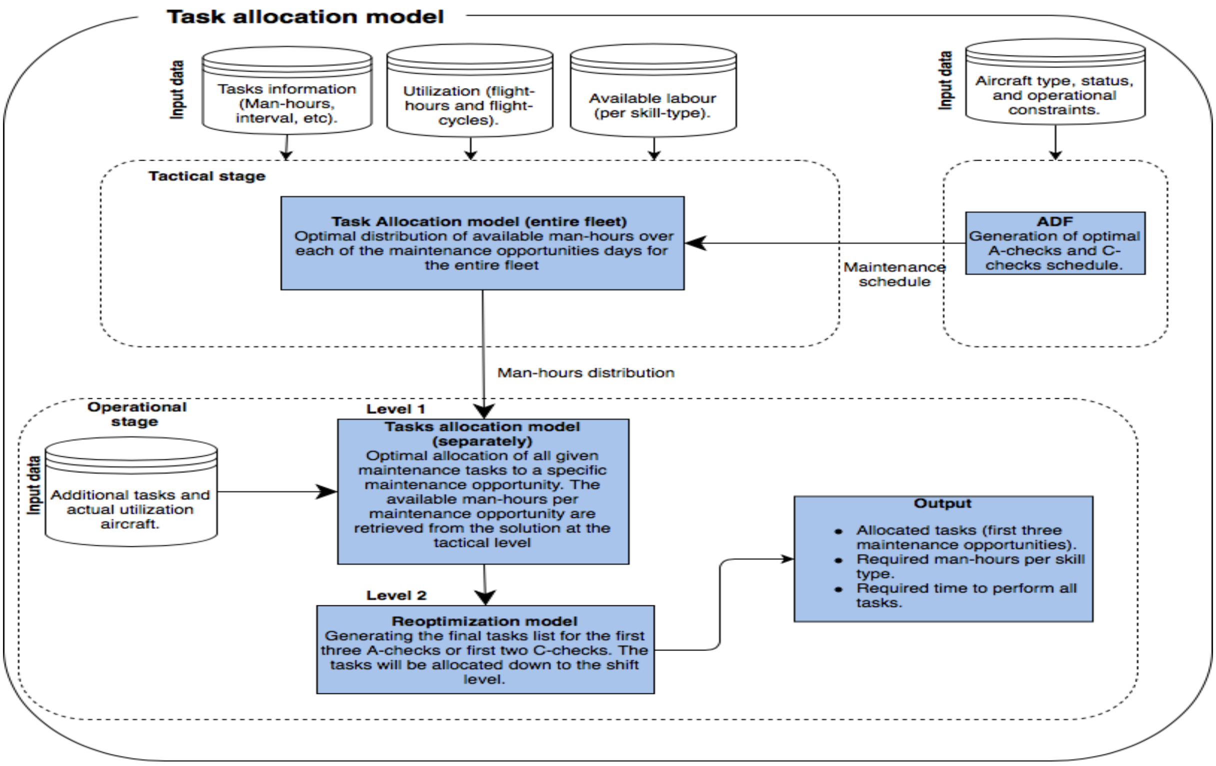

Figure 4.8: Model framework.

4.2.1. Tactical stage

At the tactical stage the problem will need to be decoupled for the operational stage. The outcome of the tactical stage is the optimal distribution of available man-hours over each of the maintenance opportunities days for the entire fleet. By having the available man-hours per maintenance opportunity days it is possible to transform the task allocation problem, at the operational level, to an independent problem for each aircraft considered. Main inputs required to obtain this output is the tasks information, aircraft utilization, available labour and the maintenance schedule for the fleet. Any significant changes in fleet size, average utilization of the aircraft, fleet modifications, non-routine maintenance tasks, and to the maintenance schedule require a new optimization at the tactical stage. For practical reasons it is decided to limit the computational time to 12 hours which is equivalent to an overnight.

4.2.2. Operational stage

Since the results of the tactical stage will transfer the interdependent problem to an independent problem it is possible to solve the task allocation problem, at the operational stage, for each aircraft separately, thereby reducing the complexity of the problem considerably. Main input to the model at the operational stage is the available man-hours for each of the maintenance opportunities days as obtained from the solution at the tactical level. The proposed operational stage model consists of two separate levels. At the first level the task allocation problem will be solved for the considered time horizon. By inserting the actual utilization of the aircraft it is possible to better estimate the due-date of each of the tasks. Furthermore, the first level model enables the addition of new non-routine maintenance tasks or new tasks that were previously not considered. At the second level the solution obtained from the first level will be used to determine the allocation of tasks down to shift level for the first three upcoming maintenance opportunities. Therefore, the outcome of the second level will be a distribution of tasks over each of the working shifts within the considered maintenance opportunities. It is expected that the operational stage models will need to be solved prior to any check for that reason the computational time of these models is set to a maximum of ten minutes.

I

Model formulation

5

Task allocation model - tactical stage

This chapter presents the modelling framework for solving the task allocation problem at the tactical stage. At the tactical stage the problem will need to be decoupled for the operational stage. The outcome of the tactical stage is the optimal distribution of available man-hours over each of the maintenance opportunities days for the entire fleet.

5.1. Exact method - tactical stage

In this section the optimization model at the tactical stage is presented for the exact method. First the formulation of the model is presented followed by a thoroughly explanation of the objective function and its constraints. The section is concluded with a discussion of the model flow.

5.1.1. Mathematical formulation - fleet level

Sets

T: set of aircraft.

I_t: set of tasks for aircraft *t*.

K_t: set of time segments for aircraft *t*.

J: set of skills.

$I_{a,t}$: set of A-tasks for aircraft *t*.

$I_{o,t}$: set of remaining tasks for aircraft *t*.

$K_{c,t}$: set of time segments belonging to a C-check maintenance opportunities for aircraft *t*.

$Nd_{i,t}$: set of linked tasks for aircraft *t* having a day interval limit.

$Nm_{i,t}$: set of linked tasks for aircraft *t* having a month interval limit.

$Nh_{i,t}$: set of linked tasks for aircraft *t* having a flight-hour interval limit.

$Nc_{i,t}$: set of linked tasks for aircraft *t* having a flight-cycle interval limit.

$O(i,t)$: set of time segments possible for scheduled task *i* of aircraft *t*.

$I(i,t)$: set of time segments possible for rescheduled task *i* of aircraft *t*.

$D_{i,t}$: set of tasks *i* having a due-date before or at the last day of the last maintenance opportunity of aircraft *t*.

$P_{i,t}$: set of time segments possible for task *i* of aircraft *t*.

$I_{k,t}$: set of tasks possible, of aircraft *t*, at time segment *k*.

$Isp_{k,t}$: set of inspection tasks, of aircraft *t*, possible at time segment *k*

Parameters

$b_{i,t}$: maximum number of days between rescheduling task *i* for aircraft *t*.

$m_{i,t}$: maximum number of months between rescheduling task *i* for aircraft *t*.

$fh_{i,t}$: maximum number of flight-hours between rescheduling task *i* for aircraft *t*.

$fc_{i,t}$: maximum number of flight-cycles between rescheduling task *i* for aircraft *t*.

$c^k_{i,t}$: cost of allocating task *i* to maintenance opportunity belonging to time segment *k* for aircraft *t*.

GR^k_j: amount of available man-hours of skill type *j* at time segment *k*.

$GR^j_{i,t}$: amount of man-hours of skill type *j* required for task *i* of aircraft *t*.

$d^{k,t}$: amount of days from the start of the simulation till maintenance opportunity belonging to time segment *k* of aircraft *t*.

$m^{k,t}$: amount of months from the start of the simulation till maintenance opportunity belonging to time segment *k* of aircraft *t*.

$fh^{k,t}$: amount of flight-hours from the start of the simulation till maintenance opportunity belonging to time segment *k* of aircraft *t*.

$fc^{k,t}$: amount of flight-cycles from the start of the simulation till maintenance opportunity belonging to time segment *k* of aircraft *t*.

π_j: non-routine factor of skill type *j*.

Decision variables

$x^k_{i,t}$: 1 if task *i* is assigned to maintenance opportunity belonging to time segment *k* for aircraft *t*, and 0 otherwise.

$$min \sum_{t \in T} \sum_{k \in K_t} \sum_{i \in I_{a,t}} c^k_{i,t} \times x^k_{i,t} + \sum_{t \in T} \sum_{k \in K_{c,t}} \sum_{i \in I_{o,t}} c^k_{i,t} \times x^k_{i,t} \tag{5.1}$$

$$\sum_{k \in P_{i,t}} x^k_{i,t} = 1 \qquad \forall i \in D_{i,t} \qquad \forall t \in T \tag{5.2}$$

$$\sum_{t \in T} \sum_{i \in I_{k,t}} GR^j_{i,t} \times x^k_{i,t} + \sum_{t \in T} \sum_{i \in Isp_{k,t}} GR^j_{i,t} \times \pi^j \times x^k_{i,t} \leq Gr^k_j \qquad \forall k \in K_t \qquad \forall j \in J \tag{5.3}$$

$$1 \leq \sum_{k \in O(i,t)} d^{k,t} \times x^k_{i,t} - \sum_{k \in I(i,t)} d^{k,t} \times x^k_{i,t} \leq b_{i,t} \qquad \forall i \in Nd_{i,t} \qquad \forall t \in T \tag{5.4}$$

$$1 \leq \sum_{k \in O(i,t)} m^{k,t} \times x^k_{i,t} - \sum_{k \in I(i,t)} m^{k,t} \times x^k_{i,t} \leq m_{i,t} \qquad \forall i \in Nm_{i,t} \qquad \forall t \in T \tag{5.5}$$

$$1 \leq \sum_{k \in O(i,t)} fh^{k,t} \times x^k_{i,t} - \sum_{k \in I(i,t)} fh^{k,t} \times x^k_{i,t} \leq fh_{i,t} \qquad \forall i \in Nh_{i,t} \qquad \forall t \in T \tag{5.6}$$

$$1 \leq \sum_{k \in O(i,t)} fc^{k,t} \times x^k_{i,t} - \sum_{k \in I(i,t)} fc^{k,t} \times x^k_{i,t} \leq fc_{i,t} \qquad \forall i \in Nc_{i,t} \qquad \forall t \in T \tag{5.7}$$

$$x^k_{i,t} \in \{0,1\} \tag{5.8}$$

5.1.2. Objective function

The objective function of the task allocation model concerns the minimization of the cost associated with allocating all tasks to their optimal maintenance opportunity (equation 5.1). The calculation of the tasks costs is based on the difference between Remaining Useful Life (RUL) of the respective task and the day of the specific maintenance opportunity, amount of man-hours required to perform the task, their respective skill type and the corresponding labour rates (equation 5.9). The formulation of the cost is similar to the formulation used in the paper of Hölzel *et al.* [12].

$$c_i^k = \frac{lifetime_i - M0date_k}{lifetime_i} \times man\ hours \times labour\ rate \tag{5.9}$$

Ideally additional tasks costs, such as material cost and cost for using special tools or equipment should be added to the cost calculation. Given that this information is unavailable it is decided to neglect these additional costs. Knowing these additional costs would produce a better solution than the current approach since these missing cost could have an important contribution to the overall cost of the tasks. Nevertheless, this will not impact the potential of the proposed model since these additional costs can easily be added to the cost formulation.

5.1.3. Set of constraints

The first constraint type (equation 5.10) is referred to as the assignment constraint. Each task having a calculated due-date before or at the last day of the last given maintenance opportunity must be assigned to at least one of the possible maintenance opportunities. This constraint guarantees that each task is allocated exactly once.

$$\sum_{k \in P_{i,t}} x_{i,t}^k = 1 \qquad \forall i \in D_{i,t} \qquad \forall t \in T \tag{5.10}$$

The maintenance opportunities schedule is generated in such a way that concurrent scheduling of C-checks are limited to three and A-checks to two. The available man-hours on the overlapping days will need to be shared among the different aircraft. The second constraint type (equation 5.11) makes sure that the available man-hours per skill type is not exceeded on each of the time segments. The definition of time segments will be explained in subsection 5.1.5.

$$\sum_{t \in T} \sum_{i \in I_{k,t}} GR_{i,t}^j \times x_{i,t}^k + \sum_{t \in T} \sum_{i \in Isp_{k,t}} GR_{i,t}^j \times \pi^j \times x_{i,t}^k \leq Gr_j^k \qquad \forall k \in K_t \qquad \forall j \in J \tag{5.11}$$

The third constraint type (equations 5.12 - 5.15) ensures that the preceding relation between scheduling and rescheduling the same task is not exceeded. This means that the interval between the scheduled and rescheduled task may not be larger than its interval limits. Depending on different interval limits, of the considered task, there can be up to four constraints of the third type per task. Constraint 5.12 will be applied to linked tasks having an interval limit in days, constraint 5.13 to linked tasks having an interval limit in months, constraint 5.14 to linked tasks having an interval limit in flight-hours and constraint 5.15 to linked tasks having an interval limit in flight cycles. These third type constraints are essential in order to keep the aircraft airworthy because each task needs to be executed within the interval given [3].

$$1 \leq \sum_{k \in O(i,t)} d^{k,t} \times x_{i,t}^k - \sum_{k \in I(i,t)} d^{k,t} \times x_{i,t}^k \leq b_{i,t} \qquad \forall i \in Nd_{i,t} \qquad \forall t \in T \tag{5.12}$$

$$1 \leq \sum_{k \in O(i,t)} m^{k,t} \times x_{i,t}^k - \sum_{k \in I(i,t)} m^{k,t} \times x_{i,t}^k \leq m_{i,t} \qquad \forall i \in Nm_{i,t} \qquad \forall t \in T \tag{5.13}$$

$$1 \leq \sum_{k \in O(i,t)} fh^{k,t} \times x_{i,t}^k - \sum_{k \in I(i,t)} fh^{k,t} \times x_{i,t}^k \leq fh_{i,t} \qquad \forall i \in Nh_{i,t} \qquad \forall t \in T \tag{5.14}$$

$$1 \leq \sum_{k \in O(i,t)} fc^{k,t} \times x_{i,t}^k - \sum_{k \in I(i,t)} fc^{k,t} \times x_{i,t}^k \leq fc_{i,t} \qquad \forall i \in Nc_{i,t} \qquad \forall t \in T \tag{5.15}$$

The last constraint (equation 5.16) makes sure that the decision variables will all be either 0 or 1, thereby completing the mathematical formulation of the task allocation problem for the entire fleet.

$$x_{i,t}^{k} \in \{0,1\} \tag{5.16}$$

5.1.4. Processing data

An important aspect to speed-up the computational time is to appropriately preprocess the available data. Appropriately preprocessing means analyze and restructure the given data such that later certain queries to the data are answered quickly [91]. Fortunately the high-level programming language Python has multiple free modules which are very well capable of processing large amount of data in a very efficient way (e.g. pandas, NumPy). The proposed model makes use of three main sets of information named as follows: horizon set, task set, and due-date set. The horizon set contains information about the simulated flight-hours, flight-cycles, days, and the available man-hours for each day during the time horizon as can be seen in figure 5.1. This set of information will be accessed for the calculation of the due-date of each task and for the man-hours constraints.

Date	27/02/2019		28/02/2019		01/03/2019		02/03/2019		03/03/2019		04/03/2019	
FH simulation	8824.4		8834.6		8845.4		8856.2		8867		8877.8	
FC simulation	3474		3478.2		3482.7		3487.2		3491.7		3496.2	
Month simulation	25		25		25		25		25		25	
Day Simulation	788		789		790		791		792		793	
Weekday	Wednesday		Thursday		Friday		Saterday		Sunday		Monday	
Available man-hours	LM	HM	LM	HM	LM	HM	LM	HM	LM	HM	LM	HM
GR1	288.6	604.5	289.9	598.9	279.8	572.6	0	0	0	0	285.9	608.8
GR2	154.2	347	153.9	344.9	147.4	326	0	0	0	0	153.8	349.3
GR4	147.5	249.4	148.6	245.9	140.2	230.5	0	0	0	0	145.8	249.3
ESHS	0	553.3	0	543.5	0	518.4	0	0	0	0	0	551.8
ICH	124.7	486	126	487.2	116.1	461.2	0	0	0	0	124.3	485
PINT	0	170	0	169.8	0	158	0	0	0	0	0	170
MAP	99.5	257	99	254.7	96.8	241	0	0	0	0	96.6	257
NDT	56	56	56	56	56	56	0	0	0	0	56	56

Figure 5.1: Snapshot of horizon set.

The task set contains essential information about each considered task such as: man-hours required, skill type required, interval limits and a reference task number as can be seen in figure 5.2. It is worth mentioning that whenever a task is classified as A-check task it can be allocated to both A and C-check maintenance opportunities. While on the other hand C-check tasks can only be allocated to C-check maintenance opportunities. The task set will be accessed during calculation of the expected due-date, the third constraint type, and for the assignment constraints.

The last main information set is the due-date set and contains information about the possible maintenance opportunities for each of the considered tasks. In figure 5.3 the possible maintenance opportunities are displayed with their corresponding task number. In this figure the ones indicate that the task can be executed on the respective maintenance opportunity, while the zeros mean that this is not possible. When looking at the figure it becomes visible that C-tasks will not be possible on A-check maintenance opportunities. This set is directly linked to the decision variables, and will be accessed for each of the constraints.

5.1.5. Model flow

The optimization model is an integer linear model in which the decision variables are binary. Based on the decision variables it is possible to compute the optimal maintenance opportunity date for the

Task number	1	2	3	4	5	6	7
Limit FH	24000	N/A	750	5500	1500	10000	750
Limit FC	N/A	N/A	750	N/A	1500	N/A	N/A
Limit CAL	N/A	144 M	4	15	8	24	N/A
Man-hours	6	34	0.2	0.5	2	4	0.2
Skill-type	GR2	ESHS	GR2	GR4	GR1	PINT	ICH
A/C -task	C	C	A	A	A	C	A

Figure 5.2: Snapshot of task set.

Maintenance opportunities →

Tasks ↓

	C5.1	A4.14	A1.15	A2.15	A3.15	C6.1	A4.16	A1.16
1	1	0	0	0	0	1	0	0
2	1	0	0	0	0	1	0	0
3	1	0	0	0	0	0	0	0
4	1	1	1	1	0	0	0	0
5	1	1	0	0	0	0	0	0
6	1	0	0	0	0	1	0	0
7	1	0	0	0	0	0	0	0

Figure 5.3: Snapshot of due-date set.

allocation of all tasks. The flow diagram of the tactical stage model is presented in figure 5.5. For practical reasons the task allocation problem, for the entire fleet, should be solvable within one night (i.e. approximately twelve hours). Given this urgency it is important to save valuable computational time. The model flow consists of two main loops: the scheduling loop and solve loop. The scheduling loop will update the task and due-date sets with information about the scheduling and possible rescheduling of each of the tasks. At each iteration, of the scheduling loop, the rescheduling problem (i.e. the possible rescheduling of a task) is solved for the optimal case[1] by calculating a new due-date for each scheduled task. The new due-date is calculated from the last possible maintenance opportunity. Whenever this due-date does not exceed the last maintenance opportunity date it is added to the task set. This approach is visualized in table 5.1 where a single task is rescheduled in the scheduling loop. The circled ones in bold represent the last possible maintenance opportunity, and from this day a new due-date is calculated. For the visualized example five iterations are required to make sure that the due-date of the task exceeds the last maintenance opportunity. Hence, the first task only needs to be scheduled five times in total.

	A3.13	C5.1	A1.14	A2.14	A3.14	A4.14	A1.15	A2.15
$Task1_0$	1	**1**	0	0	0	0	0	0
$Task1_1$	0	1	1	**1**	0	0	0	0
$Task1_2$	0	0	1	1	**1**	0	0	0
$Task1_3$	0	0	0	1	1	1	**1**	0
$Task1_4$	0	0	0	0	1	1	1	**1**

Table 5.1: Illustration rescheduling an A-check task in the optimal loop.

The second loop is referred to as the solve loop. Within this loop the model is created and solved by using a MILP solver. An important aspect of the model is the way overlapping maintenance opportunities are handled. Overlapping maintenance opportunities are divided in time segments, whereby each time

[1]The case where unlimited amount of man-hours are available at each maintenance date

segment is shared with one or more aircraft or only with itself (figure 5.4). Whenever a change is made in the unique aircraft sharing a maintenance opportunity date a new maintenance opportunity time segment is introduced to the model. This approach is much more efficient compared to a traditional approach whereby a decision variable is assigned to each maintenance opportunity day. Given that A-check maintenance opportunities consists of a single day they will always have one time segment. Based on the results, obtained from the MILP solver, for each decision variable it is possible to say whether new tasks need to be added or not. If new tasks need to be added the model will return to the scheduling loop, otherwise the model terminates and outputs the solution. The final result of the fleet allocation optimizer is a maintenance schedule with the lowest overall task cost.

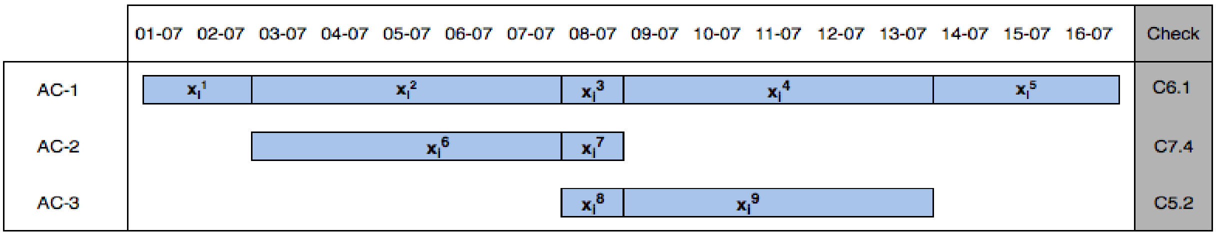

Figure 5.4: Overlapping of maintenance opportunities.

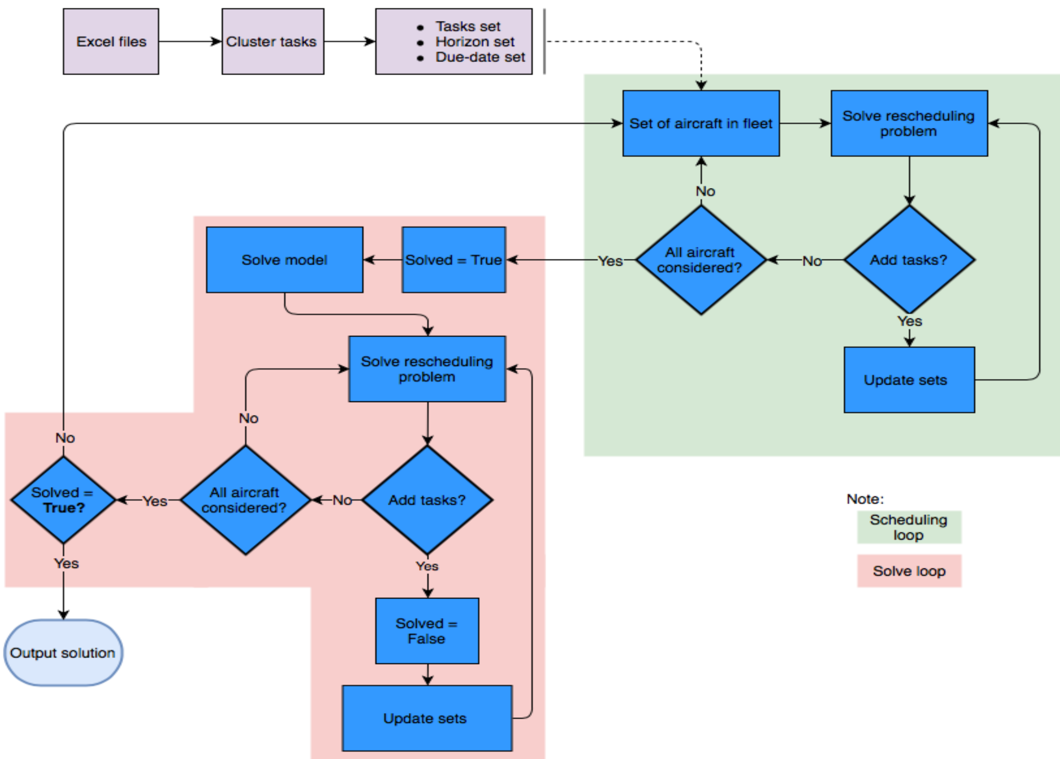

Figure 5.5: Task allocation model flow.

5.1.6. Model variant 2

In order to solve more restricted cases, at a faster rate, it was decided to improve the model by adding fictitious maintenance opportunities. This improved variant of the model has the same MILP formulation and model flow, with the only difference being the incorporation of additional fictitious maintenance opportunities. For each of the aircraft two fictitious A-check and two fictitious C-check maintenance opportunities are added after the last given maintenance opportunity. Any task allocated to these fictitious opportunities are given a cost coefficient of zero and will be removed from the final solution. This feature is introduced to better handle restricted cases by increasing the chance that the model flow terminates in a single full iteration. Table 5.2 presents an illustration of rescheduling the same task in the optimal loop of the model variant 2. In figure 5.6 it is visible that increasing the Cfactor, a factor which makes the problem more restricted, affects the solution of the proposed exact model variant 1. At a Cfactor of 2.5 and 3.0 the model requires a new full iteration and at a cfactor of 3.5 there are two additional full iterations required to solve the tactical stage model variant 1. The same situation will not require any additional iterations for the proposed exact method variant 2 as can be seen in figure 5.7. Even in the case of a Cfactor equal to 3.5 there is no new iteration required since the calculated due-date of the tasks allocated at maintenance opportunity A1.15 exceeds the last non-fictitious maintenance opportunity. These figures clearly illustrate the effect of including fictitious maintenance opportunities to the model. It is worth mentioning that the third constraint type will make sure that the gap between the scheduled and rescheduled tasks are kept within its interval limits.

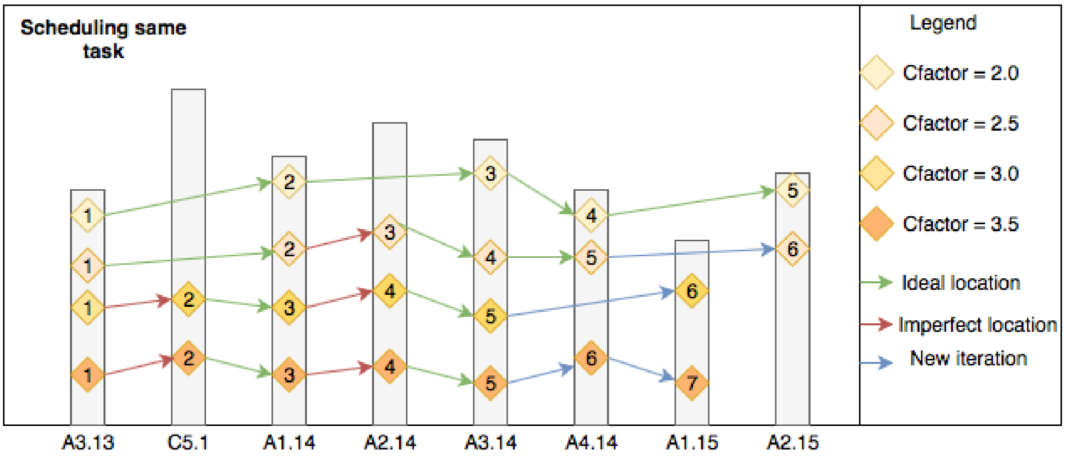

Figure 5.6: Tactical stage model variant 1: Rescheduling a task in more restricted cases.

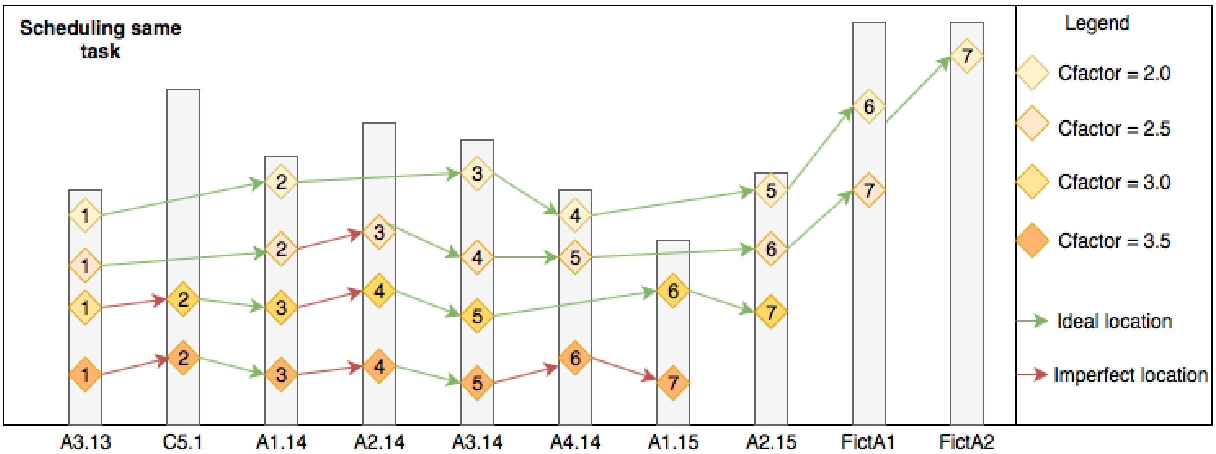

Figure 5.7: Tactical stage model variant 2: Rescheduling a task in more restricted cases.

5.2. Approximation algorithm - tactical stage

Prior to this research a literature study was conducted in order to find approaches to solve the task allocation problem. Based on the formulation of the bin packaging problem it became clear that the task allocation problem can be seen as a bin packaging problem. In the context of this research bins can be seen as maintenance opportunities which need to be filled with their respective items (i.e. preventive maintenance tasks) [12]. A common strategy in bin packaging problems is to sort the specific list of items and then use a well-known classical on-line algorithm. The proposed approximation algorithm will use a similar approach. In order to assess the quality of the approximation algorithm

	A3.13	C5.1	A1.14	A2.14	A3.14	A4.14	A1.15	A2.15	Fict A1	Fict A2
$Task1_0$	1	**1**	0	0	0	0	0	0	0	0
$Task1_1$	0	1	1	**1**	0	0	0	0	0	0
$Task1_2$	0	0	1	1	**1**	0	0	0	0	0
$Task1_3$	0	0	0	1	1	1	**1**	0	0	0
$Task1_4$	0	0	0	0	1	1	1	**1**	0	0
$Task1_5$	0	0	0	0	0	1	1	1	1	**1**

Table 5.2: Illustration rescheduling an A-check task in the optimal loop of model variant 2.

it is required to satisfy the same constraints and calculate the cost in the same way as was done in the exact method. Therefore, the mathematical formulation for the model is equivalent to the presented formulation for the exact method. In this section the optimization model at the tactical stage is presented for the approximation algorithm. First the allocation sequence is presented followed by a thoroughly explanation of how the algorithm deals with the constraints and the objective function. The section is concluded with a discussion of the model flow.

5.2.1. Notation of the allocation loop input

Sets

T: set of aircraft.

I_t: set of tasks for aircraft *t* (excluding rescheduled tasks) having a due-date before or at the last day of the last maintenance opportunity of aircraft *t*.

K_t: set of time segments for aircraft *t*.

$I_{A1,t}$: subset of A-tasks for aircraft *t* classified as class 1.

$I_{A2,t}$: subset of A-tasks for aircraft *t* classified as class 2.

$I_{A3,t}$: subset of A-tasks for aircraft *t* classified as class 3.

$I_{C1,t}$: subset of C-tasks for aircraft *t* classified as class 1.

$I_{C2,t}$: subset of C-tasks for aircraft *t* classified as class 2.

$I_{C3,t}$: subset of C-tasks for aircraft *t* classified as class 3.

$K_{c,t}$: set of time segments belonging to a C-check maintenance opportunities for aircraft *t*.

Parameters

$c_{i,t}^k$: cost of allocating task *i* to maintenance opportunity belonging to time segment *k* for aircraft *t*.

p_1: penalty for class 1 items.

p_2: penalty for class 2 items.

$$\sum_{t\in T}\sum_{k\in K_t}\sum_{i\in I_{A1,t}} c_{i,t}^k + p_1 + \sum_{t\in T}\sum_{k\in K_t}\sum_{i\in I_{A2,t}} c_{i,t}^k + p_2 + \sum_{t\in T}\sum_{k\in K_t}\sum_{i\in I_{A3,t}} c_{i,t}^k$$
$$+\sum_{t\in T}\sum_{k\in K_{c,t}}\sum_{i\in I_{C1,t}} (c_{i,t}^k + p_1) + \sum_{t\in T}\sum_{k\in K_{c,t}}\sum_{i\in I_{C2,t}} (c_{i,t}^k + p_2) + \sum_{t\in T}\sum_{k\in K_{c,t}}\sum_{i\in I_{C3,t}} c_{i,t}^k \quad (5.17)$$

5.2.2. Order of allocating tasks

In order to successfully allocate all tasks the algorithm needs to know which tasks have a higher priority and must be allocated first. Therefore, the algorithm classifies tasks based on their tasks intervals. The first class tasks consist of tasks having an interval which require them to be allocated at every maintenance opportunity. Regardless of the approach used these tasks need to be allocated to their only possible maintenance opportunity. Therefore, these tasks will receive the highest penalty forcing the algorithm to first allocate them to the maintenance opportunities. The second class tasks consist of tasks that sometimes can be scheduled on multiple maintenance opportunities, but there are also occasions where these tasks can only reach one maintenance opportunity. These tasks will receive the second highest penalty. Each of the tasks that can be allocated to two or more maintenance opportunities, at each occasion (i.e. the interval limits of the considered task are large enough to skip at least one maintenance opportunity at each location in the given time horizon), are classified as class three tasks and will receive no penalty. A visualization of the different classes for A-check tasks and C-check tasks are presented in figure 5.9 and 5.10, respectively. Besides the constant penalty the task cost consist of a variable part which is calculated in the same way as for the exact method which is presented in equation 5.9. The results of this calculation is a long list with tasks and their corresponding costs. The mathematical notation of this list is presented in subsection 5.2.1.

Given that aircraft utilization differs per aircraft it is required to calculate the classes lower and upper limits for each aircraft separately. This means that there will be specific interval limits in flight hours, flight cycles and calendar months for each of the classes per aircraft. For example when considering figure 5.8 the class one C-check tasks limit is 7,250 flight hours, 4,050 flight cycles and 35 months since this is the required limit to bridge the gap between each C-check maintenance opportunity or start/end point. The limit for the class two C-check tasks is set at the combination of the two highest consecutive maintenance opportunities gaps, in the illustrated example this is the gap between C6.1 and the end of the time horizon. This means that tasks with intervals between the upper limit of class 1 tasks and 13,850 flight hours, 7,850 flight cycles, or 65 months will be classified as class two tasks. Any tasks having intervals larger than the class two upper limits are class three tasks which will receive no cost penalty. This same procedure also applies to A-check tasks.

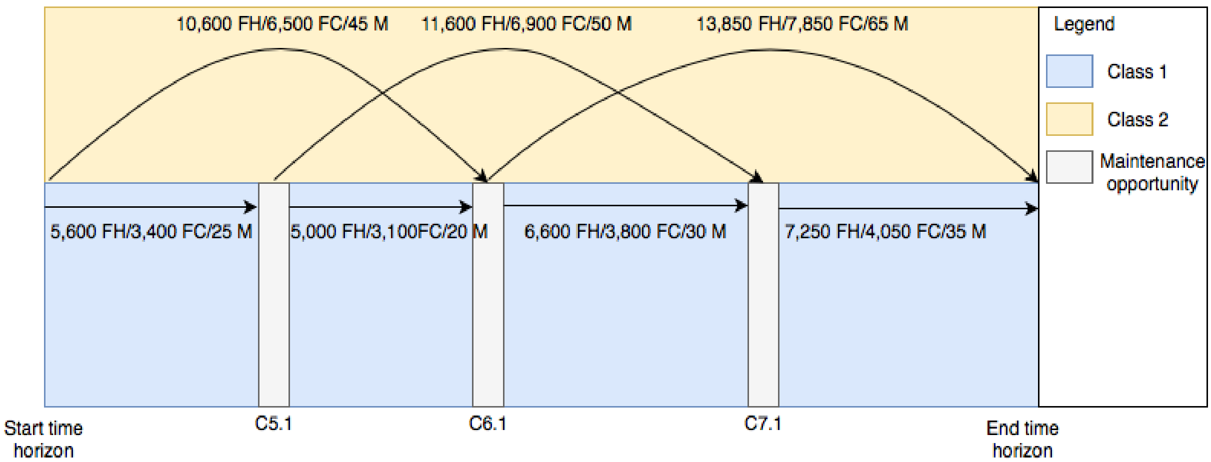

Figure 5.8: Illustration of classes limit intervals for C-check tasks.

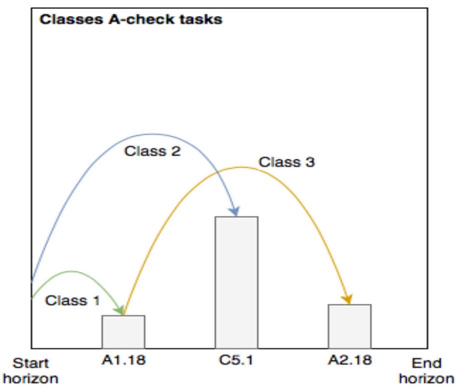

Figure 5.9: Visualization of classes for A-check tasks.

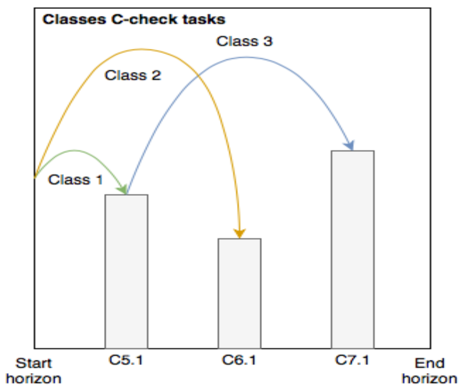

Figure 5.10: Visualization of classes for C-check tasks.

5.2.3. Set of constraints and objective function

The approximation algorithm has the same constraints as the mathematical formulation of the exact method at the tactical stage. Whenever a task is allocated the algorithm will remove the allocated task from the allocation list. This ensures that each task is allocated exactly once, which means that the first constraint type (equation 5.10) is satisfied. Each of the time segments (i.e. bins) will have a specific amount of available man-hours per skill type. Whenever the bin does not have enough man-hours to allocate the task it will be closed, for the considered task, and the next bin will be considered. This means that available man-hours on each of the time segments, for each skill type, will never be exceeded. Thus, the second constraint type (equation 5.11) is also satisfied by the approximation algorithm. Whenever a task is allocated the rescheduling problem for this same task is solved till its due-date is beyond the end of the considered time horizon. This is equivalent to the third constraint type (equations 5.12 - 5.15) of the exact method. Thus, each of the constraints considered at the exact method are also covered within the approximation algorithm. Whenever a task is allocated the cost of the task is calculated according to the cost formula (equation 5.9) and added to the objective function list. This means that the penalties used for determining the allocation order of the tasks are not included in the objective function list. This makes it possible to obtain the solution quality of the approximation algorithm since the optimal solution is obtained from the exact method.

5.2.4. Fictitious opportunities

Without the fictitious maintenance opportunity the tactical stage model would only allocate tasks to the last C-check maintenance opportunities if the calculated due-date of the task is before the end of the time horizon. However, at some occasions the last C-check maintenance opportunity, given by the maintenance schedule, is at the end of the time horizon. This means that less tasks will actually be allocated than most likely would have happened when the maintenance schedule had an extended time horizon with an additional planned C-check. To overcome this problem a fictitious maintenance opportunity for a C-check is added to the given maintenance schedule for each non-phased out aircraft. The fictitious maintenance opportunity is placed within the C-check tasks limit intervals from the last scheduled C-check maintenance opportunity of the considered aircraft. Any C-check tasks that have a calculated due-date before the fictitious C-check maintenance opportunity will need to be allocated.

5.2.5. Model flow

The flow diagram of the tactical level approximation algorithm is presented in figure 5.12. Before the allocation loop starts the tasks list, consisting of the tasks for the entire fleet and their corresponding task cost, is sorted by cost. This means that the tasks with the highest cost will be allocated first. A potential problem with using a standard bin packaging approach is that it closes bins that could potentially hold later identified items [57]. This problem becomes important when we look at the overlapping maintenance opportunities within the fleet and is referred to as the bins sequence problem. The algorithms tries to first fill the bins shared with no aircraft, since these bins does not affect any other aircraft. The bins shared with one other aircraft are filled by selecting the bin with the most available man-hours first for the considered skill type. This same procedure holds for aircraft that share the bin with two other aircraft. This way the most restricted bins (i.e. bins shared among three aircraft) are filled as last. When considering the maintenance overlaps illustrated in figure 5.11 it becomes clear how the algorithm deals with the above mentioned problem. In the illustrated example the algorithm will try, for aircraft one, to first fill bin one and bin five since these bins are not shared among other aircraft. When these bins are full bin two and bin four will be used. If there is still tasks left for aircraft one the last bin three, which is shared with two other aircraft is used. Whenever all tasks are allocated the model terminates and outputs the solution. Whenever a task can not be allocated the model will add fictitious man-hours to the best bin location and will add these man-hours to a feedback file. This way it is possible to provide feedback to the user of the model about the constricted dates. The feedback can be implemented in the maintenance schedule by extending the maintenance opportunities dates or adding additional man-hours to the restricted dates.

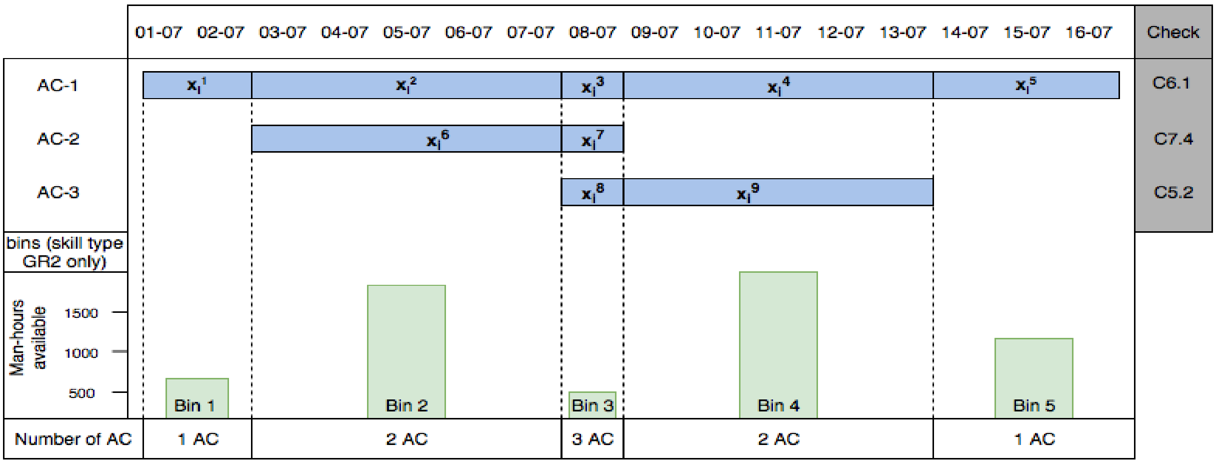

Figure 5.11: Overlapping of maintenance opportunities with bins.

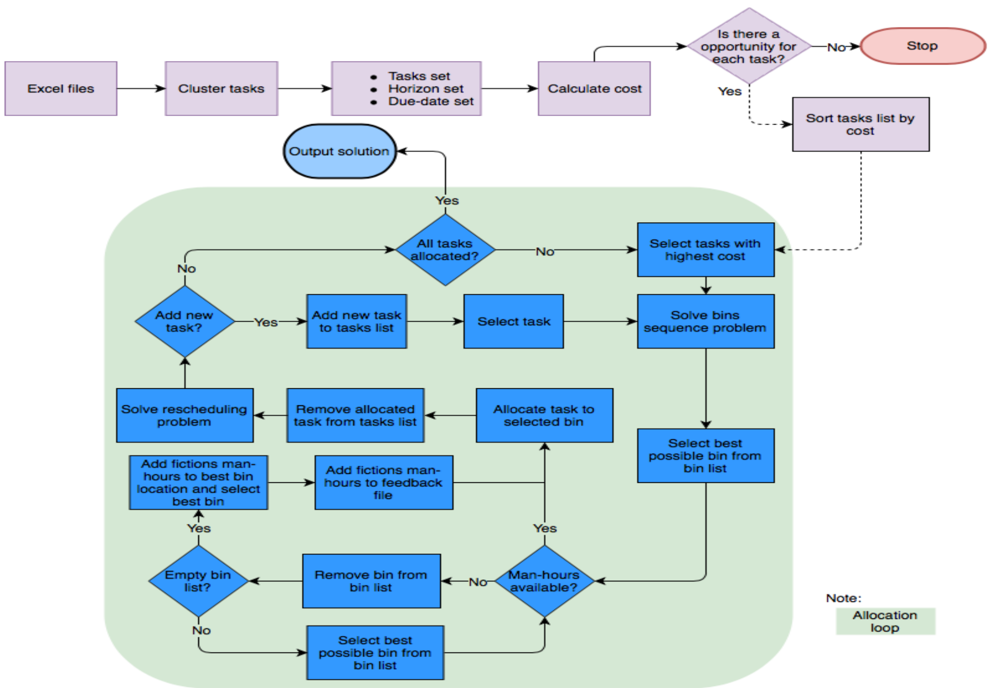

Figure 5.12: Approximation tactical stage model flow.

6

Task allocation model - operational stage

This chapter presents a two-level modelling framework for solving the task allocation problem at the operational stage. At the first operational level the task allocation problem will be solved for each aircraft seperately. At this level it is possible to add additional maintenance tasks (e.g. non-routine maintenance, modifications, deferred defects, incorporation of service bulletins (SB) or Airworthiness Directives (AD)) that were not considered at the tactical stage. Furthermore, it is possible to input the current utilization of the aircraft thereby correcting the small deviation obtained by assuming an average flight-hours and flight-cycles utilization. The second operational level will allocate the tasks down to shift level.

6.1. Exact method - operational stage level 1

In this section the optimization model at the first level of the operational stage is presented for the exact method. First the formulation of the model is presented followed by a thoroughly explanation of the objective function and its constraints. The section is concluded with a discussion of the model flow.

6.1.1. Mathematical formulation - aircraft level

Sets

I: set of tasks.

K: set of maintenance opportunities (including fictitious).

K_r: set of maintenance opportunities (excluding fictitious).

J: set of skills.

I_a: set of A-tasks.

I_o: set of remaining tasks.

K_c: set of C-check maintenance opportunities (excluding fictitious).

Nd_i: set of linked tasks having a day interval limit.

Nm_i: set of linked tasks having a month interval limit.

Nh_i: set of linked tasks having a flight-hour interval limit.

Nc_i: set of linked tasks having a flight-cycle interval limit.

O(i): set of maintenance opportunities possible for scheduled task *i*.

I(i): set of maintenance opportunities possible for rescheduled task *i*.

T_i: set of tasks *i* having a due-date before the last day of the last fictitious maintenance opportunity.

P_i: set of maintenance opportunities possible for task *i*.

I_k: set of tasks possible at maintenance opportunity *k*.

Isp_k: set of inspection tasks possible at maintenance opportunity *k*

Parameters

b_i: maximum number of days between rescheduling task *i*.

m_i: maximum number of months between rescheduling task *i*.

fh_i: maximum number of flight-hours between rescheduling task *i*.

fc_i: maximum number of flight-cycles between rescheduling task *i*.

c_i^k: cost of allocating task *i* to maintenance opportunity *k*.

GR_j^k: amount of available man-hours of skill type *j* at maintenance opportunity *k*.

GR_i^j: amount of man-hours of skill type *j* required for task *i*.

d^k: amount of days from the start of the simulation till maintenance opportunity *k*.

m^k: amount of months from the start of the simulation till maintenance opportunity *k*.

fh^k: amount of flight-hours from the start of the simulation till maintenance opportunity *k*.

fc^k: amount of flight-cycles from the start of the simulation till maintenance opportunity *k*.

π_j: non-routine factor of skill type *j*.

Decision variables

x_i^k: 1 if task *i* is assigned to maintenance opportunity *k*, and 0 otherwise.

$$min \sum_{k\in K_r} \sum_{i\in I_a} c_i^k \times x_i^k + \sum_{k\in K_c} \sum_{i\in I_o} c_i^k \times x_i^k \tag{6.1}$$

$$\sum_{k\in P_i} x_i^k = 1 \qquad \forall i \in T_i \tag{6.2}$$

$$\sum_{i\in I_k} GR_i^j \times x_i^k + \sum_{i\in Isp_{k,t}} GR_{i,t}^j \times \pi^j \times x_{i,t}^k \leq Gr_j^k \qquad \forall j \in J,\ \forall k \in K_r \tag{6.3}$$

$$1 \leq \sum_{k\in O(i)} d^k \times x_i^k - \sum_{k\in I(i)} d^k \times x_i^k \leq b_i \qquad \forall i \in Nd_i \tag{6.4}$$

$$1 \leq \sum_{k\in O(i)} m^k \times x_i^k - \sum_{k\in I(i)} m^k \times x_i^k \leq m_i \qquad \forall i \in Nm_i \tag{6.5}$$

$$1 \leq \sum_{k\in O(i)} fh^k \times x_i^k - \sum_{k\in I(i)} fh^k \times x_i^k \leq fh_i \qquad \forall i \in Nh_i \tag{6.6}$$

$$1 \leq \sum_{k\in O(i)} fc^k \times x_i^k - \sum_{k\in I(i)} fc^k \times x_i^k \leq fc_i \qquad \forall i \in Nc_i \tag{6.7}$$

$$x_i^k \in \{0,1\} \tag{6.8}$$

6.1.2. Objective function and set of constraints

Both the objective function (equation 6.1) and the set of constraints (equations 6.2 - 6.8) of the operational stage level 1 model are almost identical to the model formulation at the tactical stage, with the only difference being that there is now a single aircraft considered. Therefore, information regarding the objective function and the used constraints can be found in sections 5.1.2 and 5.1.3, respectively.

6.1.3. Model flow

The flow diagram of the operational stage model level 1 is presented in figure 6.1. The optimization model is an integer linear model in which the decision variables are binary. Based on the decision variables it is possible to compute the optimal maintenance opportunity date for the allocation of all tasks. The task allocation problem, for one single aircraft, should be solvable within ten minutes. Given this urgency it is important to save valuable computational time; therefore, the model consists of two main loops: the scheduling loop and solve loop. These loops were already explained in section 5.1.5. The model also incorporates two fictitious A-check and two fictitious C-check maintenance opportunities in the same way as was done in section 5.1.6. It is worth mentioning that these fictitious maintenance opportunities have an infinite man-hours capacity.

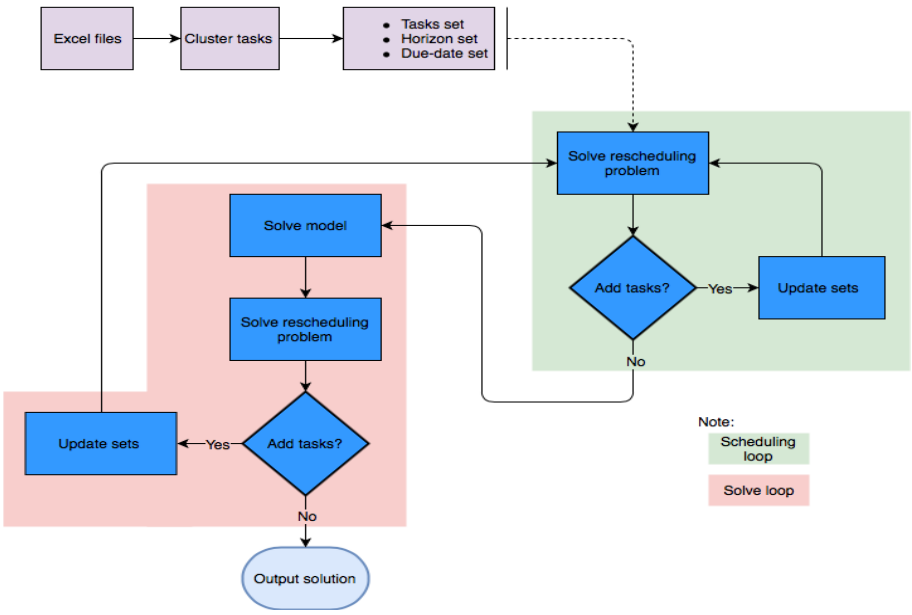

Figure 6.1: Exact method operational stage level 1 model flow.

6.2. Exact method - operational stage level 2

In this section the optimization model at the second level of the operational stage is presented for the exact method. In the second level tasks will be allocated down to shift level. First the formulation of the model is presented followed by a thoroughly explanation of the objective function and its constraints. The section is concluded with a discussion of the model flow.

6.2.1. Mathematical formulation - shift level

Sets

I: set of tasks.

S: set of shifts.

J: set of skills.

I_o: set of tasks that open access panels.

I_c: set of tasks that close the access panels.

I_{insp}: set of tasks labelled as inspection tasks.

I_r: set of remaining tasks.

N_i: set of related tasks.

O_i: set of shifts possible for scheduled task i.

R_i: set of shifts possible for linked task with scheduled task i.

Parameters

t_s: shift number.

c_i^s: cost of allocating task *i* to shift *s*.

GR_j^s: amount of available man-hours of skill type *j* at shift *s*.

GR_i: amount of man-hours required for task *i*.

GR_i^j: amount of man-hours of skill type *j* required for task *i*.

π_j: non-routine factor of skill type *j*.

Decision variables

x_i^s: 1 if task *i* is assigned to shift *s*, and 0 otherwise.

$$min \sum_{s\in S}\sum_{i\in I_r} GR_i \times t_s \times x_i^s + \sum_{s\in S}\sum_{i\in I_o} GR_i \times 0.01 t_s \times x_i^s + \sum_{s\in S}\sum_{i\in I_c} GR_i \times 0.01 t_s \times x_i^s + \sum_{s\in S}\sum_{i\in I_{insp}} GR_i \times 10 t_s \times x_i^s \quad (6.9)$$

$$\sum_{s\in S} x_i^s = 1 \qquad \forall i \in I \quad (6.10)$$

$$\sum_{i\in I} GR_i^j \times x_i^s + \sum_{i\in I_{insp}} GR_i^j \times \pi^j \times x_i^s \leq Gr_j^s \qquad \forall j \in J \qquad \forall s \in S \quad (6.11)$$

$$\sum_{s\in R_i} t^s \times x_i^s - \sum_{s\in O_i} t^s \times x_i^s \geq 0 \qquad \forall i \in N_i \quad (6.12)$$

$$x_i^s \in \{0, 1\} \quad (6.13)$$

6.2.2. Objective function

The objective function of the second stage of the operational level consists of the minimization of cost. In order to meet the scheduled aircraft maintenance downtime it is important to identify non-routine maintenance as early as possible. According to the research conducted by Michael Reopel [92] ideally 90 per cent of the non-routine findings should be found within the first 10 per cent time of the maintenance check. Most of the non-routine maintenance comes from findings during inspection tasks; thus, these tasks will be given the highest variable part in the objective function (i.e. the assigned cost to the inspections tasks will increase significantly by increased shift number). Each of the remaining tasks will get a variable part which consists of the required man-hours and the shift number. This variable part forces each of the tasks to be executed as soon as possible thereby minimizing the maintenance downtime of the aircraft.

6.2.3. Set of constraints

The first constraint type (equation 6.14) is referred to as the assignment constraint. Each of the tasks must be assigned to at least one of the available shifts during the considered maintenance opportunity. This constraint guarantees that each task is allocated exactly once during the maintenance opportunity.

$$\sum_{s\in S} x_i^s = 1 \qquad \forall i \in I \quad (6.14)$$

There are three different shifts during a single working day, morning shift, afternoon shift and night shift. In accordance with the approach of the airline the available man-hours are split over these shifts. Each morning and afternoon shift receive 40 per cent of the available man-hours, respectively. The night shift will get 20 per cent of the available man-hours on the respective day. The second constraint type (equation 6.15) makes sure that none of the shifts exceed the available man-hours per skill type.

$$\sum_{i \in I} GR_i^j \times x_i^s + \sum_{i \in I_{insp}} GR_i^j \times \pi^j \times x_i^s \leq Gr_j^s \quad \forall j \in J \quad \forall s \in S \tag{6.15}$$

The third constraint type (equation 6.16) ensures that the preceding relation between tasks is not violated. This includes tasks requiring the opening of specific access panels, these tasks can only be executed after the specific panels have been opened. Furthermore, access panels will only be closed after each of the tasks requiring these specific access panel is executed. Also tasks with the same item numbers and large tasks that were broken down, as explained in the next subsection, that need to be executed in series will be covered by this third constraint.

$$\sum_{s \in S} t^s \times x_i^s - \sum_{s \in S} t^s \times x_i^s \geq 0 \quad \forall i \in N_i \tag{6.16}$$

The last constraint makes sure that the decision variables will all be either 0 or 1 (equation 6.17) thereby completing the mathematical formulation of the task allocation problem for the second level of the operational stage.

$$x_i^k \in \{0, 1\} \tag{6.17}$$

6.2.4. Model flow

The flow diagram of the exact method at the operational stage level 2 is presented in figure 6.2. Often it is possible to maintain more than one system or component by opening the same access panel [4]. In order to prevent the repetitive opening and closing of access panels an access panel loop is introduced in the model. Within this loop access panels are grouped together in such a way that they will only be opened and closed once during a maintenance opportunity. The third constraint (equation 6.16) will make sure that the required access panels will be kept open until all tasks requiring those access panels are executed. In order to allocate the tasks down to shift level it is required to break down the large tasks into several elemental tasks. The model consists of a feature that breaks down large tasks into smaller tasks requiring a maximum of four man-hours. According to the paper of Chen *et al.* [69] breaking down large tasks is a common practise executed by aircraft maintenance managers. In this way it is possible to complete each tasks by one technician within one shift. However, it must be noted that some tasks can not be broken down to the elemental level, these tasks should be carefully planned by the respective maintenance manager. Since such data is not available it is assumed that all large tasks can be broken down to the elemental level.

6.3. Approximation algorithm - operational stage level 1

In this section the optimization model at the first level of the operational stage is presented for the approximation algorithm. The mathematical formulation used for the exact method at the operational level 1 will be equivalent to the one used for the approximation algorithm. First the allocation sequence is presented followed by a thoroughly explanation of how the algorithm deals with the constraints and the objective function. The section is concluded with a discussion of the model flow

6.3.1. Notation of the allocation loop input

Sets

I: set of tasks (excluding rescheduled tasks) having a due-date before or at the last day of the last maintenance opportunity.

K: set of maintenance opportunities.

I_{A1}: set of A-tasks classified as class 1.

I_{A2}: set of A-tasks classified as class 2.

I_{A3}: set of A-tasks classified as class 3.

I_{C1}: set of C-tasks classified as class 1.

I_{C2}: set of C-tasks classified as class 2.

I_{C3}: set of C-tasks classified as class 3.

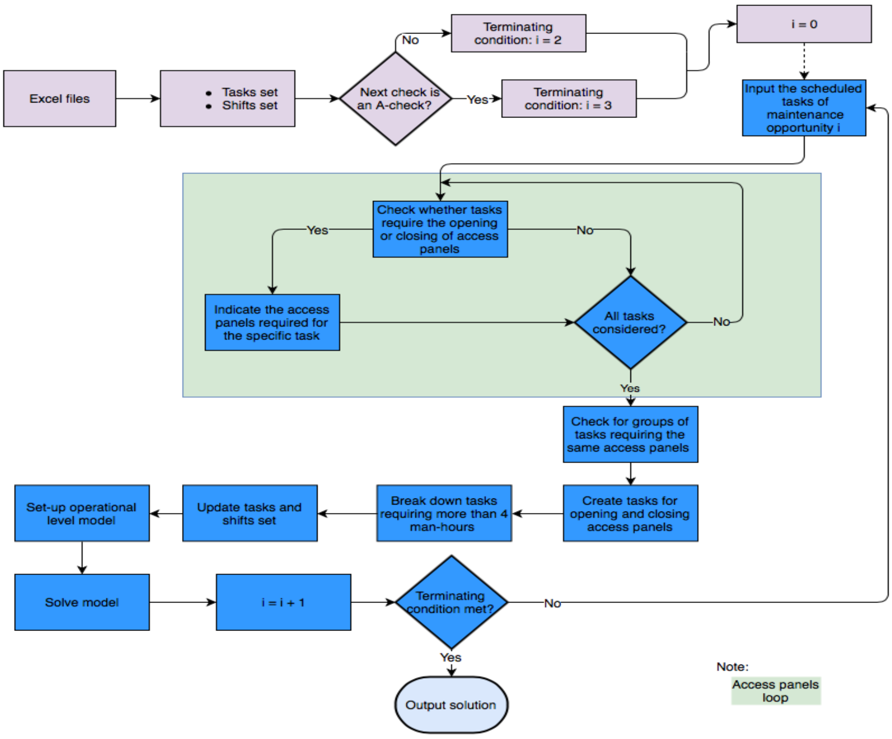

Figure 6.2: Exact method operational stage level 2 model flow.

K_c: set of C-check maintenance opportunities.

Parameters

c_i^k: cost of allocating task *i* to maintenance opportunity *k*.

p_1: penalty for class 1 items.

p_2: penalty for class 2 items.

$$\sum_{k\in K}\sum_{i\in I_{A1}} c_i^k + p_1 + \sum_{k\in K}\sum_{i\in I_{A2}} c_i^k + p_2 + \sum_{k\in K}\sum_{i\in I_{A3}} c_i^k$$
$$+ \sum_{k\in K_c}\sum_{i\in I_{C1}} c_i^k + p_1 + \sum_{k\in K_c}\sum_{i\in I_{C2}} c_i^k + p_2 + \sum_{k\in K_c}\sum_{i\in I_{C3}} c_i^k \qquad (6.18)$$

6.3.2. Order of allocating tasks

The approximation algorithm, at the operational level 1, determines tasks priorities in the same way as was explained in subsection 5.2.2. The mathematical notation for the tasks cost list, required during the allocation loop, is presented in subsection 6.3.1.

6.3.3. Set of constraints and objective function

The approximation algorithm at stage 1 of the operational level is almost identical as the one presented at the tactical level. With the exception that there is only one aircraft considered. Therefore, information about how the algorithm satisfies all the constraints and the objective function can be found in subsection 5.2.3.

6.3.4. Model flow

The model flow is exactly the same as the one presented at the tactical stage (figure 5.12). The calculation of the different class types is also in accordance with the approach explained at the approximation algorithm tactical level section.

6.4. Approximation algorithm - operational stage level 2

In this section the optimization model at the second level of the operational stage is presented for the approximation algorithm. At this operational level tasks will be allocated down to shift level. This section is concluded with a discussion of the model flow.

6.4.1. Notation of the allocation loop input

Sets

I: set of tasks.

S: set of shifts.

J: set of skills.

I_o: set of tasks that open the access panels.

I_c: set of tasks that close the access panels.

I_{insp}: set of tasks labelled as inspection tasks.

I_r: set of remaining tasks.

N_i: set of related tasks.

Parameters

GR_i: amount of man-hours required for task *i*.

c_i^s: cost of allocating task *i* to shift *s*.

p_1: penalty for access panel tasks.

p_2: penalty for inspection tasks.

$$\sum_{i \in I_o} GR_i + p_1 + \sum_{i \in I_{insp}} GR_i + p_2 \sum_{i \in I_r} GR_i + \sum_{i \in I_c} GR_i \times 0.001 \tag{6.19}$$

6.4.2. Order of allocating tasks

In order to successfully allocate all tasks the algorithm needs to know which tasks have a higher priority and must be allocated first. Given that most tasks require access panels to be opened, the tasks for opening access panels were given the highest penalty. This corresponds to reality where the first shift, during a C-check, is mostly used to open the required access panels for the C-check maintenance opportunity. Another important factor is to identify non-routine maintenance as early as possible. Therefore, the inspections tasks will be given the second highest penalty. For practical reasons the closing of access panels will be given a cost of 0.001, this forces the algorithm to allocate them as last. The remaining tasks will receive no penalties. The mathematical formulation that creates the tasks orientation list is given in subsection 6.4.1. The tasks orientation list is sorted by cost, and the task with the highest cost will be allocated first.

6.4.3. Sets of constraints

The approximation algorithm at the operational level 2 has the same constraints as the mathematical formulation of the exact method at this same level. Whenever a task is allocated the algorithm will remove the allocated task from the tasks orientation list. This way each task is allocated exactly once, which means that the first constraint type (equation 6.10) is satisfied by the algorithm. Each of the shifts will have a maximum amount of man-hours, per skill type, that can be used. Whenever the shift does not have enough man-hours to allocate the task it will be closed, for the considered task, and the next shift will be considered. Thus, the second constraint type (equation 6.11) is also satisfied by the approximation algorithm. Whenever a task is selected from the tasks orientation list it is checked whether the considered task has any linked tasks. The algorithm will make sure that the linked tasks that need to be executed prior to the considered task are first allocated, followed by the considered task, and then the linked tasks that have to be executed after the considered task will be allocated. These linked tasks consists of tasks with the same item number and large tasks that were broken down.

The algorithm will make sure that the tasks will be allocated in the same or a later shift, thereby keeping the allocation sequence of the tasks in the right order. For practical reasons it was decided that all tasks that open the access panels where allocated first and the algorithm will make sure that the closing of access panels tasks are done after or on the shift where the last task is allocated. Thereby ensuring that the algorithm satisfies the third constraint type (equation 6.12). Thus, each of the constraints considered at the exact method are covered within the approximation algorithm.

6.4.4. Model flow

The flow diagram of the operational stage level 2 approximation algorithm is presented in figure 6.3. The model flow consists of two main loops: access panels loop and allocation loop. The access panels loop is used to prevent the repetitive opening and closing of access panels by connecting tasks that require the opening of the same access panels. In the allocation loop tasks are allocated in order of cost. Whenever a task is selected, in the allocation loop, the algorithm will make sure that the linked tasks (i.e. tasks that need to be executed in series or tasks that have been broken down in parts) that need to be executed prior to the tasks are first allocated. Followed by the considered tasks, and then the linked tasks that need to be executed after the considered tasks will be allocated. The algorithm will make sure that the linked tasks will be allocated in the same or a later shift, thereby keeping the allocation sequence of the tasks in the right order. After which the new highest task from the cost list is selected and allocated in the same way. Whenever a task is allocated it is removed from the task list to ensure that each task is allocated exactly once. As with the exact method, for the same operational level, large tasks are broken down to elemental level to ensure that each of the tasks can be allocated to the available shifts. Whenever a task can not be allocated the algorithm will add fictitious man-hours to the last shift and will add these man-hours to a feedback file. This way it is possible to provide feedback to the user of the algorithm about the additional required man-hours for the considered maintenance opportunity of the considered aircraft. Since access panels tasks have the highest constant penalty the model will first allocate each of those tasks. It is worth mentioning that the algorithm will try to reduce the aircraft down time by trying to allocate the tasks to shifts as early as possible just as the objective function of the exact method aims to do.

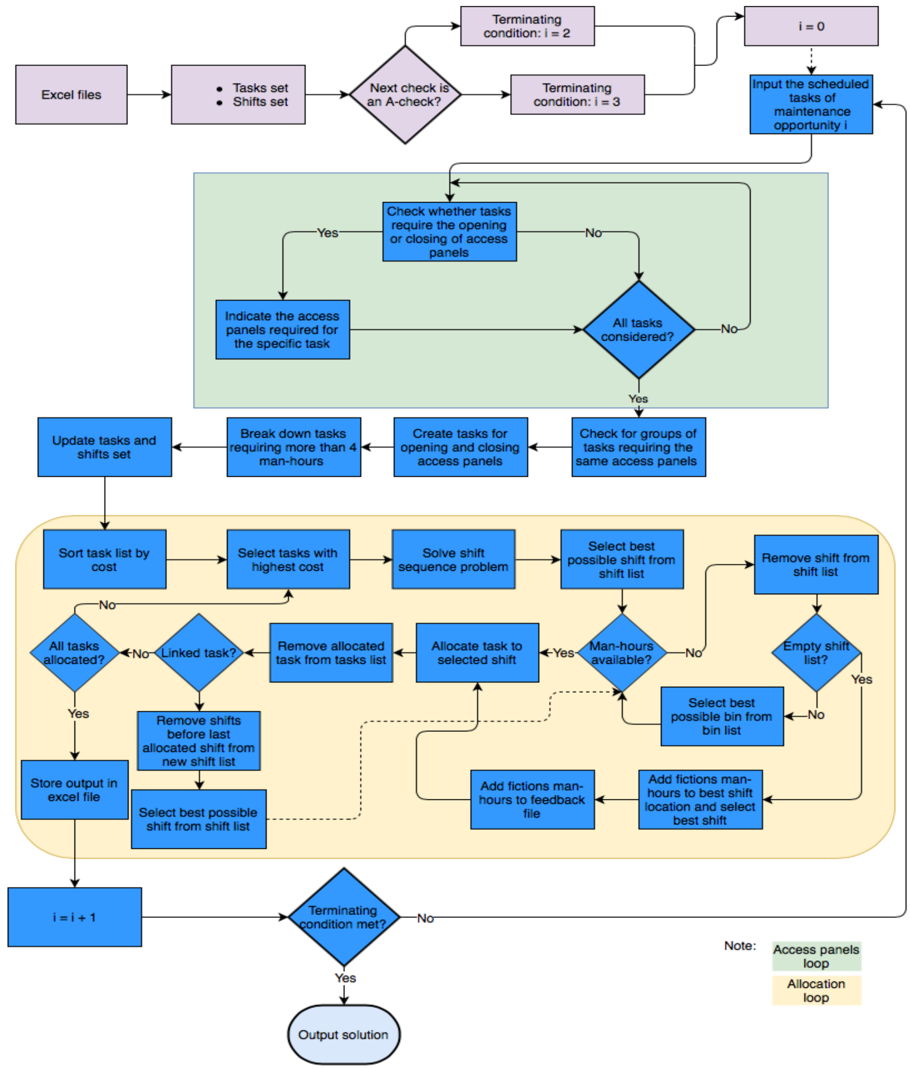

Figure 6.3: Approximation operational stage level 2 model flow.

II

Case study: European airline

7
Results

In the first part of this research three models were developed: one model at the tactical stage and two operational level models. These models will be tested with data from a European airline. The data comprises of tasks information of 45 aircraft, maintenance opportunity schedule for the entire fleet, historical data on available man-hours, non-routine rates and daily utilization for each of the considered aircraft. The considered time horizon is around four years starting at September 2017 and ending at December 2021. During this time horizon there are three aircraft, AC-24, AC-28, and AC-41, which will be phased-out. This chapter presents results from executing the models with the airlines data. Each of the exact models were coded in Python 3.7 and solved using the free academic license of Gurobi. The models were solved on an Intel Core i5 1.6 GHz laptop with 4GB ram. The following sections will provide insights in the problem size, computational time and solution quality, main solution parameters and a discussion about the obtained results for each of the considered exact models.

7.1. Exact method - results tactical stage

This section presents the results of applying the airlines data on the proposed tactical stage exact method variant 2. The considered fleet consists of 45 aircraft with three different aircraft types.

7.1.1. Problem size

The problem size, of the tactical stage model, is obtained from Gurobi and indicates that the model has 1.15 million decision variables and 327,901 constraints. From which 103,779 are type 1 constraints (equation 5.2), 6,912 are type 2 constraints (equation 5.3) and 217,210 are type 3 constraints (equations 5.4 - 5.7).

7.1.2. Computational time and solution

To solve the considered time horizon, for the entire fleet, the tactical stage model takes 1,300 seconds on the mentioned laptop. The final objective function value is 599,026. More importantly, the wasted RUL is equal to 31,016 years. The model indicates that 65,531 A-check tasks and 20,758 C-check tasks have been allocated, this excludes the tasks that were allocated to fictitious maintenance opportunities. It is worth mentioning that these numbers represent the clustered tasks, the declustering of the clustered tasks will be done at the operational stage level 2 model.

7.1.3. Visualization solution parameters

The main parameters visualized in this subsection are the wasted RUL in years, number of tasks allocated and man-hours per skill type used during the considered time horizon for each considered aircraft. Figure 7.1 presents the wasted RUL in years for each of the considered aircraft. From the analysis of the figure it becomes clear that on average, per aircraft, 689 years of RUL is wasted. When analyzing the data of the average RUL values for A-check and C-check tasks, the C-check tasks waste 205 days RUL and the A-check tasks waste 19.3 days RUL on average. The total amount of tasks allocated, per aircraft, are presented in figure 7.2. This data indicates that 24 per cent of the allocated tasks are C-check tasks and 76 per cent of the tasks are A-check tasks. The figures also clearly shows

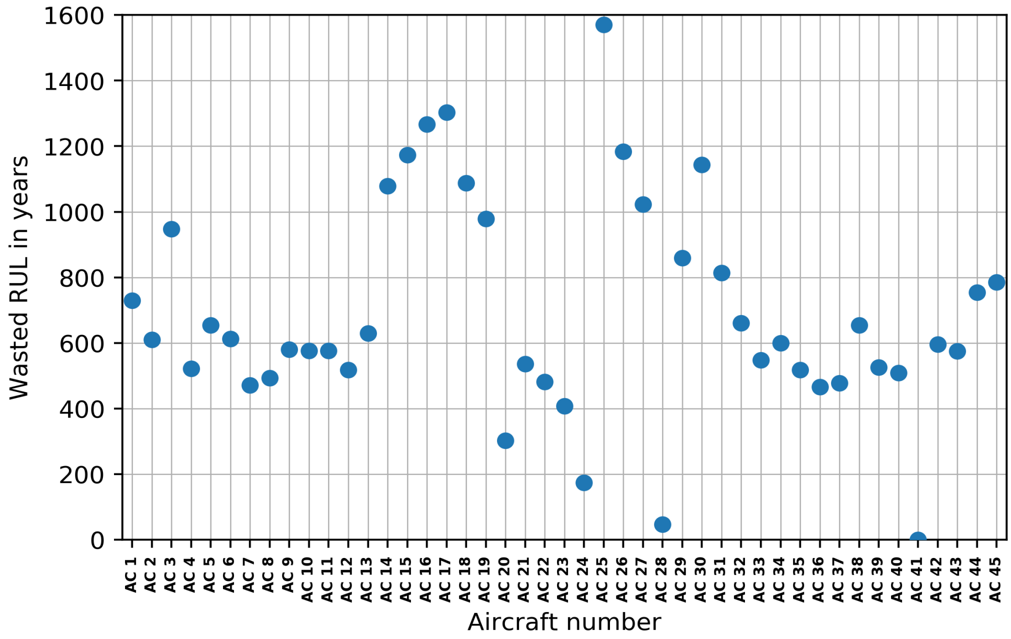

Figure 7.1: Exact method: RUL wasted during considered time horizon per aircraft tail number.

that the phased-out aircraft (i.e. AC-24, AC-28, AC-41) receive significant less tasks than the other aircraft. The last main solution parameter is the distribution of man-hours per skill type as presented

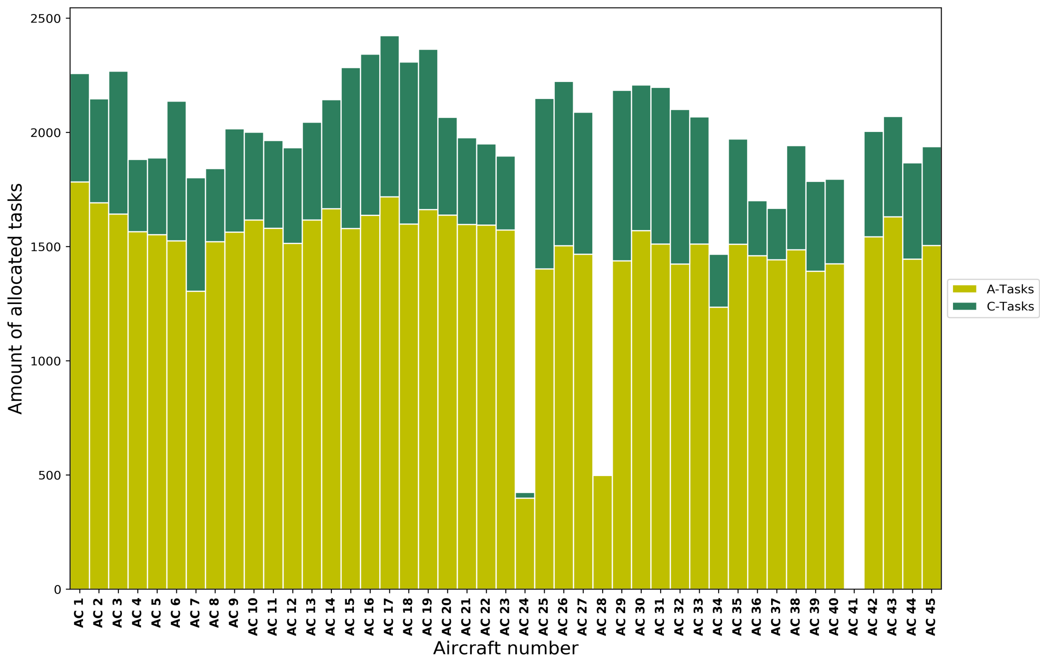

Figure 7.2: Exact method: Number of tasks allocated during considered time horizon per aircraft tail number.

in figure 7.3. Results indicate that the GR2 skill type is the most restricted skill type. When we analyze the average man-hours required per A-check tasks and C-check tasks, we can observe that on average an A-check tasks takes 0.57 man-hours and a C-check tasks takes 3.43 man-hours. Thus, the C-check tasks comprise the majority of man-hours used despite their much lower per cent of the total allocated tasks.

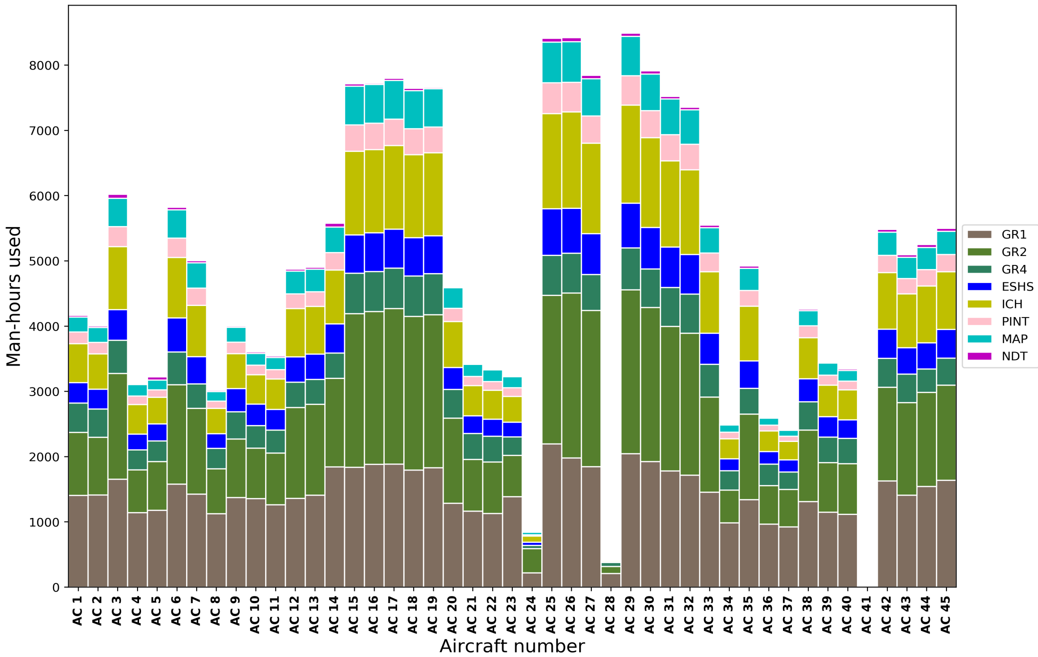

Figure 7.3: Exact method: Total amount of man-hours per skill type used during considered time horizon per aircraft tail number.

7.1.4. Output tactical stage

The output of the tactical stage model is an excel file consisting of the used and remaining man-hours per maintenance time segments for each considered aircraft. This file will be used as input to the operational level stage 1 model. A snapshot of the excel file can be seen in Appendix A.

7.1.5. Discussion results

The analysis of the results suggest that C-check tasks waste 205 days RUL on average. This is caused by the way the maintenance schedule is generated and due to operational constraints imposed by the airline. The dynamic programming based approach, used to generate a maintenance schedule, will schedule a new C-check block before an aircraft reaches its minimal C-check task intervals: 7,500 flight-hours, 5,000 flight-cycles or 24 months. This means that the maintenance schedule is optimized only for tasks having the minimal C-check task intervals. Given that the calculation of the tasks costs is based on the RUL and man-hours it may be concluded that C-check tasks contribute a major part of the final objective function value. Therefore, the C-check tasks should be the primary focus in order to reduce the objective function value of the task allocation problem. Since the man-hours of a C-check task can not be adjusted the only way to reduce the objective function value is to reduce the RUL of C-check task by a better distribution of the C-check maintenance opportunities. The proposed distribution would be to split-up the C-check in two blocks as can be seen in figure 7.4.

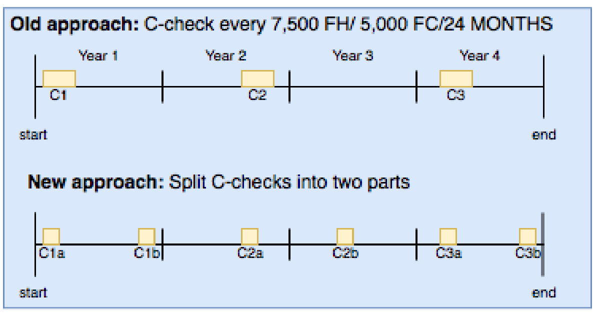

Figure 7.4: Visualization split C-check maintenance concept.

In order to support this new approach the tasks with a flight-hour limit between 7,500 and 15,000 were grouped in the following categories:

- Group 1: Tasks having a flight-hour limit interval between 7,500 and 9,000 flight-hours
- Group 2: Tasks having a flight-hour limit interval between 9,001 and 10,500 flight-hours.
- Group 3: Tasks having a flight-hour limit interval between 10,501 and 12,000 flight-hours.
- Group 4: Tasks having a flight-hour limit interval between 12,001 and 13,500 flight-hours.
- Group 5: Tasks having a flight-hour limit interval between 13,501 and 15,000 flight-hours.

The distribution of tasks that fall in the mentioned groups, for the entire fleet, are presented in figure 7.5. From this figure it becomes visible that the maintenance schedule is only optimized for the group 1 and group 5 tasks. Due to operations constraints the gap between two consecutive C-checks is mostly around 7,000 flight-hours. This means that in most cases group 2, group 3 and group 4 maintenance tasks will be performed at a C-check maintenance opportunities which is placed around every 7,000 flight-hours. The results of this approach is that those tasks will waste up to 40 per cent of their RUL. It is expected that the proposed split C-check maintenance approach will allocate tasks, belonging to group 2 and 3, to the second part of the split C-check thereby reducing the wasted RUL significantly. Unfortunately, the proposed split C-check approach could not be tested due to the extra days needed to prepare the maintenance tools for checks in this test case. Splitting C-check approach will incur more days spent on preparation, therefore increasing the elapsed time of each C-check.

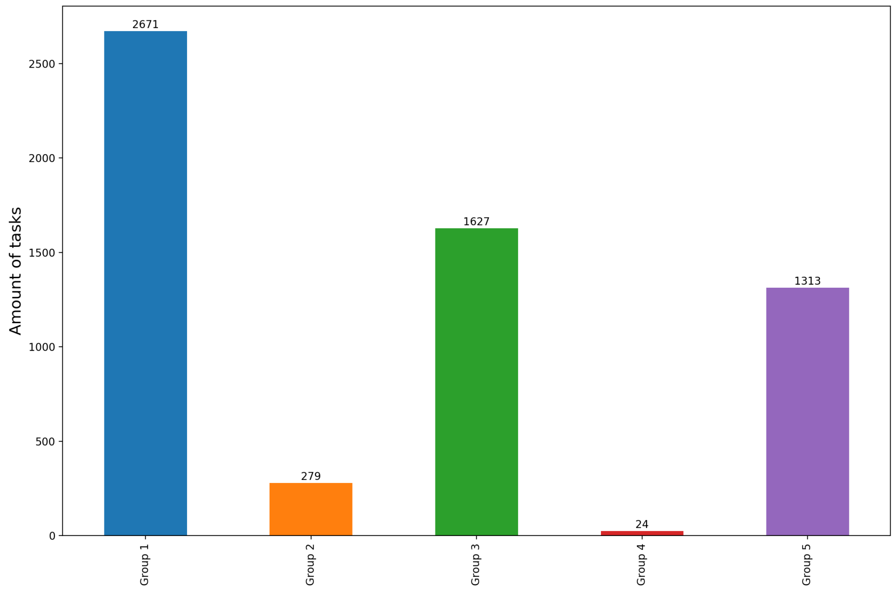

Figure 7.5: Number of allocated tasks per group.

7.2. Exact method - results operational stage level 1

This section presents the results of applying the proposed operational level 1 model to data from a single aircraft. The main input to the model will be the obtained man-hours distribution per aircraft from the tactical stage solution.

7.2.1. Problem size and computational time

It was decided to choose AC-16 as the one to indicate the problem size. The problem size is obtained from Gurobi and indicates that the model has 31,580 decision variables and 8,487 constraints. From which 2,621 are type 1 constraints (equation 6.2), 200 are type 2 constraints (equation 6.3) and 5,666 are type 3 constraints (equations 6.4 - 6.7). To solve the considered time horizon for AC-16 the operational model level 1 takes 60 seconds on the mentioned laptop. The final objective function value is equal to 41,554.

7.2.2. Visualization solution parameters

The results of applying the model to AC-16 are presented in figure 7.6, in which the number of allocated tasks[1] per maintenance opportunities are given. It is important to notice that the first five maintenance opportunities remain empty because the tasks excel file, provided by the airline, was updated on the 13th of August 2018. The results indicate that the C-check maintenance opportunities receive significantly more tasks than A-check maintenance opportunities. Furthermore, a visualization

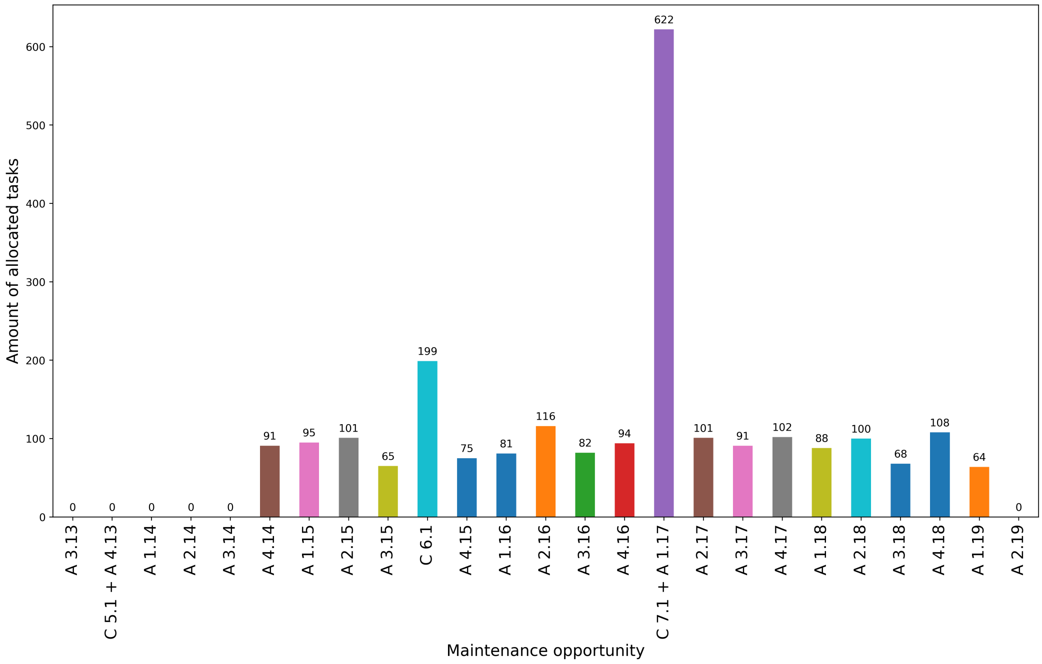

Figure 7.6: Amount of tasks allocated per maintenance opportunity for aircraft 16.

of the man-hours constraints is given in figure 7.7 till 7.11. In these figures the red transparent bars indicate the maximum available man-hours for each of the eight different skill types per maintenance opportunity. When we compare the used man-hours with the available man-hours we can observe that A-check maintenance opportunities have plenty of man-hours remaining. The C-check maintenance opportunities seem to be a bit more restricted (i.e. less remaining man-hours available). When we analyze the difference in the available man-hours and the used man-hours per skill type, we can observe that in most occasions the skill type GR2 is the most restricted skill type. This is especially visible in the last C-check maintenance opportunity of AC-16. It was mentioned in section 4.1.3 that A-check opportunities are not allowed to use skill types ESHS and PINT. However, the tasks excel file, provided by the airline, indicates that there are a few A-check tasks that require the skill type ESHS. Therefore, the model used the available man-hours of skill type ESHS and PINT at the specific day from the HM department.

In addition, the model was executed separately for each aircraft in the fleet. Figure 7.12 displays the final objective function values for each of the aircraft.

7.2.3. Output operational level 1

The output of the operational level 1 model is an excel file consisting of the allocated tasks per maintenance opportunity. In the next operational level these tasks will be allocated down to shift level. A snapshot of the excel file can be found in Appendix B.

[1]The visualized tasks are not decoupled yet, this will happen at the last operational level model.

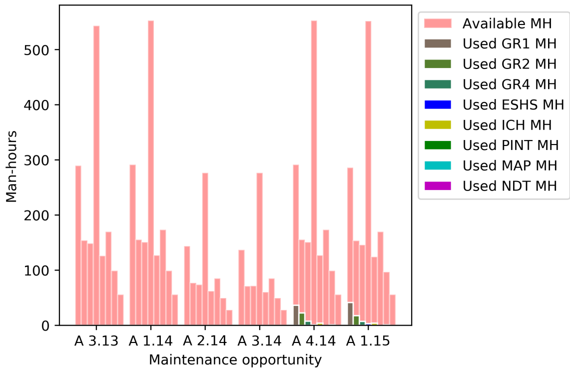

Figure 7.7: Used man-hours for the first set of A-checks.

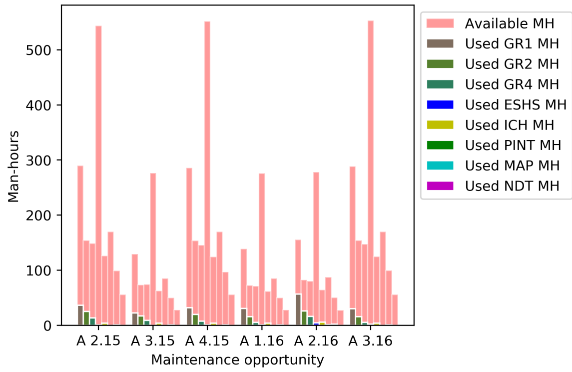

Figure 7.8: Used man-hours for the second set of A-checks.

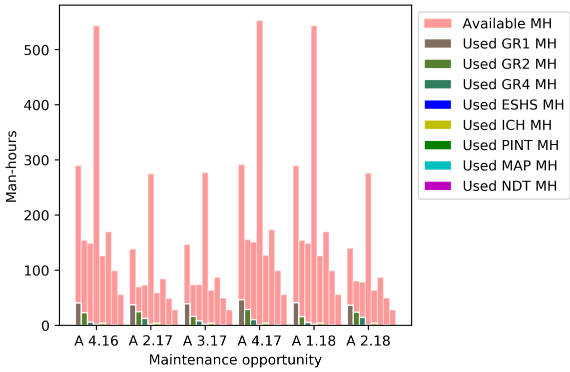

Figure 7.9: Used man-hours for the third set of A-checks.

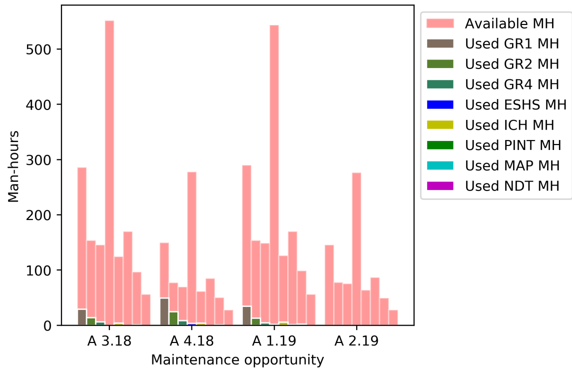

Figure 7.10: Used man-hours for the last set of A-checks.

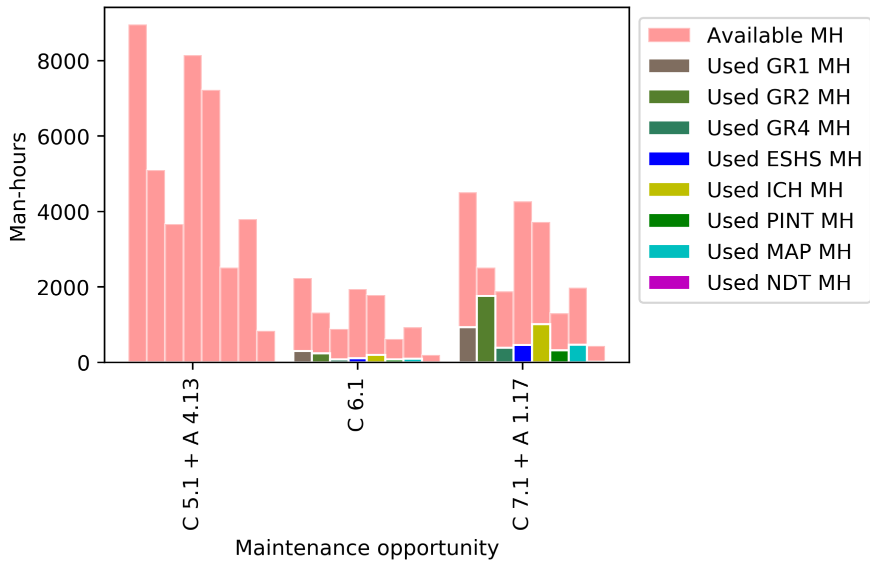

Figure 7.11: Used man-hours for the C-checks.

7.2.4. Discussion results

The analysis of the results suggest that there is a significant difference in the final objective function value among some of the aircraft. Especially, aircraft 15,16,17,18,19 and 26 have a significant higher objective function value compared to the average. This is caused by either one of the following reasons:

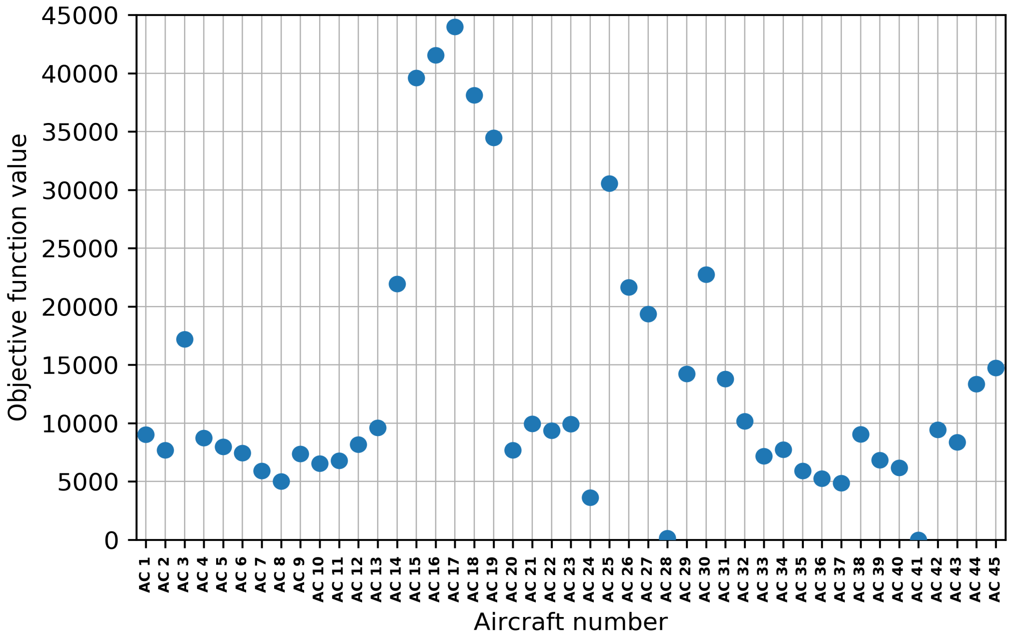

Figure 7.12: Exact method: Objective function value per aircraft tail number.

there are more C-check maintenance opportunities scheduled, the placement of some of the maintenance opportunities of these aircraft were done much earlier than their C-check minimum intervals due to requirements from the airline, or the aircraft reached a certain age whereby tasks need to be executed more frequently. The analysis of the results suggest that the airline has plenty of man-hours remaining on each of the maintenance opportunities. Hence, the capabilities of the models in more restricted cases should be tested. This is done in the sensitivity analysis chapter. In section 9.1 the available man-hours per maintenance opportunity are decreased by a constant factor. Furthermore, in section 9.2 the man-hours required for the C-check tasks were slowly increased by a constant factor.

7.3. Exact method - results operational stage level 2

This section presents the results of applying the proposed operational stage level 2 model on data of aircraft 16. The model was run twice, with different starting dates, to indicate the problem size differences between A-check and C-check maintenance opportunities. It is worth mentioning that the exact man-hours distribution per time segment, obtained from the solution at the tactical level, is used for the man-hours constraints of each of the shifts.

7.3.1. Problem size

The problem size for C-check maintenance opportunity C6.1, of aircraft 16, is obtained from Gurobi and indicates that the model has 11,691 decision variables and 1,223 constraints. From which 433 are type 1 constraints (equation 6.10), 216 are type 2 constraints (equation 6.11) and 574 are type 3 constraints (equation 6.12). The task allocation problem, for one C-check, was solved in 35 seconds by Gurobi. For the A-check maintenance opportunity A4.14 the model has 492 decision variables and 318 constraints. From which 164 are type 1 constraints, 24 are type 2 constraints and 133 are type 3 constraints. The task allocation problem, for one A-check, was solved in 5 seconds by Gurobi.

7.3.2. Visualization solution parameters

The results of running the operational level 2 model for maintenance opportunity A4.14 are presented in figure 7.13, in which the number of tasks per shift are given. At the operational level 1 it was given that 91 tasks were allocated to maintenance opportunity A4.14. However, figure 7.13 indicate that there are 164 tasks allocated this is caused by the fact that the tasks at the level 1 stage were not decoupled, while the solution from the level 2 model decoupled the tasks. Also, 13 additional tasks were added for opening and closing access panels as can be seen in table 7.1. The used man-hours per skill type per shift are presented in figure 7.14 where it is indicated that there are plenty of man-hours remaining at each of the shifts. This is expected given that maintenance opportunity A4.14 had plenty of man-hours remaining at the operational level 1 solution.

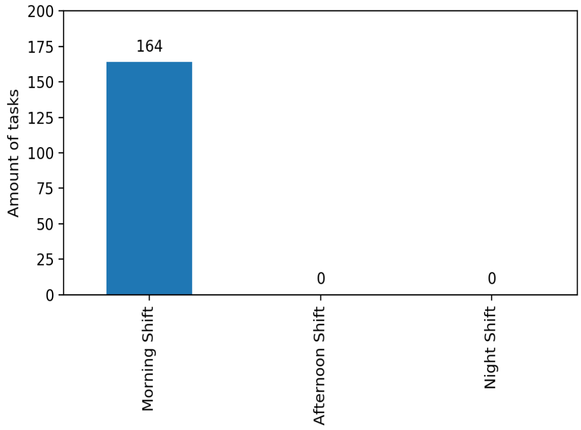

Figure 7.13: Amount of tasks allocated per shift.

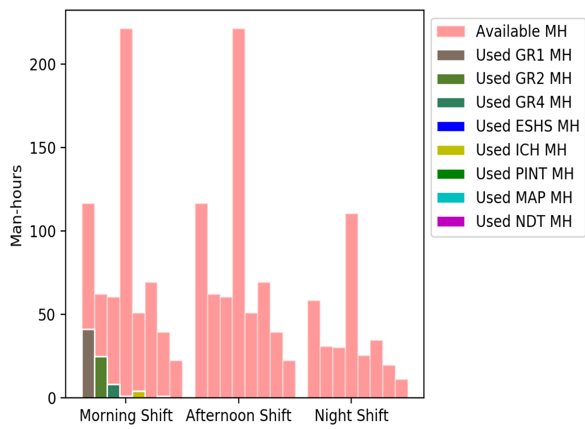

Figure 7.14: Visualization man-hours used per shift.

# Task	Panels	Linked task #
138	825	17, 19, 82, 97
139	841	102
140	831	100
141	316AR, 315AL	66, 122
142	714, 713	48, 52, 81, 88
143	832	101
144	842	103
145	826	18, 20, 21, 84, 85, 98
146	437AL, 438AR, 452AR, 451AL	39, 68, 74, 132
147	448AR, 447AL, 461AL, 462AR	12, 40, 69, 75, 133
148	827	99
149	152KW, 151KW	22
150	744, 734, 196BB, 195BB, 197CB, 197FB, 147EB	41, 42, 43, 44, 46, 47, 53, 54, 56, 83, 95, 96

Table 7.1: Added access panels tasks and tasks numbers linked to it.

7.3.3. Output operational stage level 2

The output of the operational stage level 2 model is an excel file consisting of the tasks, with their corresponding task information, allocated to each of the shifts. This file can directly be used by the maintenance manager to distribute the tasks, for each shift, among the available workforce per shift. A snapshot of the excel file can be seen in Appendix C.

7.3.4. Discussion results

The results indicate that there are additional tasks, for the opening and closing of access panels, inserted which were previously not considered. The total man-hours required for these tasks are around 4 per cent of the total man-hours during A-check maintenance opportunities and 3 per cent of the total man-hours during C-check maintenance opportunities. Given the low per cent of required man-hours for these additional tasks it was decided to not include those at earlier levels of the task allocation framework, because this would explode the problem size. A possible solution to tackle this problem is to deduct a constant amount of man-hours at each of the maintenance opportunities at the tactical stage model and operational level 1 model. This is a reasonable approach since the used man-hours for opening and closing of access panels is more or less the same at most of the maintenance opportunities. Given that the exact level stage 2 was solely used to indicate the performance of the approximation algorithm, at this same stage, this factor was not included. The approximation algorithm does have the possibility to add fictitious man-hours, which is reasonable since airlines are flexible in their use of personnel. It is worth mentioning that this problem only occurs in very restricted cases (i.e. almost all man-hours during a maintenance opportunity are used), which did not happen for each of the considered aircraft during this case study.

8

Validation and Verification

This chapter will present the verification and validation for each of the proposed exact and approximation models. Given that there were plenty of man-hours remaining, at each maintenance opportunity, it was possible to verify the approximation models by comparison with the obtained solutions from the exact methods.

8.1. Exact method: Tactical stage model

This section deals with the verification and validation of the exact model for the tactical stage of the task allocation framework.

8.1.1. Verification

An elemental way to check if the model corresponds to the mathematical formulation is to check the model size. The total amount of decision variables for the tactical stage model should equal the sum of the possible maintenance opportunities for each task having a calculated due-date before the last maintenance opportunity of the corresponding aircraft. The total amount of constraints consist of three different types. For each of the allocated tasks there should be an assignment constraint present. The second constraint type should be equal to the amount of considered time segments, for the entire fleet, times the eight different skill types. The preceding relation between scheduled and rescheduled tasks should incorporate a constraint for each of the interval limits of the considered tasks. Therefore, the expected model size is as follows:

- Problem size: $|K|$ = number of aircraft considered, $|T_k|$ = Number of tasks having a calculated due-date before the last maintenance opportunity day for aircraft k, $|O(i)^k|$ = Maintenance opportunities possible for task i of aircraft k, $|GR|$ = Amount of different skill types, $|Z|$ = Amount of different maintenance time segments, $|S_k|$ = Amount of linked tasks for aircraft k, $|Q_k|$ = Amount of different tasks intervals within the linked tasks for aircraft k.
  - Constraints: $\sum_{k \in K}(|T_k| + |S_k| \times |Q_k|) + |GR| \times |Z|$
    - $103,779 + 217,210 + 6,912 = 327,901$
  - Variables: $\sum_{k \in K} \sum_{i \in I} |O(i)^k|$
    - $1,154,822$

In the results section 7.1 it was mentioned that there are 327,901 constraints from which 103,779 are type 1 constraints (equation 5.2), 6,912 are type 2 constraints (equation 5.3) and 217,210 are type 3 constraints (equations 5.4 - 5.7). It was also mentioned that a total of 86,289 tasks were allocated, which seems incorrect since there are 103,779 constraints of type 1. However, it must be noted that any tasks allocated to the fictitious maintenance opportunities have been removed from the final solution. Running the model again, now without removing the tasks allocated to the fictitious maintenance opportunities, resulted in a total of 103,779 allocated tasks, which is exactly the same as the amount of type 1 constraints. The number of second constraint types should equal the amount of different maintenance time segments times the available skill types. During the considered time horizon there are 155 different time segments for C-checks maintenance opportunities and 709 time segments for A-check maintenance opportunities. This means that there should be $(155 + 709) \times 8 = 6,912$ constraint of type 2, since there are eight different skill types. In order to verify the third constraint type it is necessary to obtain the amount of different interval limits for each of the rescheduled tasks, including those allocated to the fictitious maintenance opportunities. Figure 8.1 gives an overview of the amount of interval limits for the tasks that need to receive type 3 constraints. The results suggest that there should be 108,605 $(1 \times 50,275 + 2 \times 6,623 + 3 \times 15,028)$ different constraints of type 3. This is exactly half the amount of type 3 constraints used in the Gurobi model. This is caused by the fact that it was not possible to present both an upper and lower bound for each of the type 3 constraints. Therefore, the Gurobi model will have a constraint for the upper bound and another one for the lower bound for each type 3 constraint. Thus, it may be concluded that the expected amount of constraints, for each constrain type, is met. The total amount of decision variables should be equal to all 1's within the due-date sets for each of the considered aircraft. Summing these values led to 1.15 million decision variables which is equivalent to the amount mentioned in the results section.

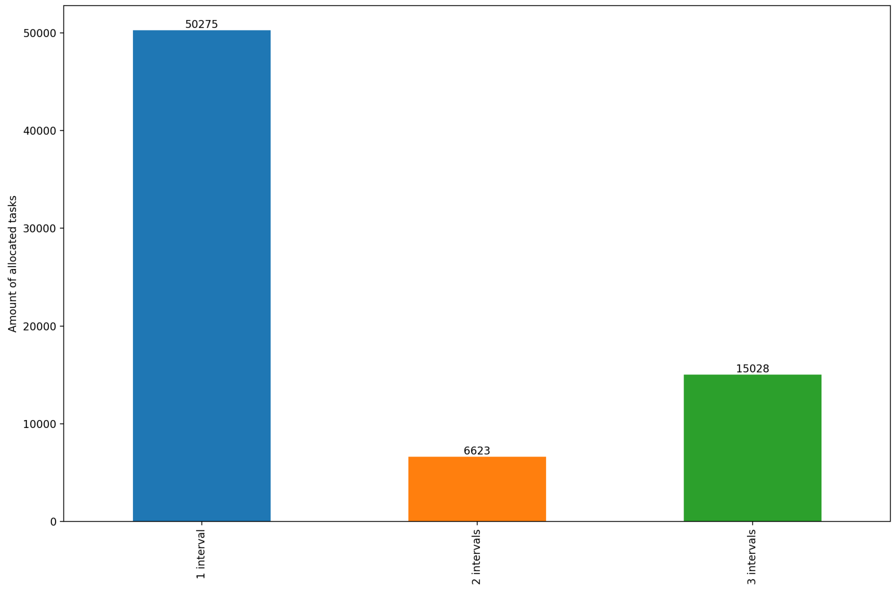

Figure 8.1: Amount of interval limits for tasks having a linked constraint.

In addition, the second constraint type can be verified by plotting the available man-hours against the used man-hours for each of the maintenance days. Given that the problem size is large, it is only possible to visualize the second constraint type by taking a snapshot of the problem as can be seen in figure 8.2. In this figure there are seven time segments, which are shared among five aircraft. The obtained results of the tactical stage model indicates the used man-hours per skill type, per aircraft, for each of the time segments as presented in figure 8.3. In the figure there are eight different bars per time segment given, these bars represents the different skill types in the following sequence: GR1, GR2, GR4, ESHS, PINT, MAP, and NDT. As with the earlier presented figures the red transparent bars indicate the maximum available man-hours. When we analyze the figure it becomes evident that only time segment 1,2 and 6 use the full capacity of skill type GR2 during the time horizon. Thus, the maximum available man-hours, per skill type, is not exceeded on each of the time segments.

8.1.2. Validation

No similar studies, examples or test cases from the airline could be compared. Therefore, expert opinion is used for validation of the model. The validation of the model is led by an expert of the airline with respect to the task allocation for the entire fleet. The expert has successfully validated the model by controlling a subset of the tasks allocated. It is worth mentioning that it was not possible to compare the obtained solution with the current maintenance schedule of the airline since the airline does not have results for a time horizon beyond a few months.

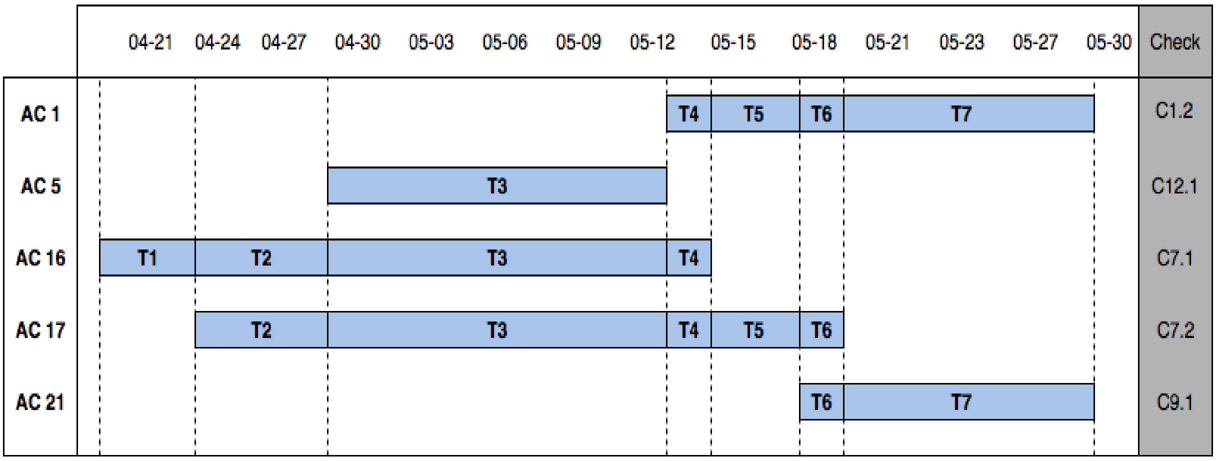

Figure 8.2: Snapshot of maintenance overlap between aircraft.

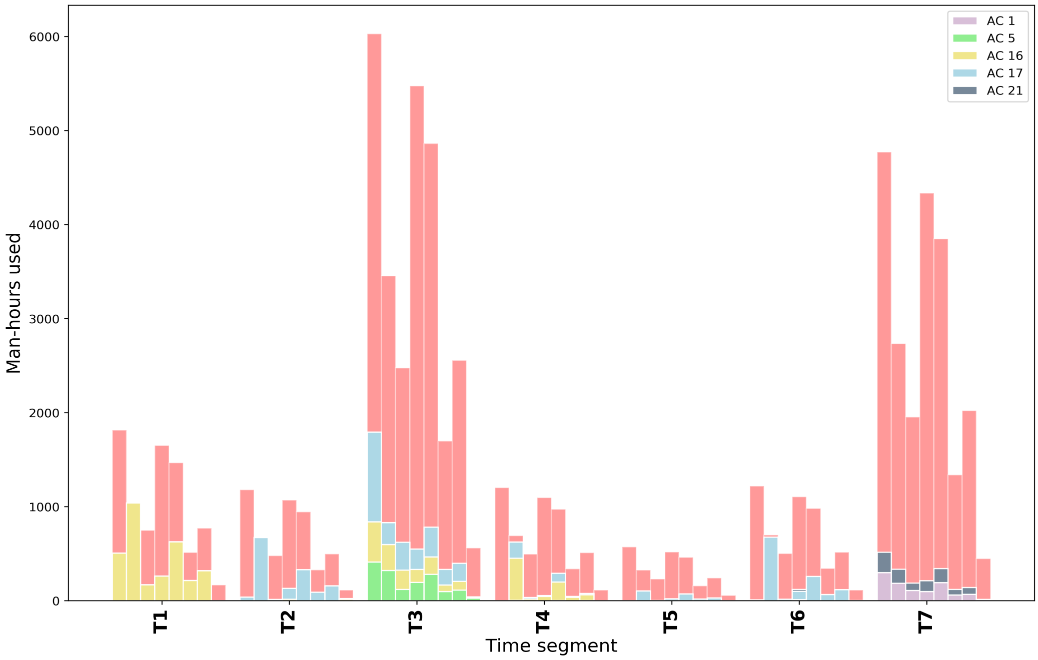

Figure 8.3: Man-hours per skill type used per time segment.

8.2. Exact method: Operational stage level 1 model

This section deals with the verification and validation of the exact model for the operational stage level 1 of the task allocation framework.

8.2.1. Verification

An important step in the verification of the operational stage model level 1 is to see if the problem size is exactly as expected. For this model the total amount of decision variables should be equal to the sum of the amount of possible maintenance opportunities for each of the tasks. When looking at the amount of constraints it is important to look at the different constraints separately. For each of the tasks scheduled there should be an assignment constraint present (i.e. type 1 constraint). This means that the amount of scheduled tasks should equal the amount of assignment constraints. Given that there are eight different skill types, there need to be eight man-hours constraints (i.e. type 2 constraint) for each maintenance opportunity. The preceding relation between scheduled and rescheduled tasks should incorporate a type 3 constraint for each of the interval limits of the considered tasks. This results in the following problem size for the task allocation model of one single aircraft:

- Problem size: $|T|$ = Number of tasks having a calculated due-date before the last maintenance opportunity day, $|O(t)|$ = Maintenance opportunities possible for task t, $|GR|$ = Amount of different skill types, $|Z|$ = Amount of different maintenance opportunities, $|S|$ = Amount of linked tasks, $|I|$ = Amount of different tasks intervals.
    - Constraints: $|T| + |GR| \times |Z| + |S| \times |I|$
        - $2,621 + 25 \times 8 + 2,833 + 5,666 = 8,487$
    - Variables: $\text{sum}_T\ |O(t)|$
        - $31,580$

The problem size will be verified with the results obtained from solving the operational level 1 model with AC-16 as presented in section 7.2. In this section it was mentioned that there are 31,580 decision variables and 8,487 constraints. From which 2,621 are type 1 constraints (equation 6.2), 200 are type 2 constraints (equation 6.3) and 5,666 are type 3 constraints (equations 6.4 - 6.7). In order to verify the first constraint type it is necessary to look at the total amount of allocated tasks. At the final solution there are 2,343 tasks allocated, which seems to be incorrect since there are 2,621 type 1 constraints. However, as with the tactical level model, the tasks allocated to the fictitious maintenance opportunities were removed. Thus, in order to verify the first constrain type the model was run again now without removing the fictitious maintenance opportunities. The results of this new run are visualized in figure 8.4, where the amount of tasks allocated to each of the maintenance opportunities are given. From the analysis of this figure it becomes evident that there are 278 tasks allocated to fictitious maintenance opportunities. This means that a total of $2,343+278 = 2,621$ tasks were allocated which is equal to the amount of constraints of type 1. The second constraint type should equal the amount of maintenance opportunities (i.e. excluding fictitious maintenance opportunities) times the different skill types. There are 25 real maintenance opportunities for AC-16 as can be seen in figure 8.4. Furthermore, there are eight different skill types which means that there should be $25 \times 8 = 200$ constraints of type 2. The Gurobi model confirms that this is indeed the number of constraints used in the model. In order to verify the third constraint type it is necessary to search the interval characteristics of each of the allocated tasks. Figure 8.5 indicate that there are 1277 tasks with a single interval limit, 166 tasks with two interval limits and 408 tasks with three interval limits. Thus, the results suggest that there should be $1 \times 1,277 + 2 \times 166 + 3 \times 408 = 2,833$ constraints of the third type. Which is exactly half the amount of type 3 constraints used in the Gurobi model. This is caused by the fact that it was not possible to present both an upper and lower bound for each of the type 3 constraints. Therefore, the Gurobi model will have a constraint for the upper bound and another one for the lower bound per type 3 constraint. This concludes that the expected amount of constraints, for each constraint type, is met. The total amount of decision variables should be equal to all 1's within the due-date set for the considered aircraft. Summing these 1's led to a total of 31,580 decision variables. Which is equivalent to the amount of decision variables mentioned in the results sections. Thus, it may be concluded that the problem size is exactly as expected.

8.2.2. Validation

The first operational level model uses the same code as the tactical stage model, with the only difference being that there is a single aircraft selected. Therefore, the model validation was already done at section 8.1.2, where an expert of the airline successfully validated the tactical stage model.

8.3. Exact method: Operational stage level 2 model

This section deals with the verification and validation of the exact model for the operational stage level 2 of the task allocation framework.

8.3.1. Verification

An important step in the validation of the operational stage model level 2 is to see if the problem size is exactly as expected. The total amount of decision variables should be equal to the sum of the amount of possible shifts for each of the allocated tasks at the considered maintenance opportunity.

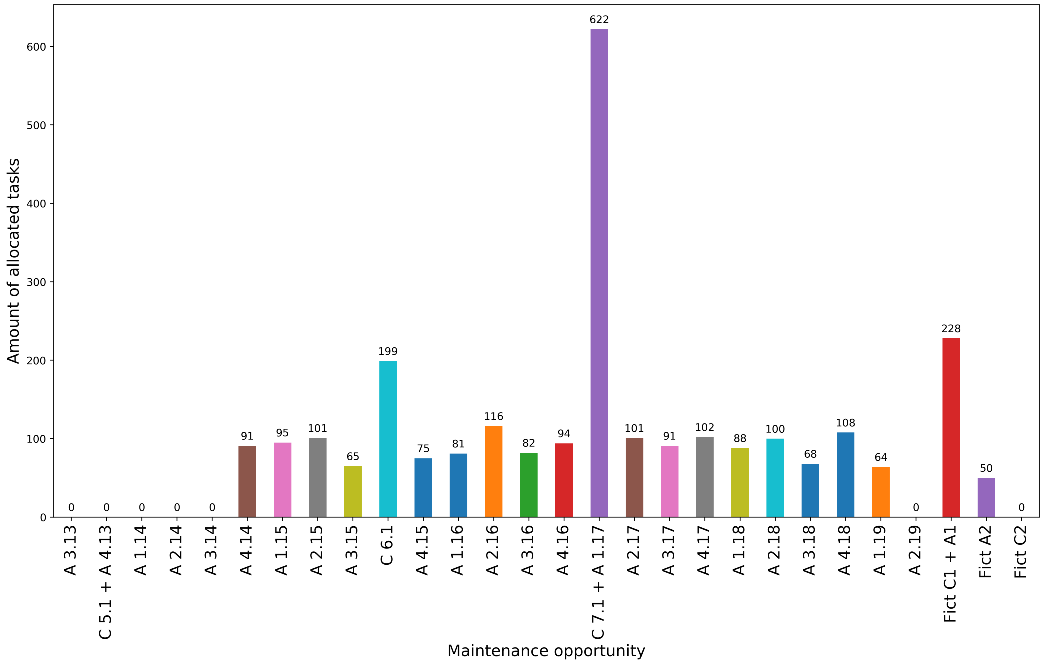

Figure 8.4: Amount of tasks allocated per maintenance opportunity (including fictitious maintenance opportunities) for aircraft 16.

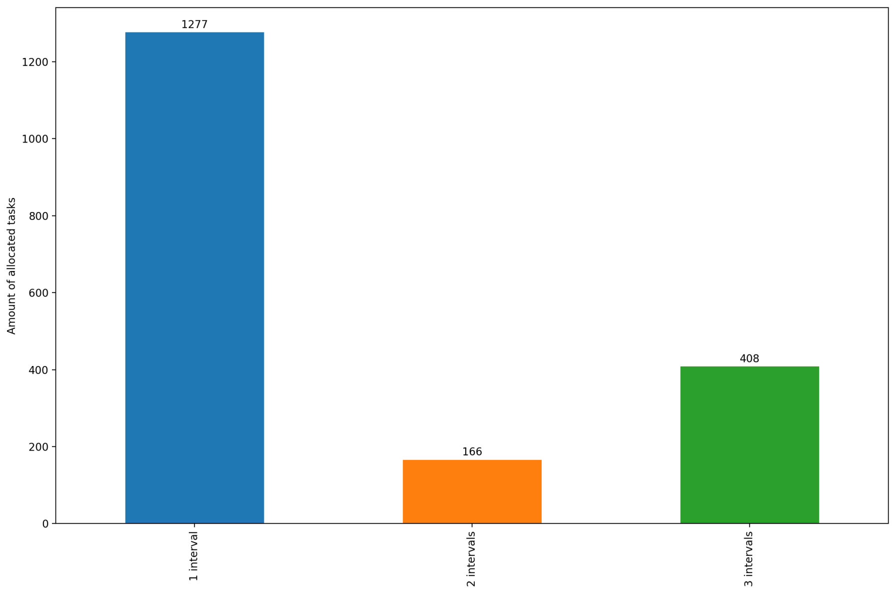

Figure 8.5: Amount of different interval limits for the allocated tasks.

When looking at the amount of constraints it is important to look at the different constraints separately. For each of the tasks scheduled there should be an assignment constraint present. This means that the amount of scheduled tasks should be equal to the amount of assignment constraints. Since there are eight different skill types available during each of the shifts, there need to be eight man-hours constraints per shift. The preceding relation between tasks should be incorporated. This means that any task that require the opening or closing of access panels and tasks that should be executed in series should receive a constraint of the third type. Therefore, the expected model size is as follows:

- Problem size: $|T|$ = Number of scheduled tasks, $|S|$ = Number of shifts, $|GR|$ = Amount of different skill types, $|K|$ = Amount of linked tasks.

- Constraints: $|T| + |S| \times |GR| + |K|$
  - $168 + 3 \times 7 + 133 = 322$
- Variables: $|T| \times |S|$.
  - $164 \times 3 = 492$

In the results section 7.3 it was mentioned that there are 492 decision variables and 318 constraints from which 164 are type 1 constraints (equation 6.10), 24 are type 2 constraints (equation 6.11) and 133 are type 3 constraints (equation 6.12). According to the problem size the amount of decision variables should equal the number of tasks allocated times the number of shifts. The results of the operational level 1 model indicate that 138 tasks are allocated[1] to maintenance opportunity A4.14. This means that there should be $138 \times 3 = 414$ decision variables, which is 78 less than the number of decision variables used in the Gurobi model. This difference is caused by the additional tasks which were added in the access panel loop. In this case 26 tasks have been added, 13 for opening the access panels and 13 for closing them, as was shown in table 7.1. Adding these tasks result in a total of 164 tasks. Given that there are three different shifts, during maintenance opportunity A4.14, there should be $3 \times 164 = 492$ decision variables. Which is exactly the same amount as used in the Gurobi model. When considering the type 1 constraints, there should be 164 assignment constraints, one for each of the allocated tasks. The results indicate that there are indeed 164 constraint of type 1 present in the model. Given that there are three shifts and eight different skill types there should be $3 \times 8 = 24$ constraints of type 2, which is equivalent to the amount of constraints of type 2 used in the Gurobi model. Furthermore, there are 47 tasks that need to be executed in series and 43 tasks that can only be executed if the access panels are open. Tasks that are related to the opening of access panels will receive two linked constraints, one to ensure that the access panels are opened before the tasks are executed and another one to make sure that the panels are only closed if each of the tasks requiring those access panels is executed. This means that there should be $47 + 43 \times 2 = 133$ constraints of the third type, which is equal to the amount of constraints of type 3 used in the model. Hence, it is concluded that the expected amount of constraints, for each constraint type is met.

8.3.2. Validation

Given that the problem size of the considered maintenance opportunity A4.14 is limited it was possible to manually validate the model. The first thing to validate is that the correct access panels are selected for each of the tasks that require the opening of access panels. Each of the allocated tasks to maintenance opportunity A4.14 were checked and the ones requiring the opening of an access panel are displayed in table 8.1. In order to prevent the repetitive opening or closing of access panels it is essential to find tasks that require the same access panels. The results of this is displayed in table 8.2, in which there are twelve additional tasks created each for the opening of specific access panels group. The linked tasks numbers in the last column of table 8.2 will all receive two linked constraints, one to ensure that the access panels are opened prior to the execution of the respective tasks and another one to make sure that the access panels are closed after each of the tasks is executed. In addition, the main thing to validate is that each of the tasks scheduled at the operational level 1 are allocated to shifts at the operational level 2. The output of the operational level 1 indicates that 138 tasks were scheduled on A4.14, while the output of the operational level 2 model indicates a total of 168 allocated tasks. This is caused by the fact that there are 13 new tasks created for opening access panels and 13 to close access panels. In addition, it was checked that the same unique tasks reference numbers, from operational level 1, were also present in the solution of the operational level 2.

8.4. Validation approximation algorithms

The case study results indicate that the man-hours are not the main constraint to the problem, for each of the maintenance opportunity dates. This makes it possible to validate the approximation algorithms by comparison with the solutions from the exact methods. Each of the approximation models should give the same solution and parameters as the exact models for the data used during the case study. The approximation algorithms were solved on an Intel Core i5 1.6 GHz laptop with 4GB ram. This laptop was also used for the exact methods.

[1]Including the decoupled tasks

# Task	Panels	#Task	Panels	# Task	Panels
12	448AR	48	713, 714	88	714, 713
17	825	52	713, 714	95	734
18	826	53	734	96	744
19	825	54	744	97	825
20	826	56	744	98	826
21	826	66	316AR, 315AL	99	827
22	152KW, 151KW	68	438AR, 437AL	100	831
39	437AL	69	447AL, 448AR	101	832
40	447AL	74	437AL	102	841
41	196BB, 195BB, 197CB, 197FB, 734	75	447AL	103	842
42	197FB, 196BB, 744	81	714, 713	122	315AL
43	744, 196BB, 197FB	82	825	132	438AR, 437AL, 451 AL, 452AR
44	744, 734	83	147EB, 734, 744	133	416AL, 448AR, 462AR, 447AL
46	734	84	826		
47	744	85	826		

Table 8.1: Tasks that require the opening of access panels.

# Task	Panels	Linked task #
138	825	17, 19, 82, 97
139	841	102
140	831	100
141	316AR, 315AL	66, 122
142	714, 713	48, 52, 81, 88
143	832	101
144	842	103
145	826	18, 20, 21, 84, 85, 98
146	437AL, 438AR, 452AR, 451AL	39, 68, 74, 132
147	448AR, 447AL, 461AL, 462AR	12, 40, 69, 75, 133
148	827	99
149	152KW, 151KW	22
150	744, 734, 196BB, 195BB, 197CB, 197FB, 147EB	41, 42, 43, 44, 46, 47, 53, 54, 56, 83, 95, 96

Table 8.2: Additional opening access panels tasks, including linked tasks numbers.

8.4.1. Approximation algorithm: Tactical stage

To solve the considered planning horizon, for the entire fleet, the tactical stage approximation model takes 840 seconds on the mentioned laptop. The solution indicates that 65,531 A-check tasks and 20,758 C-check tasks have been allocated. The number of tasks allocated, to each aircraft, is exactly the same as the solution from the exact method. An illustration of the distribution of A-tasks and C-tasks per aircraft is given in figure 8.6. The total man-hours used, for the entire fleet, corresponds to a total of 221,574 man-hours. A distribution of the man-hours per skill type for each aircraft is given in figure 8.7. The distribution per aircraft as well as the total man-hours used is identical with the solution from the exact method. This concludes, that the output of the approximation tactical stage model, which is the available man-hours for each skill type per maintenance time segment, is exactly the same as the solution from the exact method. The only difference between the two models is the final objective function value, which is just 167 higher for the approximation algorithm compared to the solution of the exact method. This difference corresponds to 0.02 per cent which is negligible and was therefore not investigated.

8.4.2. Approximation algorithm: Operational stage level 1

Since there is no additional input at the operational level 1 (e.g. additional tasks or utilization correction) the man-hours distribution per time segment, obtained from the tactical stage solution, is used as only

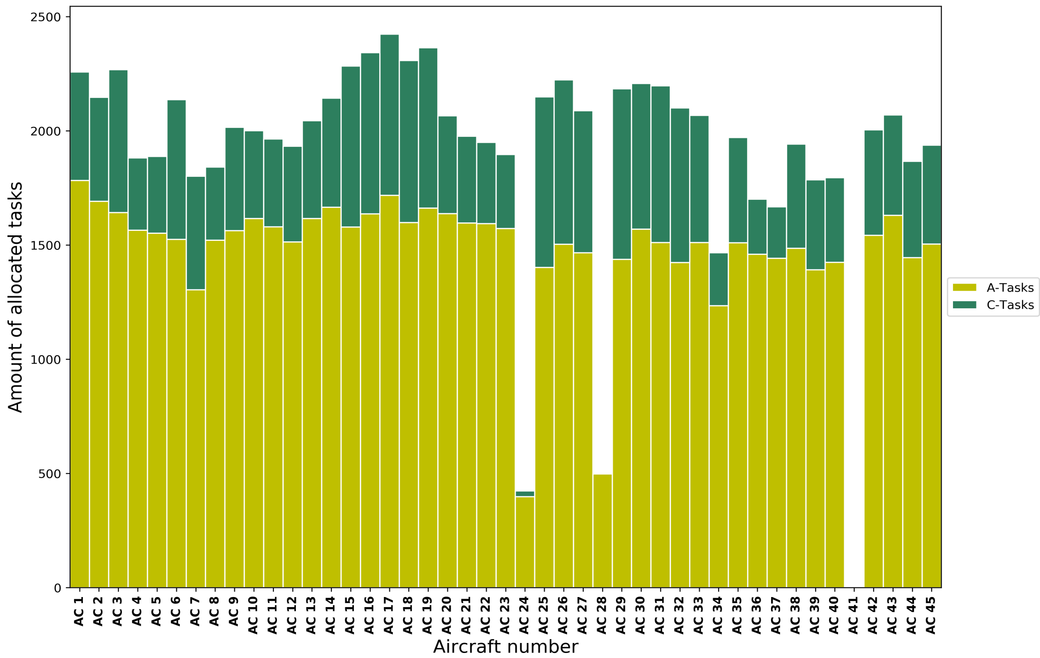

Figure 8.6: Approximation algorithm: Number of tasks allocated during considered time horizon per aircraft tail number.

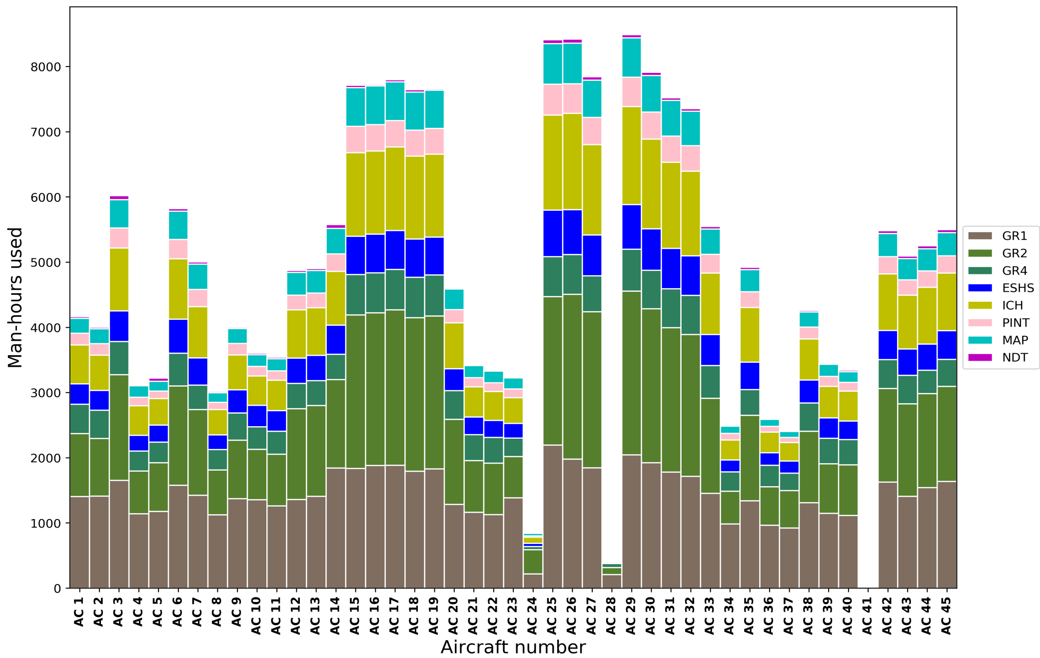

Figure 8.7: Approximation algorithm: Number of tasks allocated during considered time horizon per aircraft tail number.

new input. The operational level 1 model was run for each of the aircraft and the corresponding RUL wasted and the objective function values were compared to the results from the exact method as can be seen in figure 8.8 and 8.9, respectively. The figure indicates that the calculated objective function value, for each aircraft, is equivalent to the ones presented for the exact method. In addition, the number of tasks allocated and man-hours used per skill type for each of the aircraft was exactly the same as the solution of the exact method. This can be expected since there was a surplus of man-hours available on each of the maintenance opportunities.

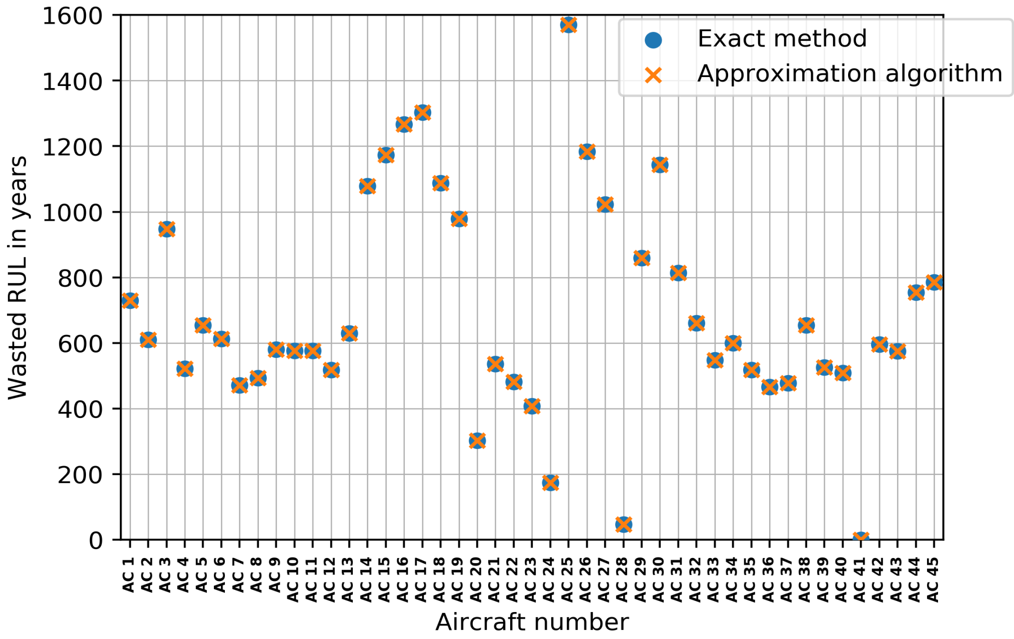

Figure 8.8: Comparison RUL wasted per aircraft obtained from exact method and approximation algorithm solution.

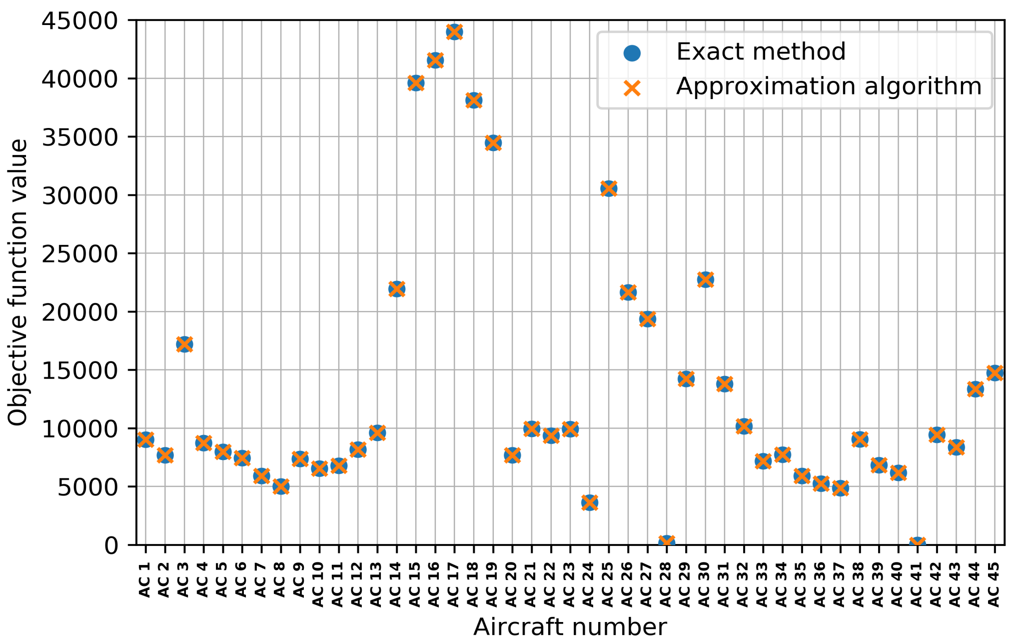

Figure 8.9: Comparison of objective function values obtained from exact method and approximation algorithm solution.

8.4.3. Approximation algorithm: Operational stage level 2

The first thing to check at the operational stage level 2 model is that the correct access panels will be opened. The results of executing the approximation operational level 2 model, on data of aircraft 16, indicate that for maintenance opportunity A4.14 13 groups of access panels need to be opened. The different groups of access panels that need to be opened, according to the model, are given in table 8.3. These are exactly the same as the ones presented in the solution of the exact method. Another important thing to confirm is that the amount of tasks as well as the unique tasks reference numbers are equal to the ones presented in the solution of the exact method. The results of the approximation model indicate that there are 164 tasks allocated, with the same unique reference tasks numbers as the ones presented for the exact method solution. The last main thing to check is if the tasks that represent a series need to be executed in the right sequence. These tasks all received a linked constraint in the exact model. The approximation algorithm will allocate tasks with a linked constraint at the same or a later shift. Results indicate that each of the linked tasks were allocated in the right sequence to shift 1 of maintenance opportunity A4.14. This concludes that the model at the operational level 2 was successfully validated.

# Task	Panels #
138	825
139	841
140	831
141	316AR, 315AL
142	714, 713
143	832
144	842
145	826
146	437AL, 438AR, 452AR, 451AL
147	448AR, 447AL, 461AL, 462AR
148	827
149	152KW, 151KW
150	744, 734, 196BB, 195BB, 197CB, 197FB, 147EB

Table 8.3: Newly inserted opening access panels tasks.

8.5. Incorporation models into AIRMES

The AIRMES project is a European Union funded research project that focuses on optimizing end-to-end maintenance activities within an operator's environment. The proposed approximation algorithms are successfully integrated with the AIRMES tool as can be seen in Appendix E and F. On Wednesday 27th of March 2019 a successful live demo was given to stakeholders of the AIRMES project. In the demo the results of the models were shown using data of 55 different aircraft, a 4 year planning horizon and an updated task excel file that was aligned with the current operation of the fleet.

9

Sensitivity analysis

A sensitivity analysis is conducted to seek how some of the key parameters and assumptions associated with the case study affect the results of the proposed models. Each of the exact models were coded in Python 3.7 and solved using the free academic license of Gurobi. The approximation algorithms were coded and solved with Python 3.7. The computational times are obtained by solving the models on an Intel Core i5 1.6 GHz laptop with 4GB ram.

9.1. Tactical stage models: Reduced man-hours factor

In the results chapter it was already noticed that each of the maintenance opportunities had a surplus in man-hours. In order to assess the behaviour of the model, under more restricted cases, the available man-hours was slowly reduced. In this section results will be presented from simulating the tactical stage model with reduced man-hours, by slowly reducing a factor, that is multiplied with the available man-hours at both the A and C check maintenance opportunities, from 0.6 to 0.4.

9.1.1. Results

Figure 9.1 presents the results of simulating different man-hours factors, for the exact method and approximation algorithm, between 0.6 and 0.4. In this figure each of the round points indicate a decrease of the man-hours factor with 0.1. Based on the results it may be concluded that the objective function value starts to decrease below a man-hours factor of 0.58. This means that below this factor parts of the time segments reach their full man-hours capacity. The consequence of this is that some of the tasks will be allocated to earlier maintenance opportunities than originally intended at higher man-hours factors. If one compares the solution quality of the approximation algorithm to the reference solution (i.e. solution of the exact method), there is a small difference obtained which slowly increases to 4.9 per cent at a man-hours factor of 0.4. It is relevant to mention that the approximation algorithm used a few A-check fictitious man-hours for the case of a man-hours factor of 0.41 and 0.40. The computational time required to solve the approximation algorithm behaves linear and takes only 948 seconds for a man-hours factor of 0.4. This is already lower than the time required to solve the man-hours factor of 0.6 with the exact method, which takes 1,213 seconds. The computational time required to solve the exact method shows a strong increasing trend that starts at 1213 seconds for a man-hours factor of 0.6 and ends up with an astonishing 24,614 seconds for a man-hours factor of 0.4. This means that the exact method takes almost 26 times longer to find a solution compared to the approximation algorithm for a man-hours factor of 0.4. The obtained simulation results visualized in figure 9.1 are presented in Appendix G.

Figure 9.2 presents, for three different man-hours factors, the RUL wasted per aircraft as obtained from the approximation algorithm solution. The obtained results of the man-hours factor of 0.35 required 96 fictitious C-check man-hours and 6 fictitious A-check man-hours. The figure shows that decreasing the available man-hours has as affect that more RUL is wasted for most aircraft. This means that decreasing the available man-hours has as direct affect that more tasks will need to be executed on an earlier maintenance opportunity than originally intended at a man-hours factor above 0.58. In order

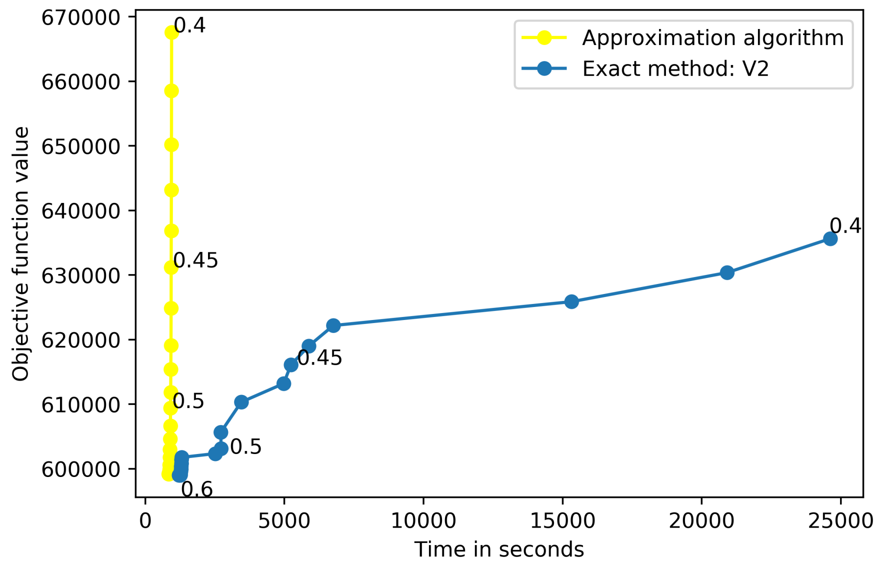

Figure 9.1: Comparison models tactical level with decreased man-hours factor from 0.6 to 0.4.

to visualize the man-hours constraints a snapshot of overlap among aircraft is analysed. Figure 9.3 illustrate a snapshot of the maintenance schedule where five different aircraft are linked in seven time segments. In this figure there are eight different bars per time segment given, these bars represents the different skill types in the following sequence: GR1, GR2, GR4, ESHS, PINT, MAP, and NDT. The available man-hours (i.e. red transparent bars) and the used man-hours (i.e. legend colours for each aircraft), for a factor of 0.42, per time segments are presented in figure 9.4. This figure confirms that skill type GR2 is the most restricted skill type. Within time segments 1,2,3,4,5 and 6 the maximum man-hours capacity of skill type GR2 was used. This means that there will be tasks, requiring skill type GR2, that are allocated at an earlier maintenance opportunity compared to the case of a man-hours factor above 0.58. The exact method solution show similar results for this same situation as can be seen in appendix D.

9.1.2. Discussion of results

The results indicate that decreasing the amount of man-hours available affect the wasted RUL for most of the aircraft. In figure 9.4 it can be seen that AC-5, AC-16, and AC-17 used all their man-hours of skill type GR2 during their time segments. The affect on the RUL becomes visible in figure 9.2 wherein AC-5, AC-16, and AC-17 show a significant increase in RUL wasted. Overall, the results indicate that both models are capable of providing solutions even for cases where tasks are allocated to earlier maintenance opportunities than originally intended at non-restricted situations.

9.2. Tactical stage models: Increased man-hours factor for C-check tasks

From the solutions, obtained in the case study, it was observed that C-check maintenance opportunities are more restricted than A-check maintenance opportunities. In order to further assess the capabilities of the tactical level models it was decided to increase the man-hours required for C-check tasks. For this experiment the non-routine rates, as obtained from the airline, are excluded and replaced with a factor that is multiplied with each of the C-check tasks. By slowly increasing this so called 'cfactor' it is possible to assess the capabilities and quality of the tactical level models.

9.2.1. Results

Figure 9.5 presents the results of simulating cfactors, from 2.0 to 4.0, for the proposed tactical level models in chapter 5. If one compares the solution quality of the approximation algorithm to the ref-

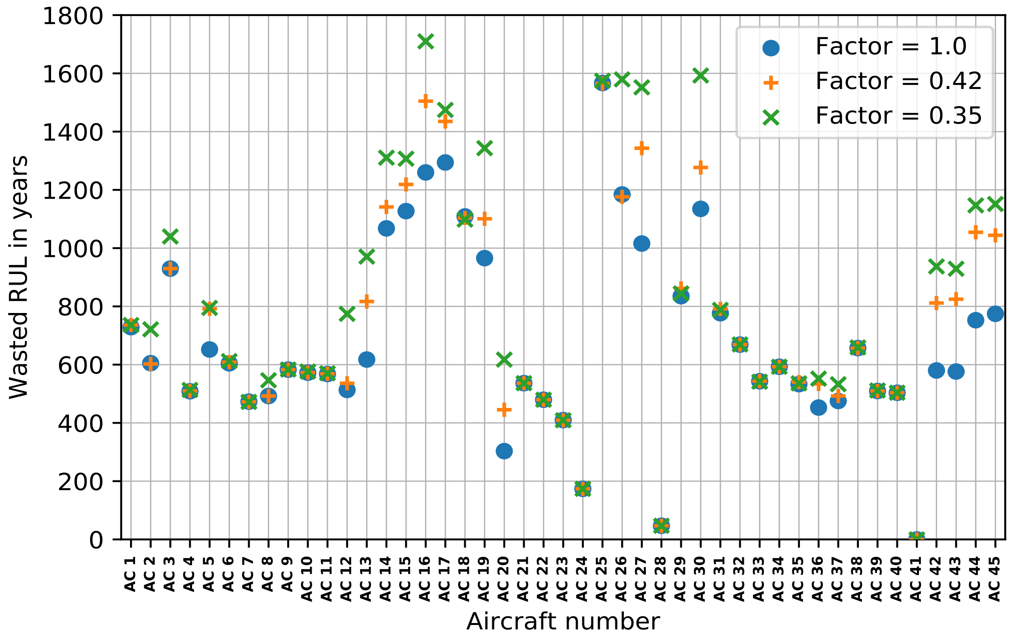

Figure 9.2: Approximation algorithm: Wasted RUL per aircraft tail number under various conditions.

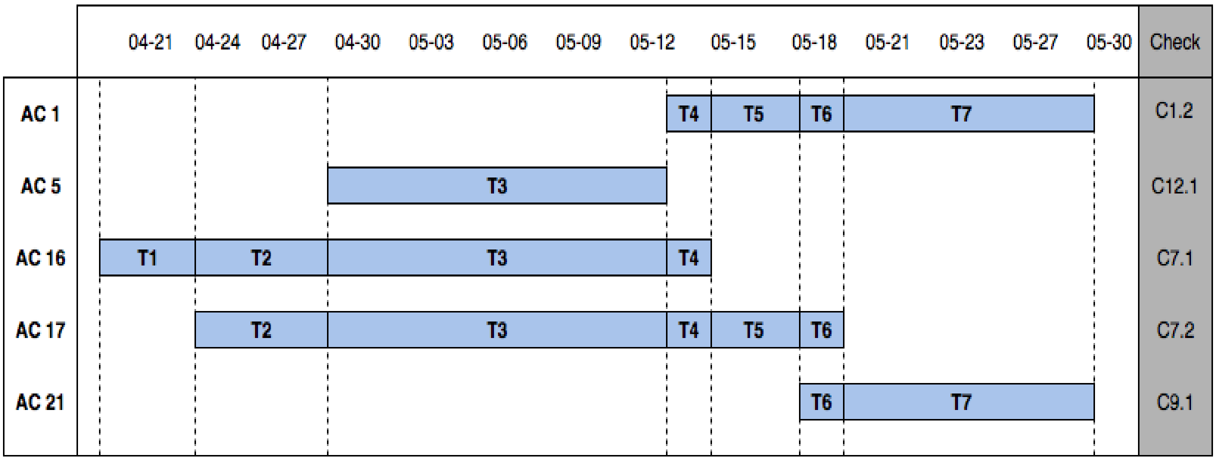

Figure 9.3: Snapshot of maintenance overlap between aircraft.

erence solution (i.e. solution Exact method variant 2), there is a small error obtained which slowly increases by increasing cfactor up to a 1% difference for the case of a cfactor of 3.5. The fictitious maintenance opportunities of the second variant of the exact method results into slightly better solutions than the first variant of the exact method for restricted cases with a cfactor of 2.5 and above. It was observed that the computational time of the approximation algorithm behaves linear with increasing cfactor. On the other hand the exact methods show a strong increasing trend especially at higher cfactors. If one compares the computational time of the approximation algorithm to the reference solution, exact method variant 2, the computational time of the exact method is already more than twice as high for a cfactor of 2.0 and increases up to six times as high for a cfactor of 3.5. It is relevant to mention that the second variant of the exact method has a lower computational time for restricted cases than the first variant of the exact method. Each of the models have been tested up to a cfactor of 4.0, only the approximation algorithm was not able to provide a solution, without using fictitious man-hours, for cases with a cfactor higher than 3.5. The obtained simulation results visualized in figure 9.5 are presented in Appendix H.

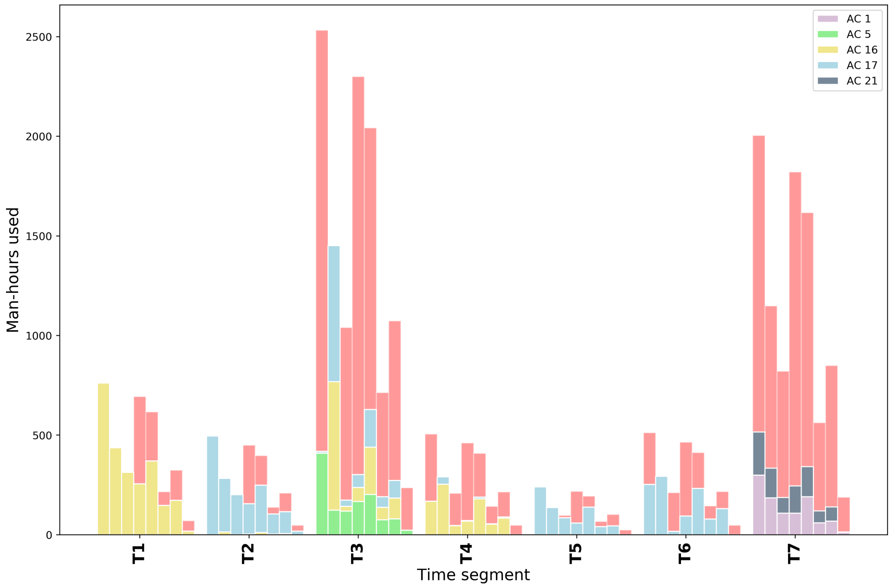

Figure 9.4: Approximation algorithm: Man-hours per skill type used per time segments for the man-hours factor equal to 0.42.

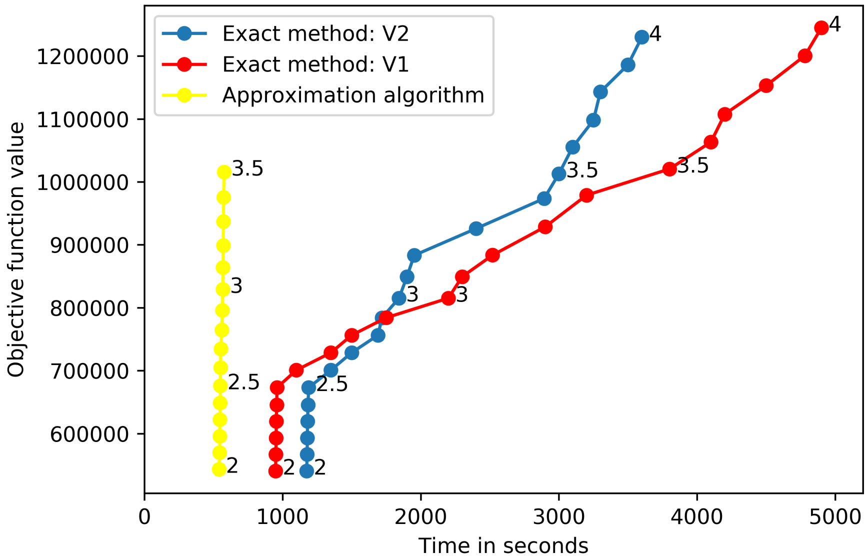

Figure 9.5: Comparison models tactical level with increased cfactor from 2.0 to 4.0.

9.2.2. Discussion of results

It was observed that the approximation algorithm was not able to provide a solution, without using fictitious man-hours, in the case of a cfactor above 3.5. In reality airlines will try to execute their tasks at their last possible maintenance opportunity, which corresponds to the situation of a cfactor between 2.0 and 2.5. Results of cfactors above 2.5 causes the execution of tasks before their optimal maintenance opportunity which will be avoided by most airlines. Therefore, it may be concluded that the approximation algorithm is capable enough of handling real life situations. Results suggest that the solution quality of the exact method variant 2 are slightly better for cfactor values above 2.5. This was expected since the fictitious maintenance opportunity give the second variant model more freedom to choose which tasks to add in the solve loop compared to the first variant of the exact method. Furthermore, the computational time of the second variant model is lower than the first variant model for restricted cases. This is caused by the fact that the fictitious maintenance opportunities will increase the change of solving the problem without using additional iterations.

9.3. Tactical stage models: Allow small C-check tasks into A-check maintenance opportunities

Results clearly indicated that A-check maintenance opportunities have much more man-hours remaining than C-check maintenance opportunities. Therefore, it might be interesting to allocate some of the smaller C-check tasks into A-check opportunities. It is expected that this will lead to a decrease in the objective function caused by a decrease of the average RUL value of C-check tasks. It is worth mentioning that small C-check tasks, that require skill types ESHS or PINT, will not be allowed on A-check maintenance opportunities. In this section results of simulating the models with allowing small C-check tasks at A-check maintenance opportunities will be presented. Various scenarios were tested whereby the limit of man-hours C-check tasks, to be allowed on A-check maintenance opportunities, was changed from 0 to 2.5.

9.3.1. Results

It was mentioned in subsection 7.1.3 that for the case study data the average RUL wasted per C-check task was 205 days and per A-check task 19.3 days. Figure 9.6 presents the average RUL wasted per C-check tasks and per A-check tasks for the various maximum C-check tasks man-hours limits that were allowed on A-check maintenance opportunities. When analyzing the results it is clearly visible that the average RUL wasted per C-check task shows a decreasing trend. At a C-check tasks man-hours limit of 2.5 the average RUL wasted, in days, of a C-check tasks is 132. This is an improvement of 42 per cent compared to the situation where no C-check tasks are allowed during the A-check maintenance opportunities. Furthermore, figure 9.7 presents the average RUL wasted in years, per aircraft tail number, for the different limit C-check man-hours. For each of the aircraft tail numbers, except two of the phased-out aircraft, the average wasted RUL in years will decrease when the C-check man-hours limit is increased. The solutions from the approximation algorithm show similar results.

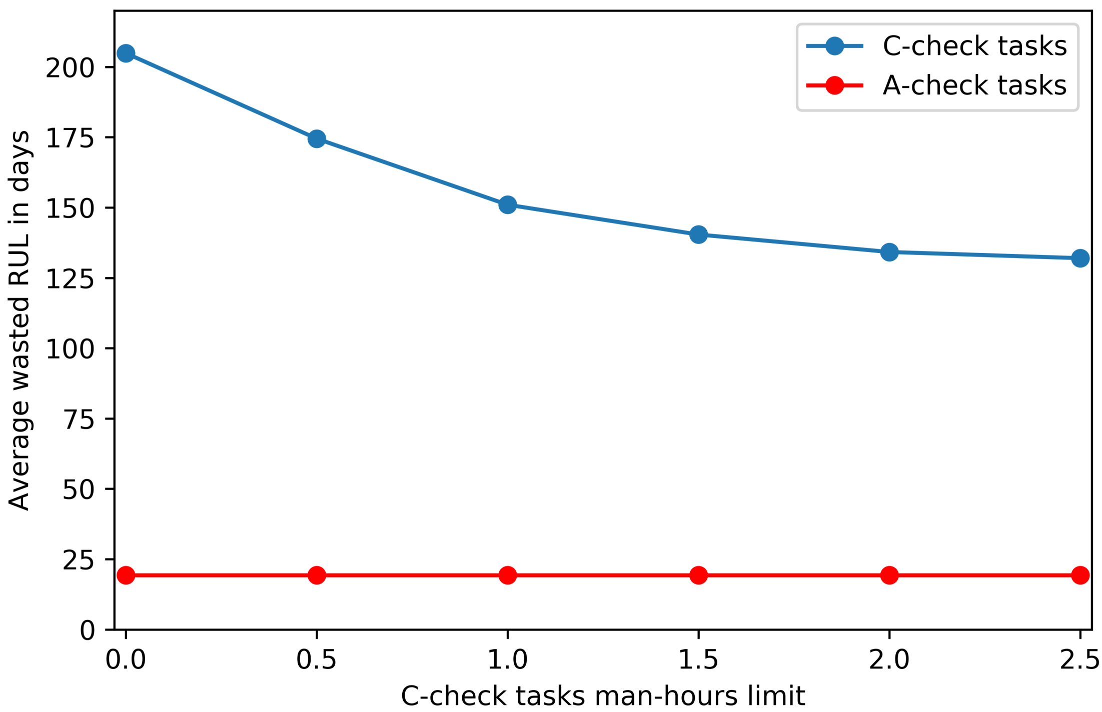

Figure 9.6: Exact method: Average wasted RUL in days for increased C-check task man-hours limits.

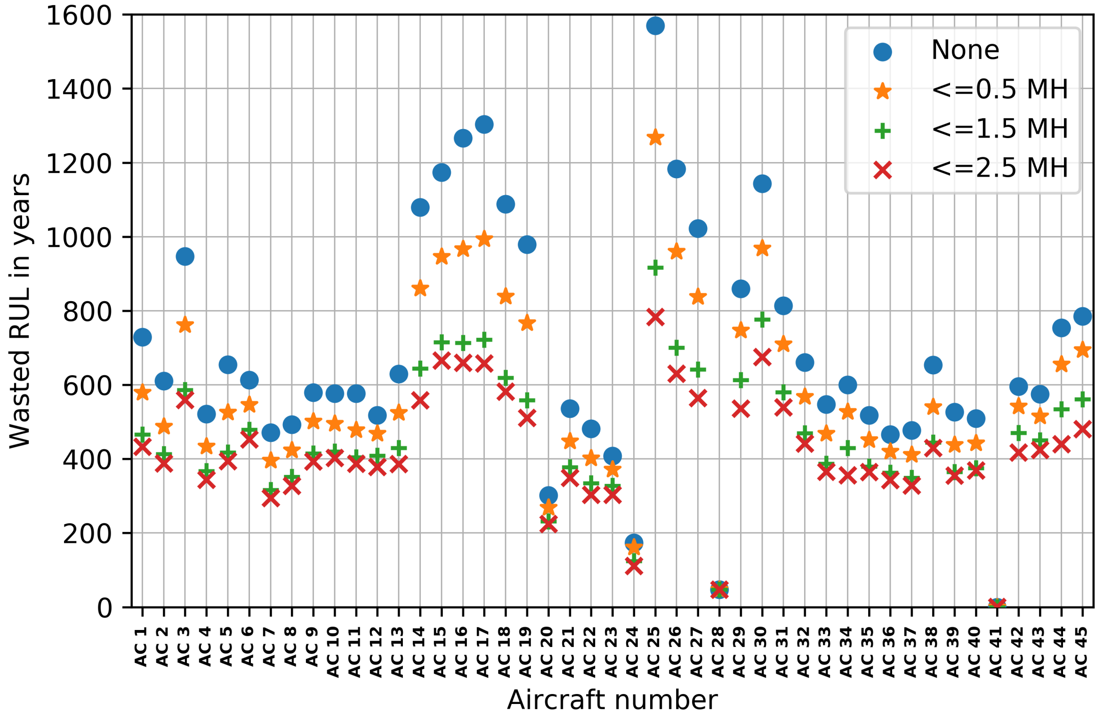

Figure 9.7: Exact method: Average wasted RUL in years per aircraft tail number for various C-check task man-hours limits.

9.3.2. Discussion of results

The results indicate that C-check tasks will have a lower average wasted RUL when some of the C-check tasks are allowed on A-check maintenance opportunities. This was expected given that the maintenance schedule is optimized for C-check tasks that have interval limits around the minimal C-tasks intervals 7,500 flight-hours, 5,000 flight-cycles or 24 months and their consecutive intervals.

9.4. Approximation tactical stage model: Maintenance schedule

During the case study results were obtained by executing the tactical stage model on a maintenance schedule with a planning horizon of 4 years. The model was also tested for different maintenance schedules with a 3, 4, 5 and 6 years planning horizon. Appendix I presents the computational times required to solve the different planning horizons with the approximation algorithm at the tactical stage. The results indicate that the approximation algorithm has a linear time behaviour when the planning horizon is extended.

9.5. Operational stage level 1 model: Reduced man-hours factor

Since the tactical stage, of the model framework, will transfer the interdependent problem to an independent problem it is possible to solve the task allocation problem, at the operational stage level 1, for each aircraft separately, thereby reducing the complexity of the problem considerably. The output of the tactical stage will be used as main new input at the operational stage level 1 model. In this section the effect of reducing the man-hours with a factor of 0.6 and 0.4 will be presented for the operational stage level 1 model.

9.5.1. Results

The operational stage level 1 model was tested with the excel file obtained from simulating the tactical stage model with a man-hours factor of 0.6 and 0.4. The average computational time of the approximation operational stage level 1 model was around 30 seconds, while the exact method took around 90 seconds. Given that there is no additional input inserted in the operational level 1 model (e.g. additional tasks, utilization correction) the solution quality difference, on average per aircraft, between the exact method and approximation algorithm remains the same as the tactical stage model. A snapshot of the maintenance schedule (figure 9.3) was investigated in section 9.1. In which it was indicated that reducing the man-hours factor to 0.4 results in a restricted situation for AC-5, AC-16 and AC-17 (as can be seen in figure 9.4). Therefore, the approximation operational level 1 model was solved for AC-5, AC-16 and AC-17 with a man-hours factor of 0.6 and 0.4. Results of the man-hours distribution for the C-checks are presented in figure 9.8 till 9.13. Each of the figures indicate that no restriction in man-hours occur when a man-hours factor of 0.6 was used. However, the figures show that a man-hours factor of 0.4 results in a restricted situation. Whereby, tasks are allocated to an earlier C-check maintenance opportunity than originally planned. Table 9.1 presents simulations results of using the man-hours factor of 0.6 and 0.4 for the approximation algorithm. The following is observed:

- AC-5: At the restricted maintenance opportunity C12.1 a total of 188.78 required man-hours of skill type GR-2 (which corresponds to 58 per cent of the total used man-hours at the man-hours factor of 0.6) could not be used, when considering a man-hours factor of 0.4, and was therefore shifted to an earlier maintenance opportunity.
- AC-16: At the restricted maintenance opportunity C7.1 + A1.17 a total of 466.11 required man-hours of skill type GR-2 (which corresponds to 26.5 per cent of the total used man-hours at the man-hours factor of 0.6) could not be used, when considering a man-hours factor of 0.4, and was therefore shifted to an earlier maintenance opportunity.
- AC-17: At the restricted maintenance opportunity C7.1 + A3.16 a total of 521.93 required man-hours of skill type GR-2 (which corresponds to 28 per cent of the total used man-hours at the man-hours factor of 0.6) could not be used, when considering a man-hours factor of 0.4, and was therefore shifted to an earlier maintenance opportunity.

Furthermore, the simulations results from executing the exact method with a man-hours factor of 0.6 and 0.4 are presented in table 9.2. From the comparison of these tables it may be concluded that the amount of skill type GR-2 man-hours used at the considered C-checks, for each aircraft, is almost equivalent for both models. The only difference can be found in the total amount of tasks allocated to the specific C-check maintenance opportunities. As expected the exact method, for restricted cases, seems to be more capable of determining which tasks to allocate to the restricted maintenance opportunities.

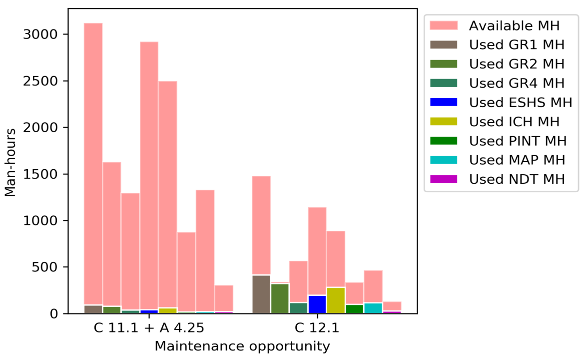

Figure 9.8: AC-5 results with a man-hours factor of 0.6

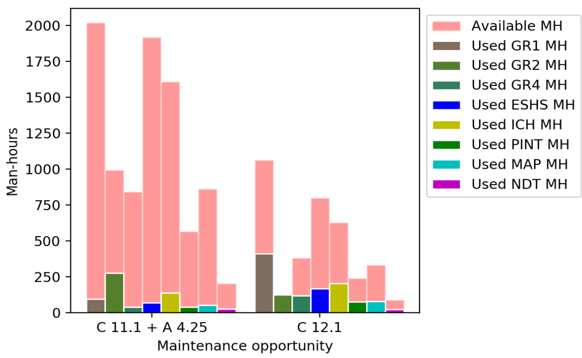

Figure 9.9: AC-5 results with a man-hours factor of 0.4

9.5.2. Discussion of results

The presented results confirm that both the exact and approximation algorithm are capable of allocating a large portion of the required man-hours to an earlier maintenance opportunities in restricted cases (i.e. up to 58 per cent for AC-5). The presented results for a man-hours factor of 0.4 show that the exact method of the operational stage level 1 model performs better than the approximation algorithm. However, it must be noted that the 0.4 man-hours factor leads to an unrealistic situation where a significant amount of man-hours required at unrestricted situations (i.e. AC-5 58 per cent, AC-16 25

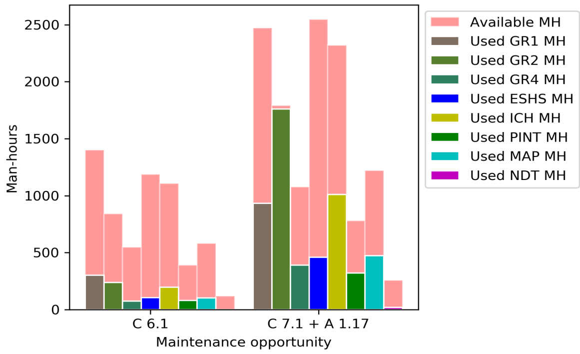

Figure 9.10: AC-16 results with a man-hours factor of 0.6

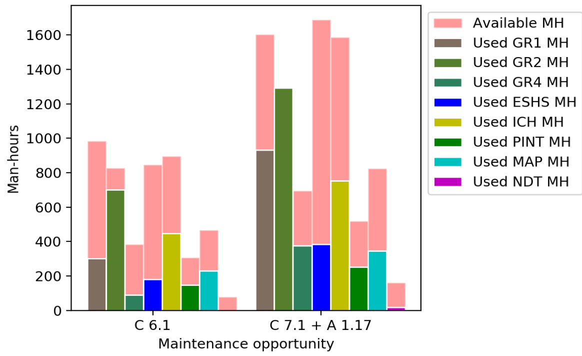

Figure 9.11: AC-16 results with a man-hours factor of 0.4

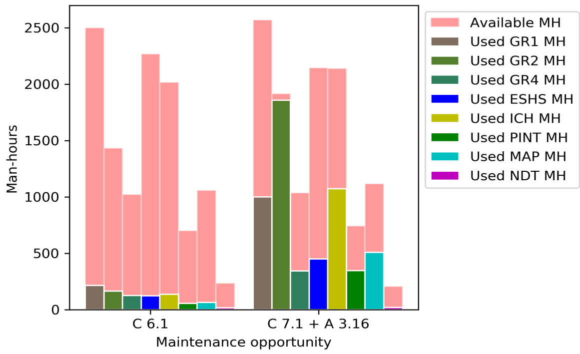

Figure 9.12: AC-17 Results with a man-hours factor of 0.6

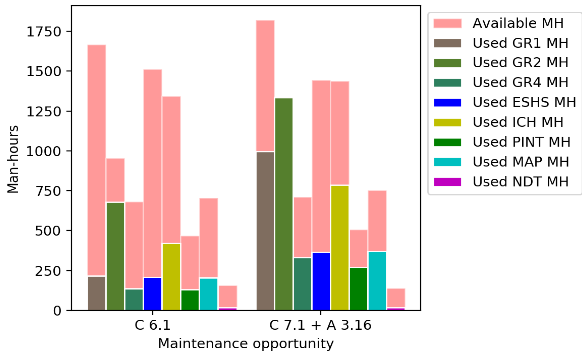

Figure 9.13: AC-17 results with a man-hours factor of 0.4

AC	MO	MH factor	GR2 available	Approximation algorithm: GR2 used	# tasks allocated
AC-5	C11.1 + A4.25	0.6	1926.8	77.5	118
AC-5	C11.1 + A4.25	0.4	994.3	274.9	162
AC-5	C12.1	0.6	339.4	321.6	308
AC-5	C12.1	0.4	123.84	123.82	263
AC-16	C6.1	0.6	841	237.79	199
AC-16	C6.1	0.4	828	701.6	299
AC-16	C7.1 + A1.17	0.6	1790	1759.8	622
AC-16	C7.1 + A1.17	0.4	1294.4	1293.69	520
AC-17	C6.1	0.6	1432.35	164.4	166
AC-17	C6.1	0.4	954.5	678.7	275
AC-17	C7.1 + A3.16	0.6	1916.4	1857.6	654
AC-17	C7.1 + A3.16	0.4	1336.8	1335.67	540

Table 9.1: Approximation algorithm: Solutions obtained from using a man-hours factor of 0.6 and 0.4

per cent and AC-17 28 per cent), were allocated to earlier maintenance opportunities. This is especially undesired during C-check maintenance opportunities since the gap between two C-checks is around 2 years.

9.6. Operational stage level 2 model: Reduced man-hours factor

As was explained earlier, in the discussion section of section 7.3, the operational stage level 2 models introduce a small amount of additional tasks to open and close specific access panels. This leads to problems if the full man-hours capacity of a specific skill type is used during a maintenance opportunity. Therefore, it was recommended to already reserve some of the man-hours for opening and closing of

AC	MO	MH factor	GR2 available	Exact method: GR2 used	# tasks allocated
AC-5	C11.1 + A4.25	0.6	1926.8	77.5	118
AC-5	C11.1 + A4.25	0.4	994.3	280	159
AC-5	C12.1	0.6	339.4	321.6	308
AC-5	C12.1	0.4	123.84	123.84	266
AC-16	C6.1	0.6	841	237.79	199
AC-16	C6.1	0.4	828	700	235
AC-16	C7.1 + A1.17	0.6	1790	1759.8	622
AC-16	C7.1 + A1.17	0.4	1294.4	1294	581
AC-17	C6.1	0.6	1432.35	164.4	166
AC-17	C6.1	0.4	954.5	678.4	195
AC-17	C7.1 + A3.16	0.6	1916.4	1857.6	654
AC-17	C7.1 + A3.16	0.4	1336.8	1336.8	614

Table 9.2: Exact method: Solutions obtained from using a man-hours factor of 0.6 and 0.4

access panels at the tactical stage model. Results presented in this section did not reserve any man-hours during the tactical stage model. Therefore, only the results of the approximation algorithm at the operational stage level 2 are presented for various man-hours factors.

9.6.1. Results

The approximation algorithm at the operational stage level 2 was executed for the restricted C-check maintenance opportunity of AC-5 (figure 9.14 and 9.15), AC-16 (figure 9.16 and 9.17) and AC-17 (figure 9.18 and 9.19) with a man-hours factor of 0.8 and 0.6, respectively. Each of the figures clearly indicate that the downtime of the aircraft is shorter when a man-hours factor of 0.8 is used. This makes absolutely sense since more man-hours are available at higher man-hour factors. The figures also confirm that the man-hours factor of 0.6 corresponds to the situation where each of the shifts use all their GR2 skills for AC-5 and AC-16. The maintenance downtime of each aircraft would be much lower if there would be more man-hours of skill type GR2. For example when considering figure 9.16 the shifts 9 till 59 are used to allocate tasks that only require skill type GR2 and any tasks that may not be executed earlier due to the linked constraints with the allocated GR2 tasks. This means that during these shifts there is plenty of man-hours remaining of the other skill types.

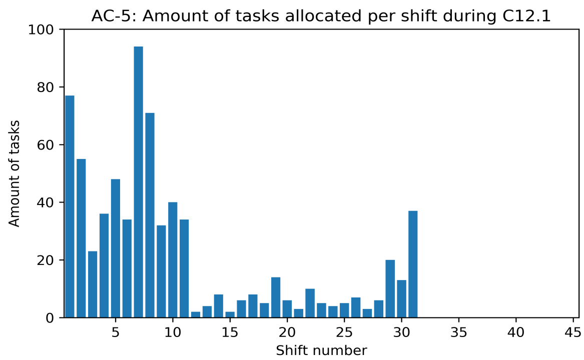

Figure 9.14: AC-5 results with a man-hours factor of 0.8

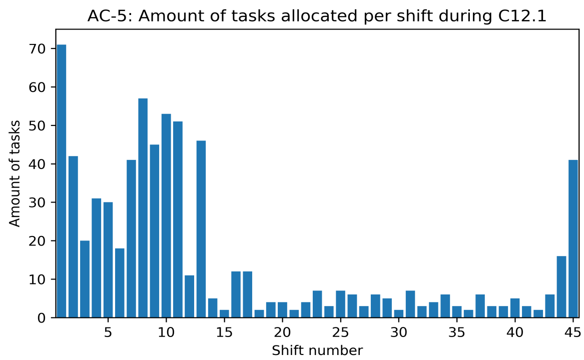

Figure 9.15: AC-5 results with a man-hours factor of 0.6

9.6.2. Discussion of results

The results indicate that the approximation algorithm is capable of allocating the tasks to the available shifts within seconds. It was shown in the results that a lower man-hours factor results in a reduced aircraft downtime. Furthermore, the results confirm that the GR2 skill type is the most constricted skill type. Therefore, the recommendation would be to certify some of the engineers, currently certified with skill type GR4, ESHS, PINT or MAP, with skill type GR2. This allows for a better match between

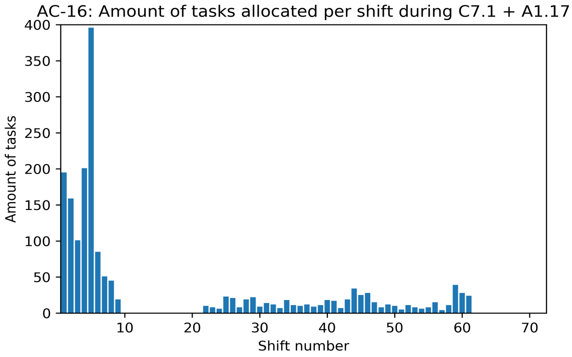

Figure 9.16: AC-16 results with a man-hours factor of 0.8

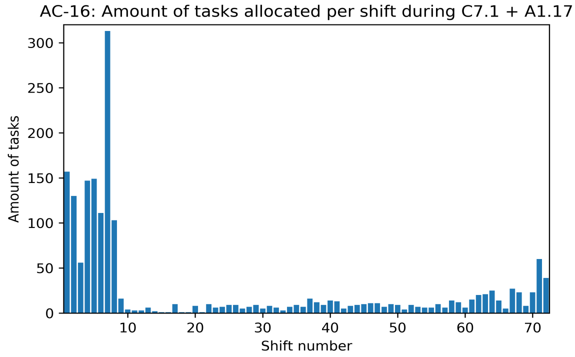

Figure 9.17: AC-16 results with a man-hours factor of 0.6

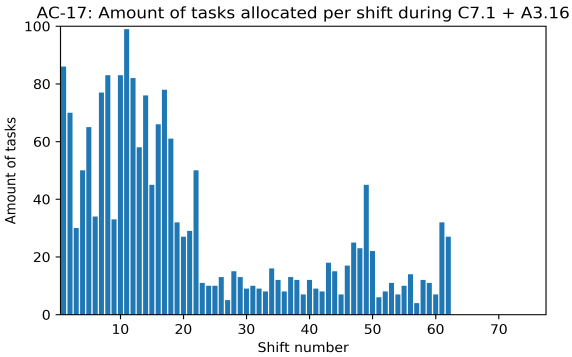

Figure 9.18: AC-17 results with a man-hours factor of 0.8

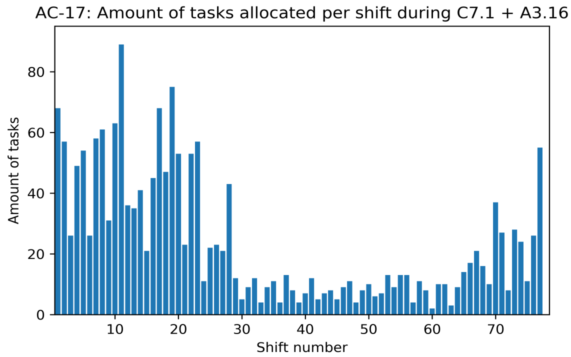

Figure 9.19: AC-17 results with a man-hours factor of 0.6

skill types demand and available man-hours per skill type. This better match between demand and available man-hours per skill type will result in a lower overall aircraft maintenance downtime.

9.7. Approximation algorithm operational stage level 2: Additional constraint

The third constraint type (equation 6.12) of the shift level model ensures that any preceding relation between tasks is not violated. Among these tasks are large tasks that are broken down into smaller tasks, requiring a maximum of four man-hours, and tasks that need to be executed in series. The third constraint type made sure that these tasks would be executed within the same or a later shift depending on the man-hours available during that shift. The results of the model indicated that most of those tasks would be allocated to the same shift. The intention was that those tasks should be spread over the different shifts so that just one or two technicians could work on part of those tasks per shift. In order to accomplish this the following change was made for C-check maintenance opportunities:

- Any task that is broken down to smaller tasks or tasks that need to be executed in series may only be allocated to the same shift if the sum of the allocated preceding tasks of the task did not exceed eight man-hours at the considered shift. Otherwise the task is allocated to the next shift.

The change was implemented into the approximation algorithm of the operational stage level 2. The improved algorithm was tested on the restricted maintenance opportunity of AC-5, AC-16 and AC-17 with a man-hours factor of 0.8. Results indicate that more of the reserved shifts during the considered checks are used (figures 9.20, 9.22, 9.24) compared to the original results (figures 9.14, 9.16, 9.18). Furthermore, the man-hours distribution during the maintenance overlap between AC-5, AC-16 and AC-17 was obtained for the restricted skill type GR2. The considered planning horizon starts from 2020-04-21 and stops at 2020-05-19. These days cover the overlap situation, as was presented in figure 9.3, of the three considered aircraft. The considered C-check, C12.1, of AC-5 starts 7 days after 2020-04-21 and ends at 2020-05-12. Figure 9.21 presents the used man-hours of skill type GR-2 for AC-5 during the considered planning horizon. The results indicate that the C-check is done 4 days earlier than planned. The considered C-check, C7.1 + A1.17, of AC-16 starts at the start of the planning horizon and ends at 2020-05-14. Figure 9.23 presents the used man-hours of skill type GR-2 for AC-17 during the considered planning horizon. The results indicate that the C-check can be finished exactly in time. The considered C-check, C7.1 + A3.16, of AC-17 starts 3 days after 2020-04-21 and ends at 2020-05-19. Figure 9.25 presents the used man-hours of skill type GR-2 for AC-17 during the considered planning horizon. The results indicate that the C-check can also be finished exactly in time. The total man-hours distribution during the planning horizon for skill type GR1 and skill type GR2 are presented in 9.26 and 9.27, respectively. From these figures it may be concluded that the skill type GR2 is the most restricted skill type.

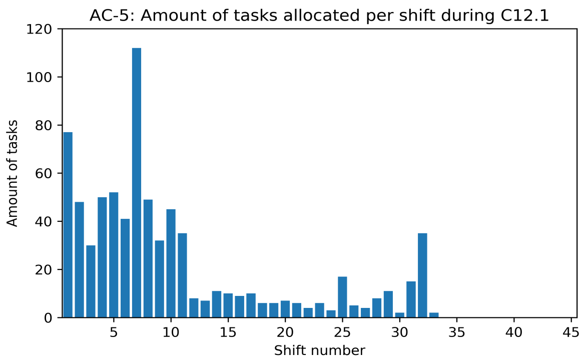

Figure 9.20: New AC-5 results with man-hours factor 0.8

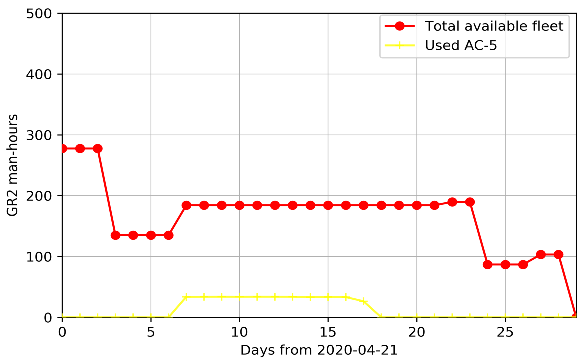

Figure 9.21: Man-hours distribution per day for AC-5 and skill type GR2.

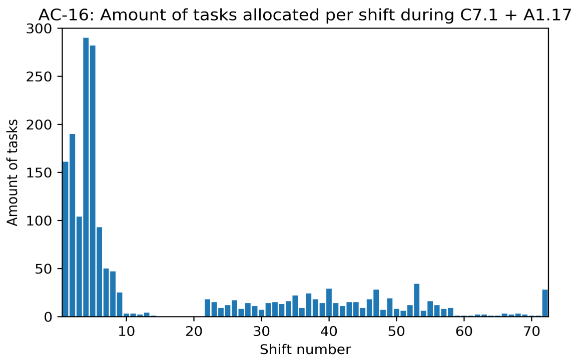

Figure 9.22: New AC-16 results with man-hours factor 0.8

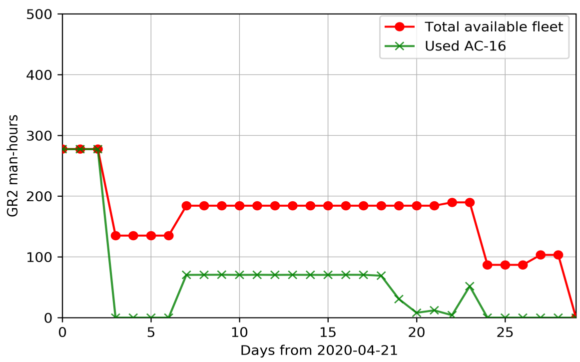

Figure 9.23: Man-hours distribution per day for AC-16 and skill type GR2.

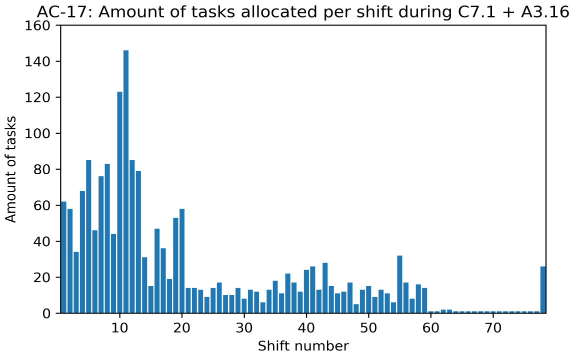

Figure 9.24: New AC-17 results with man-hours factor 0.8

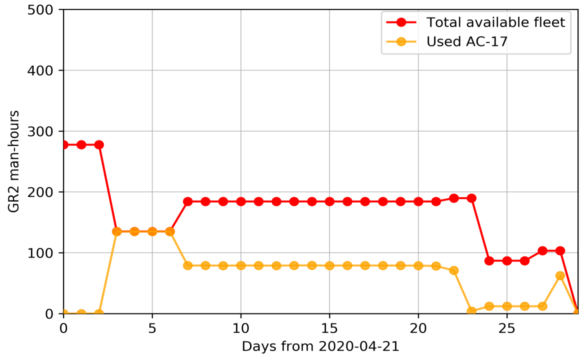

Figure 9.25: Man-hours distribution per day for AC-17 and skill type GR2.

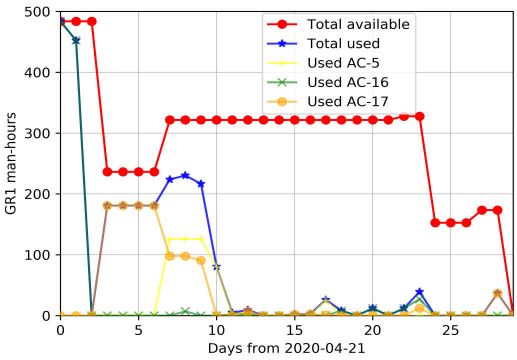

Figure 9.26: Combined man-hours distribution skill type GR1.

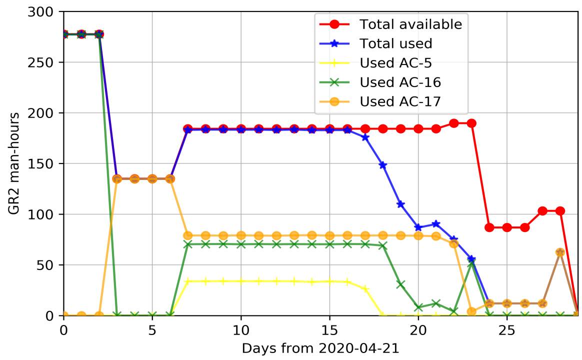

Figure 9.27: Combined man-hours distribution skill type GR2.

III

Conclusions and recommendations

10

Discussion and conclusions

In the research framework the following research objective was stated: *To develop a maintenance tasks packaging model for all the tasks required to ensure ongoing airworthiness of the aircraft by developing a framework capable of delivering optimized schedules for the allocation of maintenance tasks to the available maintenance opportunities.*
In order to achieve the research objective, two main research questions were developed. The main research questions were further split into sub questions, which together provide the answer for each corresponding main research question.

1. How can the modelling framework be designed?
   (a) What are the main functions required for the model?
   (b) What solution technique(s) will be used to solve the model?
   (c) What are the results of simulating the model with data from an airline?
2. Is the final framework capable of delivering better quality schedules compared to the current approach?
   (a) Do the results of simulating the framework deliver the required features?
   (b) Is the final framework acceptable in terms of computational time, optimization and assumptions?

1a: What are the main functions required for the model?
By studying the current approaches of solving the task allocation problem it was possible to understand the requirements necessary for the model. Literature indicated that long-term maintenance plans were frequently affected by deviations from flight plans or unforeseen events. This causes the need for frequently modifying or redoing the maintenance schedule [26]. Therefore, the first main requirement of the model was that it should be capable to reoptimize at the operational level. Furthermore, literature indicated that the due-date of each task should be an important criterion in deciding which tasks to allocate to which maintenance opportunity. This is a very important requirement because an aircraft is essentially not airworthy when any of the tasks reaches their due-date.

In addition, the airline supporting this research also included a few requirements. To ensure that the model does not impose financial burden for the airline it is required to create models that run on free software. In order to make the models useful for the airline they required a maximum computational time of 12 hours for the tactical stage model and 10 minutes for the operational stage level models. In addition, the airline requested that the solution quality of the approximation algorithm should be known. Therefore, the last main requirement was that the approximation algorithm should be validated by comparing its solution with an optimal solution obtained from an exact method.

1b: What solution technique(s) will be used to solve the model?

The literature study, conducted prior to this research, was concluded with a proposed approach to solve the task allocation problem. It was recommended to first formulate the problem as a MILP formulation. After which a commercial solver (e.g. CPLEX or Gurobi) should be used to solve the problem. The solution from this exact method could then be used as reference solution for the approximation algorithm. Based on the literature findings it was recommended to formulate the task allocation problem as a bin packaging problem, after which a combination of well-known bin packaging solution techniques could be used to solve the problem.

The proposed approach was successfully followed throughout this research. Each of the exact models were presented as a MILP formulation, and were solved with the commercial solver Gurobi. Obtained results were useful for understanding the main characteristics of the task allocation problem and to benchmark the solution quality of the approximation algorithms. This knowledge was used to formulate the task allocation problem as a bin packaging approach. Three approximation algorithms that use a combination of well-known bin packaging solution techniques were proposed to solve the task allocation problem down to shift level.

1c: What are the results of simulating the model with data from an airline?
From the simulation of the tactical stage models, with the airlines data, we retrieve excellent solution quality for both the exact and approximation model. Both models' output provide similar results, which is expected since the airline had a surplus in man-hours for each of the maintenance days. This research was initiated because the airline was not able to make a maintenance program for a considered time horizon beyond 1 year. The tactical stage models were successfully tested for a time horizon of 4 years. This is a major improvement for the airline and allows them to better estimate the man-hours required for the upcoming years.

Since the results of the tactical stage model will transfer the interdependent problem to an independent problem it is possible to solve the task allocation problem, at the operational stage, for each aircraft separately, thereby reducing the complexity of the problem considerably. The simulation of the operational stage level 1 models, with the airlines data, resulted in similar outcomes for both the approximation and exact model. The model will allow the airline to add additional maintenance tasks (e.g. non-routine maintenance, modifications) without influencing the other aircraft in the fleet. Furthermore, it is possible to input the current utilization of the aircraft thereby correcting the small deviation obtained by assuming an average flight-hours or flight-cycles utilization. This will lead to an improved RUL of the tasks and ensures that none of the tasks exceed their due-date. The simulation of the operational stage level 2 models resulted in solving the task allocation problem down to shift level. The results of this last operational level is an excel file with the distribution of tasks over each of the working shifts for the upcoming maintenance opportunity. This file can directly be used by the maintenance manager to distribute the tasks, for each shift, among the available workforce per shift.

Overall the obtained results confirm that the proposed models are very well capable to solve the task allocation problem with the airlines data, being solved down to shift level. The benefits of the airlines are numerous. Firstly, the time consuming job of creating a maintenance program, for an entire fleet, is fully automated by the proposed model framework. Secondly, the models will allow the airline to adequately react to cases were aircraft deviate from their flight plan or in the case of unforeseen events (e.g. non-routine tasks, required modifications, additional maintenance opportunities). Lastly, the tactical stage model allows the airline to test new maintenance concepts. This was tested in the sensitivity analysis chapter where results indicated the effect of allocating small C-check tasks to A-check maintenance opportunities.

2a: Do the results of simulating the framework deliver the required features?
The first main requirement of the framework was that it should be able to reoptimize at an operational level. The operational stage level 1 model enables the user to reoptimize the task allocation problem for a considered aircraft within a minute. The second requirement was that the due-date should be an important criterion to determine the best maintenance opportunity location for each task. The due-date is used as an important criterion for calculation of the task cost used in the objective function. In addition, the operational level 1 model enables the user to correct the flight-hour and flight-cycle

simulation with the real utilization of the considered aircraft. This enables the user to very accurately calculate the due-date of tasks having a flight-hour or flight-cycle limit for a short term horizon.

2b: Is the final framework acceptable in terms of computational time, optimization and assumptions?

The computational time requirement, set by the airline, was a maximum of 12 hours for the tactical stage models. Results indicated that the approximation algorithm, even for restricted cases, remains within the area of 900 seconds. The computational time of the exact method, at the tactical stage, shows a strong increasing trend when the problem becomes more restricted. However, results indicate that the computational time, for a restricted case where the man-hours factor is equal to 0.4 (i.e. the available man-hours at each of the maintenance days is multiplied with 0.4), is approximately 7 hours. This means that the optimal solution could still be found in a reasonable amount of time. The computational time requirements, set by the airline, was a maximum of 10 minutes for the operational stage models. Both the approximation operational stage level 1 and the operational stage level 2 model have a computational time of approximately 30 seconds. The exact method operational stage level 1 was solved in a few minutes. The models at the operational stage level 2 show similar computational results.

The optimization results indicate that the approximation algorithm is capable of finding the optimal solution for non-restricted cases. The model was further tested for more restricted cases. Results indicate that the approximation algorithm of the tactical stage model, for the most restricted tested case, was 4.9 per cent worse than the optimal solution. In this situation a significant amount of tasks are allocated to their non-optimal maintenance opportunity, which is an unwanted scenario for most airlines. It is expected that most airlines will try to avoid this situation by hiring additional maintenance personnel, shifting tasks to line maintenance or subcontracting the tasks to third party MRO suppliers. Therefore, it may be concluded that the optimization results, for normal conditions, deliver the desired quality.

During this research it was assumed that engineers produce eight effective man-hours per day. This assumption was further tested in the sensitivity analysis where the man-hours factor was reduced to 0.4. This corresponds to 3.2 effective man-hours per workday. Results indicate that the models were still capable of finding a solution for the mentioned situation. Another main assumption was that any task can be executed simultaneously on the same aircraft. In reality this is not always possible, however such information was not available for this research. The output of the last operational level model enables the maintenance manager to identify such tasks and to allocate them accordingly. Therefore, it may be concluded that the main assumptions made during this research are acceptable.

Final remarks

The task allocation framework and its corresponding models show significant improvements compared to current approaches of solving the task allocation problem for an entire fleet. The research objective was to develop a maintenance tasks packaging model for all the tasks required to ensure ongoing airworthiness of the aircraft. It may be concluded that this objective is not entirely met. The developed framework and its corresponding models were only tested for all A and C-check tasks for each aircraft as provided by the airline. However, the computational results of the tactical stage approximation algorithm is much lower than expected and it might therefore be capable in solving the task allocation problem for all tasks necessary to ensure ongoing airworthiness for an entire fleet. Overall, this research can be seen as an important first step in solving the task allocation problem for an entire fleet. In addition, the model framework enables the user to gain better insights into the distribution of man-hours per skill type over a time horizon of 4 years. This enables the airline to better adapt their recruitment process to future needs. Also, the outcome is a fully automated process that enables the airline to react adequately to deviations from flight plans or unforeseen events. In the next chapter recommendations will be given on how the obtained results, during this research, can be further improved.

Novelty of the research
The author of this thesis claims the following about his research:

- This research is the first to present an optimization model to solve the task allocation problem, being solved at the work shift level.
- This research is the first to propose a MILP formulation of the task allocation problem that deals with maintenance tasks individually.
- This research is the first to present an approximation algorithm to solve the task allocation problem and to benchmark its performance using a MILP formulation, solved with a commercial solver.

11 Recommendations

The recommendations for improving the results of this research are as follows:

- The exact model at the tactical stage shows positive results for the fleet of 45 aircraft. It may be expected that the problem becomes more difficult for larger fleets or cases where more than three aircraft compete for the same man-hours at a specific day. Therefore, it is recommended to further improve and extend the model for larger fleet sizes. I would like to advise researchers to use parallel programming in the scheduling loop, of the exact method at the tactical stage, and to implement enough fictitious maintenance opportunities in order to ensure that the problem can be solved in a single iteration.
- The performance of the approximation algorithms could be improved by further subdividing the defined classes. Furthermore, the final solution obtained by the approximation algorithms could be further improved by using a genetic algorithm or any other approximation algorithm that is capable of searching possible swaps of tasks that improve the found objective function value. It is expected that especially the solution quality of the model at the operational stage level 1 benefits from this.
- The current approach does not reserve man-hours for the opening or closing of access panels at the tactical stage models. This means that in restricted cases, the operational level 2 solution requires fictitious man-hours. Therefore, it is recommended to reserve a subset of man-hours for the opening or closing of access panels at each maintenance opportunity. It is advised to find the average required man-hours for opening and closing of access panels at both A and C-check maintenance opportunities. These numbers should then be subtracted from the available man-hours at the tactical stage model and operational level 1 model for each of the A and C-check maintenance opportunities.
- The output of the tactical stage model is the amount of man-hours required per aircraft and per maintenance time segment. Currently, the remaining man-hours, for each time segment, will be divided equally between the aircraft competing for the respective time segment. It is recommended to look further into the distribution of remaining man-hours to perhaps decrease the average maintenance downtime of the fleet.
- The approximation algorithm, at the tactical stage, can be used to test new maintenance strategies. This was indicated in the sensitivity analysis chapter, where results of allocating small C-checks tasks to small A-/C-check maintenance opportunities were presented. The approximation algorithm at the tactical stage has a very reasonable computational time and is therefore especially useful for trying new maintenance approaches. For example, an approach that considers each of the overnight stays, at the home base, as maintenance opportunity could be implemented.
- During this research the main assumption made was that each of the tasks can be executed simultaneously. However, to which extent this assumption holds in practice is unknown. There-

fore, before using the proposed operational stage level 2 model it is recommended to further research this assumption and its effects on the outcome of the models.

- Another limitation of this study is that it is assumed that spare parts, tools and equipment are always available. However, in reality this will not always be possible. Therefore, it is recommended to further investigate how this assumption affects the final results.
- The operational stage level 1 model enables the user to insert additional tasks and to correct the flight-hours and flight-cycles utilization. To what extent these additional options affect the overall solution quality of the entire fleet is unknown. Therefore, it might be interesting to see how the addition of new tasks and utilization correction influence the solution quality of the entire fleet.
- The model at the operational stage level 2 was used to prove that the model framework works. The model was aimed at reducing the aircraft downtime by preventing the repetitive opening of access panels and by allocating inspection tasks as early as possible. However, reducing the aircraft maintenance downtime is a study on its own. Therefore, it is recommended to further extend the model at the operational stage level 2 with the latest research on this subject.

A

Tactical stage output file

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	MO	Start date	End date	Used GR1	Used GR2	Used GR4	Used ESHS	Used ICH	Used PINT	Used MAP	Used NDT	# days	Rem GR1	Rem GR2	Rem GR4	Rem ESHS	Rem ICH	Rem PINT	Rem MAP	Rem NDT
2	A 1.28	2017-11-15	2017-11-15	0	0	0	0	0	0	0	0	1	230,902	123,3569	117,9608	442,6039	99,76471	136,0314	79,56078	44,8
3	A 2.28	2018-01-11	2018-01-11	0	0	0	0	0	0	0	0	1	231,936	123,136	118,912	434,816	100,864	135,808	79,232	44,8
4	A 3.28	2018-03-20	2018-03-20	0	0	0	0	0	0	0	0	1	116,6	62,06667	60,33333	221,2	50,86667	69,4	39,53333	22,4
5	A 4.28	2018-05-23	2018-05-23	0	0	0	0	0	0	0	0	1	115,451	61,67843	58,98039	221,302	49,88235	68,01569	39,78039	22,4
6	A 1.29	2018-07-25	2018-07-25	0	0	0	0	0	0	0	0	1	115,451	61,67843	58,98039	221,302	49,88235	68,01569	39,78039	22,4
7	A 2.29	2018-09-25	2018-09-25	31,4255	10,4671	7,37197	0,407295	4,4365	0,312	1,32604	0	1	89,65973	50,35218	53,43323	220,4964	46,6168	69,244	38,38919	22,4
8	C 12.1	2018-11-27	2018-11-28	0	0	0	0	0	0	0	0	2	307,5147	180,2494	130,2421	293,7237	259,7197	91,14246	137,2542	29,867
9	C 12.1	2018-11-29	2018-12-10	395,54	409,089	159,268	161,4221	314,629	105,092	128,394	7,5	12	1710,511	890,0723	701,5818	1652,569	1381,657	482,7624	744,3048	175,45
10	A 4.29	2019-02-18	2019-02-18	47,1442	14,7833	12,2459	4,240125	6,11617	1,71597	2,51304	0	1	181,5918	108,2247	104,3621	437,2319	93,33983	134,092	74,79896	44,8
11	A 1.30	2019-04-23	2019-04-23	43,4061	22,2386	9,67026	3,907295	4,4365	0,312	1,32604	1	1	87,04773	44,65089	53,61444	218,4964	46,6168	69,244	38,38919	21,9
12	A 2.30	2019-06-26	2019-06-26	47,8312	34,8765	5,49546	5,435094	8,85048	2,37066	3,67016	0,5	1	71,10529	35,73283	49,40413	218,0844	43,42549	66,83035	37,41418	22,15
13	A 3.30	2019-08-26	2019-08-26	40,4057	23,0593	7,69483	2,407295	4,4365	0,312	1,32604	0	1	188,3303	99,94867	108,9132	439,0647	95,0195	135,496	75,98596	44,8
14	A 4.30	2019-10-29	2019-10-29	39,2457	26,713	25,794	2,198155	7,95783	1,08092	2,78094	0,5	1	78,67018	39,73799	43,97195	219,6009	44,00962	68,85954	38,09876	22,15
15	A 1.31	2020-01-06	2020-01-06	35,9793	25,8298	7,6996	1	4,06324	0	1,06226	0	1	192,7567	97,17817	108,9084	440,472	95,39276	135,808	76,24974	44,8
16	A 2.31	2020-03-16	2020-03-16	66,2833	19,114	8,50603	2	4,06324	0	1,06226	3	1	162,4527	103,894	108,102	439,472	95,39276	135,808	76,24974	41,8
17	C 1.2 +	2020-05-13	2020-05-14	0	0	0	0	0	0	0	0	2	307,4246	177,0006	129,267	292,14	258,1551	90,61312	136,1641	29,867
18	C 1.2 +	2020-05-15	2020-05-17	0	0	0	0	0	0	0	0	3	0,391258	0,179917	45,74084	146,9762	83,03096	20,08123	38,0431	13,4
19	C 1.2 +	2020-05-18	2020-05-19	0	4,9787	0	0,599078	1,9473	0,54046	1,35934	0	2	314,4064	176,3647	130,5624	294,091	259,3636	91,35585	136,8599	29,867
20	C 1.2 +	2020-05-20	2020-05-29	298,946	185,315	108,406	108,4339	190,972	59,8997	68,4047	12,5	10	1650,777	926,5908	687,7346	1611,589	1368,443	475,9642	738,8453	171,95
21	A 4.31	2020-07-30	2020-07-30	33,7947	13,6016	12,204	0	4,06324	0	1,06226	0,5	1	198,1413	109,5344	106,708	434,816	96,80076	135,808	78,16974	44,3
22	A 1.32	2020-09-29	2020-09-29	42,5752	27,7945	12,2426	3,907295	4,4365	0,312	1,32604	0	1	190,6248	96,33881	108,4241	438,4927	97,29683	138,488	77,74063	44,8
23	A 2.32	2020-12-02	2020-12-02	37,517	15,8682	7,4331	3,314591	4,80976	0,62399	1,58981	1	1	193,385	107,4886	110,5277	439,2893	94,95495	135,4074	77,97097	43,8
24	A 3.32	2021-02-08	2021-02-08	53,1114	27,6449	7,71746	3,64742	6,48943	2,02797	2,77681	0	1	175,6246	95,36313	108,8905	437,8246	92,96657	133,78	74,53519	44,8
25	A 4.32	2021-04-20	2021-04-20	46,3714	19,2776	19,2889	2	4,06324	0	1,06226	0	1	186,8286	104,8557	101,3777	440,4	97,67009	138,8	78,00441	44,8
26	A 1.33	2021-06-23	2021-06-23	38,2608	17,489	9,62656	2,907295	4,4365	0,312	1,32604	1	1	71,71233	44,80524	49,91953	219,3483	45,63248	67,85969	38,63624	21,9
27	A 2.33	2021-08-19	2021-08-19	33,0906	20,5091	9,97761	1	4,06324	0	1,06226	0,5	1	198,8454	102,6269	108,9344	433,816	96,80076	135,808	78,16974	44,3
28	A 3.33	2021-10-19	2021-10-19	52,6291	38,3342	10,7527	4,935094	8,85048	2,37066	3,67016	0	1	180,5709	85,79911	109,914	437,4649	92,88285	136,4293	75,39651	44,8
29	A 4.33	2021-12-27	2021-12-27	10,6493	9,48538	3,94	0	4,06324	0	1,06226	0	1	218,0867	113,5226	112,668	441,472	95,39276	135,808	76,24974	44,8

Figure A.1: Screenshot of tactical stage output excel file.

B

Operational stage level 1 output file

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	MO	Begin date	End date	ITEM	BLOCK	SKILL	CAT TASK	LAST EXEC DT	PER FH	PER FC	PER CALEND	MH GR1	MH GR2	MH GR4	MH ESHS	MH ICH	MH PINT	MH MAP	MH NDT	LIMIT EXEC DT	RUL in days
2	A 1.14	2018-08-21	2018-08-21	562113-TP-1	INSP	GR2	A-Task	2018-08-02	750	750	4 M	0,0589	3,84904	0,033	0	0	0	0	0	2018-10-03	43
3	A 1.14	2018-08-21	2018-08-21	242000-21-1	INSP	GR1	A-Task	2018-08-02	800	0	0	0,5908	0,00313	0,0037	0	0	0	0	0	2018-10-07	47
4	A 1.14	2018-08-21	2018-08-21	242000-21-1	INSP	GR1	A-Task	2018-08-02	800	0	0	0,5908	0,00313	0,0037	0	0	0	0	0	2018-10-07	47
5	A 1.14	2018-08-21	2018-08-21	242000-21-1	INSP	GR1	A-Task	2018-08-02	800	0	0	0,5908	0,00313	0,0037	0	0	0	0	0	2018-10-07	47
6	A 1.14	2018-08-21	2018-08-21	242000-21-1	INSP	GR1	A-Task	2018-08-02	800	0	0	0,5908	0,00313	0,0037	0	0	0	0	0	2018-10-07	47
7	A 1.14	2018-08-21	2018-08-21	242000-21-1	INSP	GR1	A-Task	2018-08-02	800	0	0	0,5908	0,00313	0,0037	0	0	0	0	0	2018-10-07	47
8	A 1.14	2018-08-21	2018-08-21	242000-21-1	INSP	GR1	A-Task	2018-08-02	800	0	0	0,5908	0,00313	0,0037	0	0	0	0	0	2018-10-07	47
9	A 1.14	2018-08-21	2018-08-21	253000-02-1	LUB	GR2	A-Task	2018-08-02	750	0	0	0	2	0	0	0	0	0	0	2018-10-03	43
10	A 1.14	2018-08-21	2018-08-21	253000-02-1	LUB	GR2	A-Task	2018-08-02	750	0	0	0	1	0	0	0	0	0	0	2018-10-03	43
11	A 1.14	2018-08-21	2018-08-21	781112-TP-1	INSP	GR1	A-Task	2018-08-02	750	750	4 M	1,1817	0,00627	0,0074	0	0	0	0	0	2018-10-03	43
12	A 1.14	2018-08-21	2018-08-21	781112-TP-1	INSP	GR1	A-Task	2018-08-02	750	750	4 M	1,1817	0,00627	0,0074	0	0	0	0	0	2018-10-03	43
13	A 1.14	2018-08-21	2018-08-21	251100-TP-5	INSP	ICH	A-Task	2018-08-02	750	750	4 M	0	0	0	0	2,0316	0	0,03113	0	2018-10-03	43
14	A 1.14	2018-08-21	2018-08-21	251100-TP-1	INSP	ICH	A-Task	2018-08-02	750	750	4 M	0	0	0	0	2,0316	0	0,03113	0	2018-10-03	43
15	A 1.14	2018-08-21	2018-08-21	742130-TP-1	LUB	GR1	A-Task	2018-08-02	750	750	4 M	0,5	0	0	0	0	0	0	0	2018-10-03	43
16	A 1.14	2018-08-21	2018-08-21	742130-TP-1	LUB	GR1	A-Task	2018-08-02	750	750	4 M	0,5	0	0	0	0	0	0	0	2018-10-03	43
17	A 1.14	2018-08-21	2018-08-21	562113-TP-2	LUB	GR2	A-Task	2018-08-02	750	750	4 M	0	0,1	0	0	0	0	0	0	2018-10-03	43
18	A 1.14	2018-08-21	2018-08-21	562113-TP-2	LUB	MAP	A-Task	2018-08-02	750	750	4 M	0	0	0	0	0	0	0,9	0	2018-10-03	43
19	A 1.14	2018-08-21	2018-08-21	330000-TP-1	TEST	GR4	A-Task	2018-08-02	750	750	4 M	0	0	1	0	0	0	0	0	2018-10-03	43
20	A 1.14	2018-08-21	2018-08-21	255000-16-1	INSP	GR2	A-Task	2018-08-02	750	0	0	0,0039	0,2566	0,0022	0	0	0	0	0	2018-10-03	43
21	A 1.14	2018-08-21	2018-08-21	255000-16-1	INSP	GR2	A-Task	2018-08-02	750	0	0	0,0039	0,2566	0,0022	0	0	0	0	0	2018-10-03	43
22	A 1.14	2018-08-21	2018-08-21	255000-16-1	INSP	GR2	A-Task	2018-08-02	750	0	0	0,0039	0,2566	0,0022	0	0	0	0	0	2018-10-03	43
23	A 1.14	2018-08-21	2018-08-21	323000-19-1	INSP	GR1	A-Task	2018-08-02	750	0	0	0,2363	0,00125	0,0015	0	0	0	0	0	2018-10-03	43
24	A 1.14	2018-08-21	2018-08-21	323000-19-1	INSP	GR1	A-Task	2018-08-02	750	0	0	0,2363	0,00125	0,0015	0	0	0	0	0	2018-10-03	43
25	A 1.14	2018-08-21	2018-08-21	323000-19-1	INSP	GR1	A-Task	2018-08-02	750	0	0	0,2363	0,00125	0,0015	0	0	0	0	0	2018-10-03	43
26	A 1.14	2018-08-21	2018-08-21	323000-01-1	INSP	GR1	A-Task	2018-08-02	750	0	4 M	0,2363	0,00125	0,0015	0	0	0	0	0	2018-10-03	43
27	A 1.14	2018-08-21	2018-08-21	323000-01-1	INSP	GR1	A-Task	2018-08-02	750	0	4 M	0,2363	0,00125	0,0015	0	0	0	0	0	2018-10-03	43
28	A 1.14	2018-08-21	2018-08-21	323000-01-1	INSP	GR1	A-Task	2018-08-02	750	0	4 M	0,2363	0,00125	0,0015	0	0	0	0	0	2018-10-03	43
29	A 1.14	2018-08-21	2018-08-21	253000-TP-7	INSP	GR2	A-Task	2018-08-02	750	750	4 M	0,0098	0,64151	0,0055	0	0	0	0	0	2018-10-03	43

Figure B.1: Screenshot of operational stage level 1 output excel file.

C

Operational stage level 2 output file

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	SHIFT	ITEM	Complete task	BLOCK	SKILL	MXH GR1	MXH GR2	MXH GR4	MXH ESHS	MXH ICH	MXH PINT	MXH MAP	MXH NDT
2	Morning Shift 0	Open access panels	Complete task		GR1	2,1	0						
3	Morning Shift 0	Open access panels	Complete task		GR1	2,1	0						
4	Morning Shift 0	Open access panels	Complete task		GR1	27,4	37,1						
5	Morning Shift 0	Open access panels	Complete task		GR1	12,9	0						
6	Morning Shift 0	Open access panels	Complete task		GR1	0,1	0						
7	Morning Shift 0	Open access panels	Complete task		GR2	0	0,1						
8	Morning Shift 0	Open access panels	Complete task		GR2	0	0,1						
9	Morning Shift 0	Open access panels	Complete task		GR2	0	0,2						
10	Morning Shift 0	Open access panels	Complete task		GR2	0	0,2						
11	Morning Shift 0	Open access panels	Complete task		GR1	0,4	0						
12	Morning Shift 0	Open access panels	Complete task		GR1	0,4	0						
13	Morning Shift 0	200127-02-1	Part of task	INSP	GR4	0	0	4	0	0	0	0	0
14	Morning Shift 0	200127-02-1	Part of task	INSP	GR4	0	0	0,56437	0	0	0	0	0
15	Morning Shift 0	200127-07-1	Part of task	INSP	GR4	0	0	4	0	0	0	0	0
16	Morning Shift 0	200127-07-1	Part of task	INSP	GR4	0	0	0,56437	0	0	0	0	0
17	Morning Shift 0	554009-01-1	Part of task	INSP	NDT	0	0	0	0	0	0	0	4
18	Morning Shift 0	200171-01-1	Part of task	INSP	GR4	0	0	4	0	0	0	0	0
19	Morning Shift 0	200171-01-1	Part of task	INSP	GR4	0	0	0,56437	0	0	0	0	0
20	Morning Shift 0	200125-05-1	Part of task	INSP	GR4	0	0	4	0	0	0	0	0
21	Morning Shift 0	200125-05-1	Part of task	INSP	GR4	0	0	0,21587	0	0	0	0	0
22	Morning Shift 0	200125-05-1	Part of task	INSP	GR4	0	0	4	0	0	0	0	0
23	Morning Shift 0	335100-06-1	Part of task	INSP	GR4	0	0	4	0	0	0	0	0
24	Morning Shift 0	281800-05-1	Part of task	INSP	ICH	0	0	0	0	4	0	0	0
25	Morning Shift 0	281800-05-1	Part of task	INSP	ICH	0	0	0	0	2,7559	0	0	0
26	Morning Shift 0	281800-05-1	Part of task	INSP	ICH	0	0	0	0	4	0	0	0
27	Morning Shift 0	200001-02-1	Part of task	INSP	GR4	0	0	4	0	0	0	0	0
28	Morning Shift 0	ZL-127-02-2	Complete task	INSP	GR2	0	2,3787	0	0	0	0	0	0
29	Morning Shift 0	ZL-127-02-2	Part of task	INSP	ICH	0	0	0	0	2,7559	0	0	0
30	Morning Shift 0	ZL-127-02-2	Part of task	INSP	ICH	0	0	0	0	4	0	0	0
31	Morning Shift 0	ZL-127-02-2	Part of task	INSP	ICH	0	0	0	0	4	0	0	0
32	Morning Shift 0	534164-01-1	Part of task	INSP	GR2	0	4	0	0	0	0	0	0
33	Morning Shift 0	534164-01-1	Part of task	INSP	GR2	0	3,8935	0	0	0	0	0	0
34	Morning Shift 0	534164-01-1	Part of task	INSP	GR2	0	4	0	0	0	0	0	0
35	Morning Shift 0	534164-01-1	Complete task	LUB	GR2	0	0,1	0	0	0	0	0	0
36	Morning Shift 0	534164-01-1	Complete task	LUB	MAP	0	0	0	0	0	0	0,5	0
37	Morning Shift 0	282100-TP-1	Part of task	LUB	GR1	4	0	0	0	0	0	0	0
38	Morning Shift 0	282100-TP-1	Part of task	LUB	GR1	3,5	0	0	0	0	0	0	0
39	Morning Shift 0	282100-TP-1	Part of task	LUB	GR1	4	0	0	0	0	0	0	0
40	Morning Shift 0	282100-TP-1	Complete task	TEST	GR1	1,5	0	0	0	0	0	0	0
41	Morning Shift 0	282100-TP-1	Part of task	INSP	GR1	2,9238	0	0	0	0	0	0	0
42	Morning Shift 0	282100-TP-1	Part of task	INSP	GR1	4	0	0	0	0	0	0	0
43	Morning Shift 0	282100-TP-1	Part of task	INSP	GR1	4	0	0	0	0	0	0	0
44	Morning Shift 0	321100-15-1	Complete task	ABAC	GR1	0,1	0	0	0	0	0	0	0
45	Morning Shift 0	321100-15-1	Part of task	INSP	GR1	1,1644	0	0	0	0	0	0	0
46	Morning Shift 0	252331-TP-1	Complete task	ABAC	ICH	0	0	0	0	3	0	0	0
47	Morning Shift 0	252331-TP-1	Complete task	INSP	GR2	0	3,56805	0	0	0	0	0	0
48	Morning Shift 0	252331-TP-1	Part of task	INSP	ICH	0	0	0	0	2,7559	0	0	0
49	Morning Shift 0	252331-TP-1	Part of task	INSP	ICH	0	0	0	0	4	0	0	0
50	Morning Shift 0	252331-TP-1	Part of task	INSP	ICH	0	0	0	0	4	0	0	0

C 7.1 + A 3.16

Figure C.1: Screenshot of operational stage level 2 output excel file.

D

Exact method solution

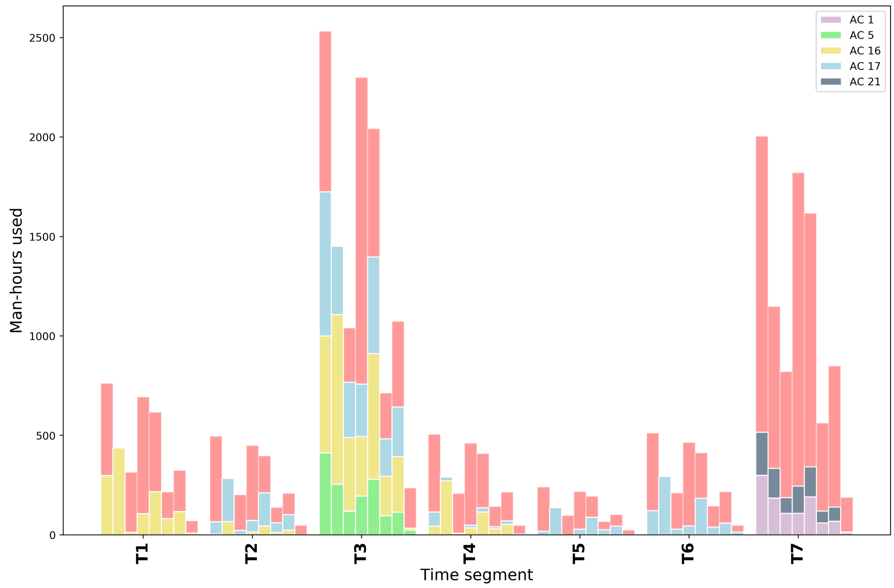

Figure D.1: Exact method: Man-hours per skill type used per time segments for the man-hours factor equal to 0.42.

E

AIRMES maintenance schedule overview

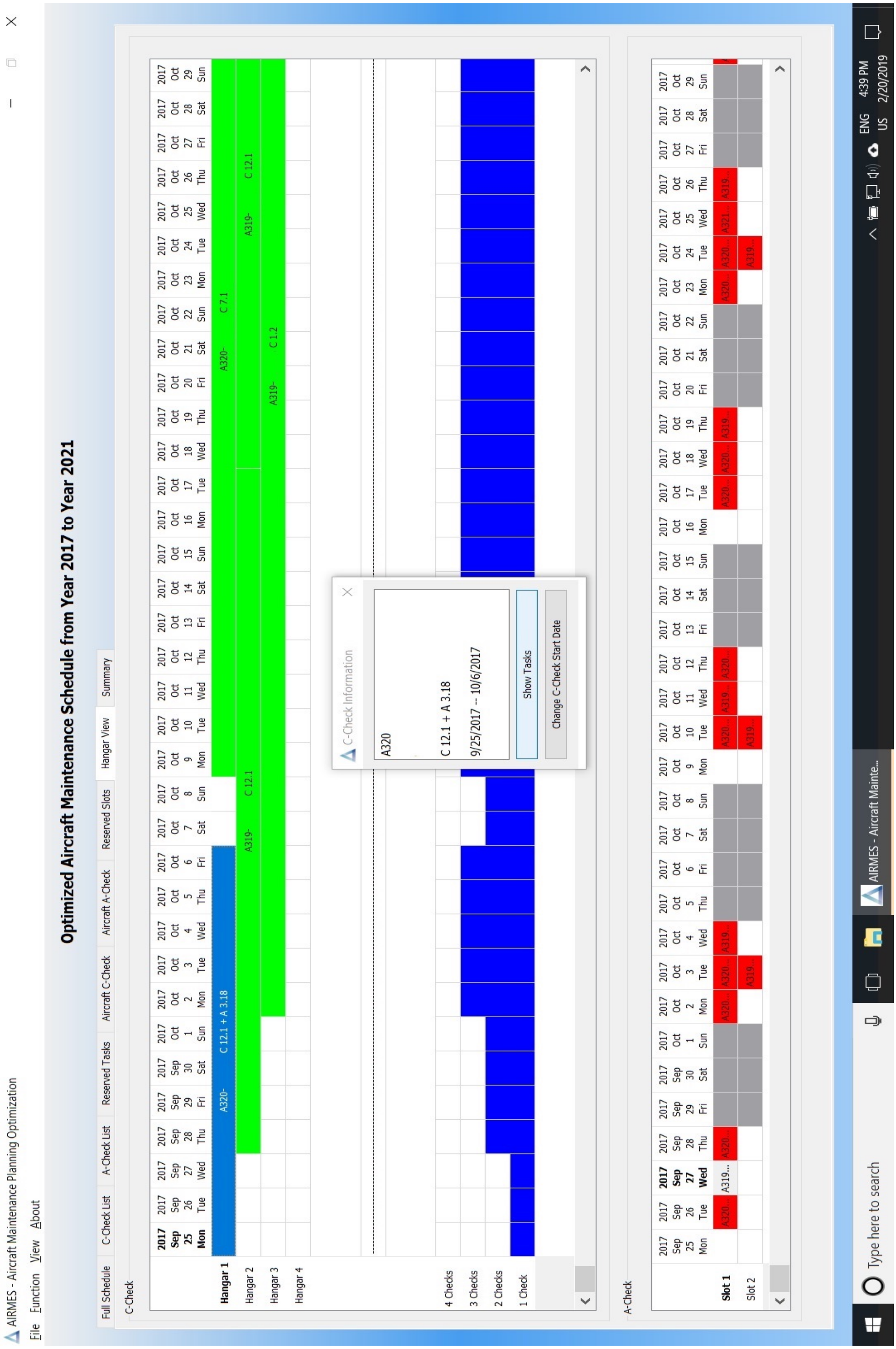

Figure E.1: AIRMES screenshot maintenance schedule overview.

F

AIRMES task function

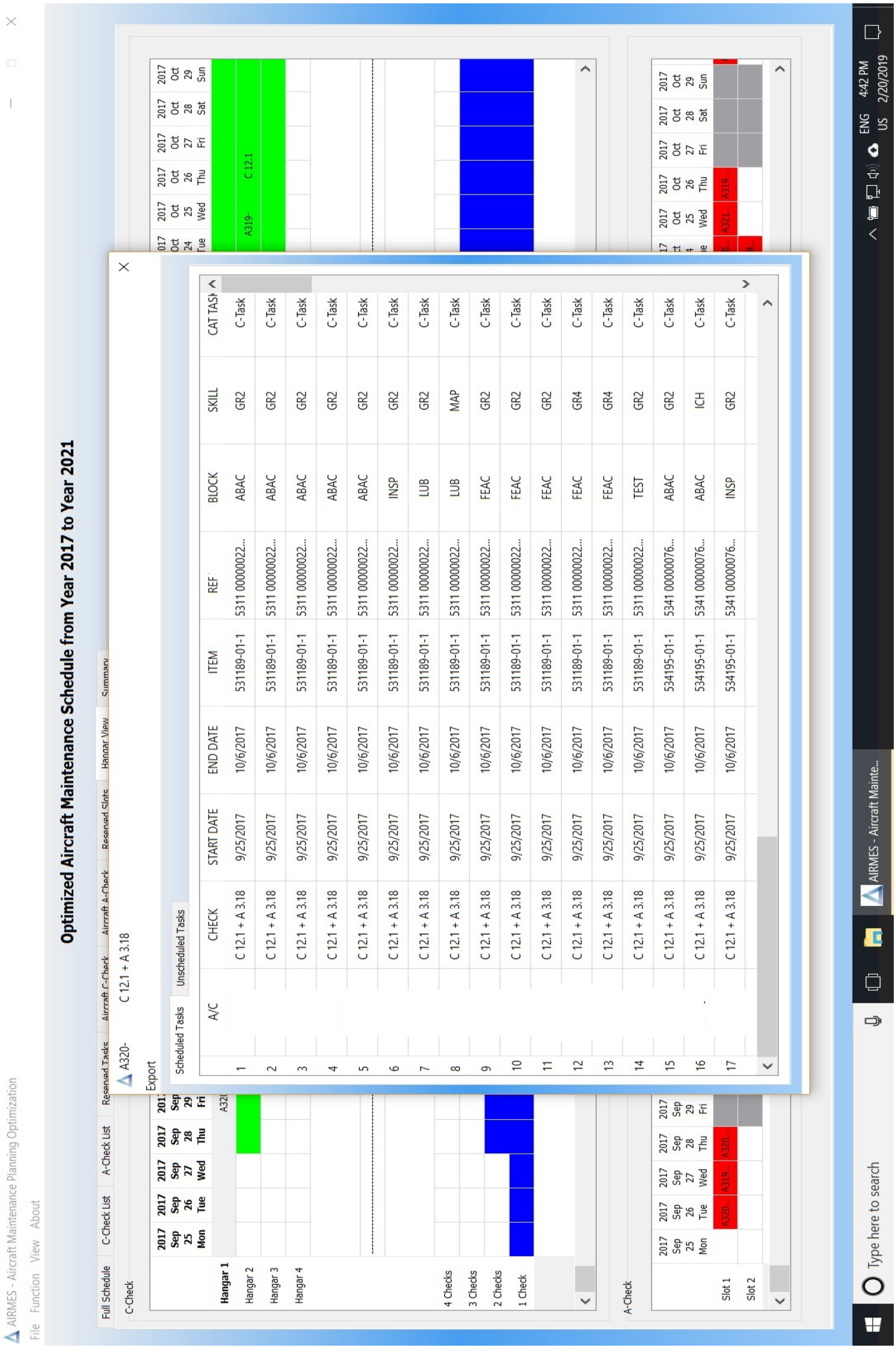

Figure F.1: AIRMES screenshot with task planning function.

G

Sensitivity 1 results

# Factor	EXACT: OBJ value	time (s)	Approx: OBJ value	time (s)
0.6	599,026	1,213	599,193	845
0.59	599,026	1257	599,193	846
0.58	599,026	1,249	599,195	850
0.57	599,101	1,269	599,359	855
0.56	599,596	1,274	599,885	870
0.55	599,927	1,282	600,609	880
0.54	600,381	1,286	601,752	885
0.53	600,815	1,296	602,988	890
0.52	601,389	1,300	604,623	900
0.51	601,761	1,305	606,621	905
0.50	602,350	2,517	609,394	910
0.49	603,143	2,724	611,858	915
0.48	605,671	2,725	615,400	920
0.47	610,319	3,448	619,077	925
0.46	613,193	4,975	624,849	930
0.45	616,107	5,235	631,183	932
0.44	619,021	5,880	636,840	935
0.43	622,177	6,757	643,181	939
0.42	625,870	15,321	650,198	942
0.41	630,367	20,906	658,539	945
0.40	635,627	24,614	667,593	948

Table G.1: Sensitivity analysis 1 simulation results tactical level models.

H

Sensitivity 2 results

# Factor	EXACT V2: OBJ value	time (s)	EXACT V1: OBJ value	time (s)	Approx: OBJ value	time (s)
2.0	540,708	1,175	540,709	950	543,410	540
2.1	566,989	1,178	566,991	952	569,755	545
2.2	593,234	1,180	593,274	954	596,041	546
2.3	619,515	1,181	619,539	955	622,372	547
2.4	645,830	1,185	645,834	958	648,697	549
2.5	673,130	1,189	673,139	960	676,149	550
2.6	700,827	1,350	700,862	1,100	705,019	551
2.7	728,575	1,500	728,600	1,350	734,642	554
2.8	756,303	1,690	756,316	1,500	764,593	560
2.9	784,078	1,720	784,172	1,750	795,916	565
3.0	815,213	1,843	815,291	2,200	829,060	568
3.1	849,219	1,900	849,433	2,300	864,072	569
3.2	883,269	1,953	883,582	2,520	898,861	571
3.3	925,760	2,400	928,513	2,900	937,221	572
3.4	973,580	2,893	978,858	3,200	980,646	574
3.5	1,012,888	3,000	1,020,496	3,800	1,025,637	577
3.6	1,055,468	3,100	1,063,267	4,100		
3.7	1,098,023	3,250	1,107,573	4,200		
3.8	1,142,943	3,300	1,153,038	4,500		
3.9	1,186,136	3,500	1,200,535	4,780		
4.0	1,229,905	3,600	1,244,906	4,900		

Table H.1: Sensitivity analysis 2 simulation results tactical level models.

I

Different maintenance schedules

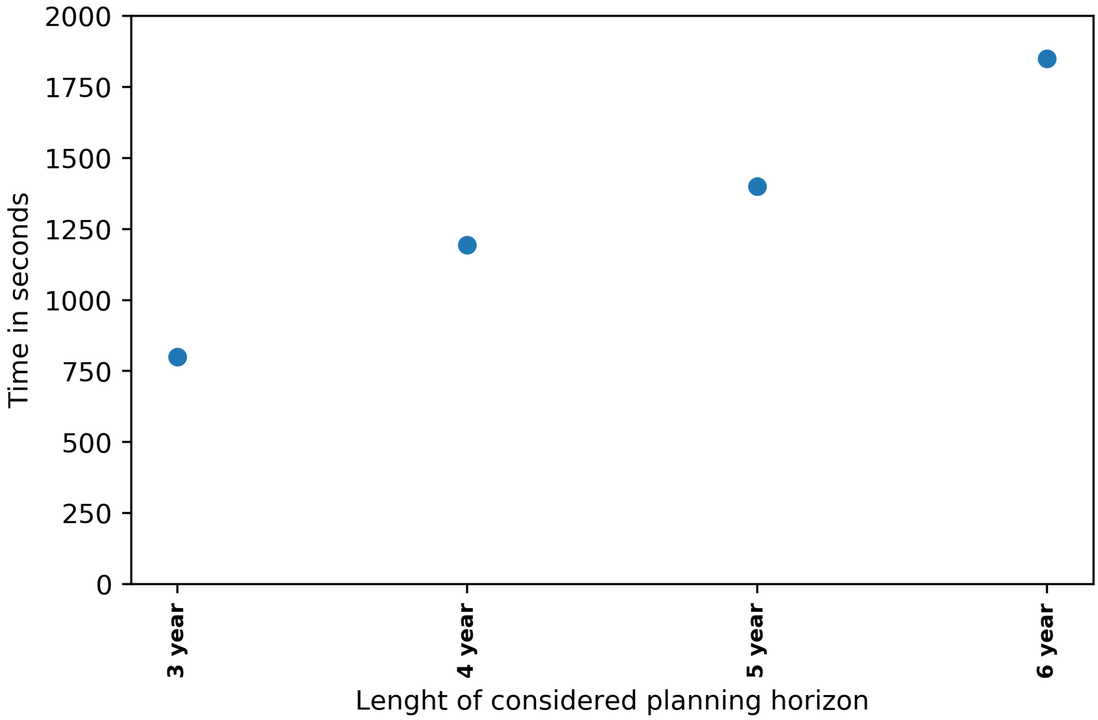

Figure I.1: Computational times for different planning horizon lengths.

Bibliography

[1] N. Papakostas, P. Papachatzakis, V. Xanthakis, D. Mourtzis, and G. Chryssolouris, *An approach to operational aircraft maintenance planning,* Decision Support Systems **48**, 604 (2010).

[2] A. Steiner, *A heuristic method for aircraft maintenance scheduling under various constraints,* in *Proceedings of the 6th Swiss Transport Research Conference, Ascona, Switzerland* (2006).

[3] C. Senturk, M. S. Kavsaoglu, and M. Nikbay, *Optimization of aircraft utilization by reducing scheduled maintenance downtime,* in *10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference* (2010) p. 9143.

[4] M. A. Pereira and J. A. Babu, *Information support tool for aircraft maintenance task planning,* International Advanced Research Journal in Science, Engineering and Technology **3** (2016).

[5] H. A. Kinnison and T. Siddiqui, *Aviation maintenance management* (2012).

[6] M. Witteman, *A maintenance task packaging and scheduling model applicable to aircraft maintenance (unpublished),* (2018).

[7] A. Ahmadi, P. Söderholm, and U. Kumar, *On aircraft scheduled maintenance program development,* Journal of Quality in Maintenance Engineering **16**, 229 (2010).

[8] I. Ozkol and C. Senturk, *The effects of the use of single task-oriented maintenance concept and more accurate letter check alternatives on the reduction of scheduled maintenance downtime of aircraft,* in *Mechanical and Aerospace Engineering (ICMAE), 2017 8th International Conference on* (IEEE, 2017) pp. 67–74.

[9] F. Gargiulo, D. Pascar, and S. Venticinque, *A multi-agent and dynamic programming algorithm for aeronautical maintenance planning,* in *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on* (IEEE, 2013) pp. 410–415.

[10] *Maintenance Planning Document A320*, Document Revision 34, Nov 01/10 (Airbus, 2010).

[11] Q. Hao, W. Shen, Y. Xue, and S. Wang, *Task network-based project dynamic scheduling and schedule coordination,* Advanced Engineering Informatics **24**, 417 (2010).

[12] N. Hölzel, C. Schröder, T. Schilling, and V. Gollnick, *A maintenance packaging and scheduling optimization method for future aircraft,* (2012).

[13] P. Samaranayake and S. Kiridena, *Aircraft maintenance planning and scheduling: an integrated framework,* Journal of Quality in Maintenance Engineering **18**, 432 (2012).

[14] H. K. Alfares, *Aircraft maintenance workforce schedulinga case study,* Journal of Quality in Maintenance Engineering **5**, 78 (1999).

[15] L. J. S. Rosales, J.-B. Yang, and Y.-W. Chen, *Analysing delays and disruptions in aircraft heavy maintenance,* in *International Conference of the System Dynamics Society* (2014).

[16] P. Samaranayake, *Current practices and problem areas in aircraft maintenance planning and scheduling: interfaced/integrated system perspective,* in *INDUSTRIAL ENGINEERING AND MANAGEMENT SYSTEMS, Proceedings of the 7th asia-pacific conference in bangkok, UWS, Sidney* (2006) pp. 2245–2256.

[17] L. J. S. Rosales, Y.-W. Chen, and J.-B. Yang, *Estimation of uncertain unscheduled activities in aircraft maintenance using er rule,* in *Transportation Information and Safety (ICTIS), 2015 International Conference on* (IEEE, 2015) pp. 675–681.

[18] J. R. Grey, *Buffer techniques for stochastic resource constrained project scheduling with stochastic task insertions problems*, Ph.D. thesis, University of Central Florida (2007).

[19] J.-B. Yang and D.-L. Xu, *Evidential reasoning rule for evidence combination,* Artificial Intelligence **205**, 1 (2013).

[20] M. Kroes, W. Watkins, F. Delp, and R. Sterkenburg, *Aircraft Maintenance and repair* (2013).

[21] H. Li, H. Zuo, D. Lei, K. Liang, and T. Lu, *Optimal combination of aircraft maintenance tasks by a novel simplex optimization method,* Mathematical Problems in Engineering **2015** (2015).

[22] H. Li, H. Zuo, K. Liang, J. Xu, J. Cai, and J. Liu, *Optimizing combination of aircraft maintenance tasks by adaptive genetic algorithm based on cluster search,* Journal of Systems Engineering and Electronics **27**, 140 (2016).

[23] A. K. Muchiri and S. Klaas, *Application of maintenance interval de-escalation in base maintenance planning optimization,* Enterprise Risk Management **1** (2009).

[24] C. Sriskandarajah, A. Jardine, and C. Chan, *Maintenance scheduling of rolling stock using a genetic algorithm,* Journal of the Operational Research Society **49**, 1130 (1998).

[25] A. Muchiri, *Optimizing aircraft line maintenance through task re-clustering and interval de-escalation,* Sustainable Research and Innovation Proceedings **3** (2011).

[26] C. Vanbuskirk, B. Dawant, G. Karsai, J. Sprinkle, G. Szokoli, R. Currer, *et al.*, *Computer-aided aircraft maintenance scheduling,* (2002).

[27] M. P. Kleeman and G. B. Lamont, *Solving the aircraft engine maintenance scheduling problem using a multi-objective evolutionary algorithm,* in *International Conference on Evolutionary Multi-Criterion Optimization* (Springer, 2005) pp. 782–796.

[28] N. Safaei, D. Banjevic, and A. K. Jardine, *Workforce-constrained maintenance scheduling for military aircraft fleet: a case study,* Annals of Operations Research **186**, 295 (2011).

[29] G. Quan, G. W. Greenwood, D. Liu, and S. Hu, *Searching for multiobjective preventive maintenance schedules: Combining preferences with evolutionary algorithms,* European Journal of Operational Research **177**, 1969 (2007).

[30] G. Cavory, R. Dupas, and G. Goncalves, *A genetic approach to the scheduling of preventive maintenance tasks on a single product manufacturing production line,* International Journal of Production Economics **74**, 135 (2001).

[31] N. Grudinin, *Reactive power optimization using successive quadratic programming method,* IEEE Transactions on Power Systems **13**, 1219 (1998).

[32] S. Martorell, S. Carlos, A. Sanchez, and V. Serradell, *Constrained optimization of test intervals using a steady-state genetic algorithm,* Reliability Engineering & System Safety **67**, 215 (2000).

[33] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed. (Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989).

[34] C. A. C. Coello, *A comprehensive survey of evolutionary-based multiobjective optimization techniques,* Knowledge and Information systems **1**, 269 (1999).

[35] M. R. Garey and D. S. Johnson, *Computers and intractability: a guide to np-completeness,* (1979).

[36] E. K. Burke, M. R. Hyde, and G. Kendall, *Evolving bin packing heuristics with genetic programming,* in *Parallel Problem Solving from Nature-PPSN IX* (Springer, 2006) pp. 860–869.

[37] T. Dokeroglu and A. Cosar, *Optimization of one-dimensional bin packing problem with island parallel grouping genetic algorithms,* Computers & Industrial Engineering **75**, 176 (2014).

[38] B. Korte and J. Vygen, *Combinatorial optimization*, Vol. 2 (Springer, 2011).

[39] S. Ahire, G. Greenwood, A. Gupta, and M. Terwilliger, *Workforce-constrained preventive maintenance scheduling using evolution strategies,* Decision Sciences **31**, 833 (2000).

[40] K. Genova, L. Kirilov, and V. Guliashki, *A survey of solving approaches for multiple objective flexible job shop scheduling problems,* Cybernetics And Information Technologies **15**, 3 (2015).

[41] A. Azzouz, M. Ennigrou, and L. B. Said, *Flexible job-shop scheduling problem with sequence-dependent setup times using genetic algorithm.* in *ICEIS (2)* (2016) pp. 47–53.

[42] X. Li and L. Gao, *A collaborative evolutionary algorithm for multi-objective flexible job shop scheduling problem,* in *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on* (IEEE, 2011) pp. 997–1002.

[43] P. Brucker, B. Jurisch, and A. Krämer, *Complexity of scheduling problems with multi-purpose machines,* Annals of Operations Research **70**, 57 (1997).

[44] M. Dave and K. Choudhary, *Job shop scheduling algorithms—a shift from traditional techniques to non-traditional techniques,* in *Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on* (IEEE, 2016) pp. 169–173.

[45] D. Dinis and A. P. Barbosa-Póvoa, *On the optimization of aircraft maintenance management,* in *Operations Research and Big Data* (Springer, 2015) pp. 49–57.

[46] S. K. Chaharsooghi and A. H. M. Kermani, *An effective ant colony optimization algorithm (aco) for multi-objective resource allocation problem (morap),* Applied mathematics and computation **200**, 167 (2008).

[47] Y. Mati and X. Xie, *Multiresource shop scheduling with resource flexibility and blocking,* IEEE transactions on automation science and engineering **8**, 175 (2011).

[48] J. Błażewicz, W. Domschke, and E. Pesch, *The job shop scheduling problem: Conventional and new solution techniques,* European journal of operational research **93**, 1 (1996).

[49] P. Fattahi, M. S. Mehrabad, and F. Jolai, *Mathematical modeling and heuristic approaches to flexible job shop scheduling problems,* Journal of intelligent manufacturing **18**, 331 (2007).

[50] C. Özgüven, L. Özbakır, and Y. Yavuz, *Mathematical models for job-shop scheduling problems with routing and process plan flexibility,* Applied Mathematical Modelling **34**, 1539 (2010).

[51] M. Abdi Khalife, B. Abbasi, *et al., A simulated annealing algorithm for multi objective flexible job shop scheduling with overlapping in operations,* Journal of Optimization in Industrial Engineering , 17 (2010).

[52] K. Thörnblad, T. Almgren, M. Patriksson, and A.-B. Strömberg, *Mathematical optimization of a flexible job shop problem including preventive maintenance and availability of fixtures,* in *Proceedings of the 4th World P&OM Conference/19th International Annual EurOMA Conference, Amsterdam, Netherlands, July 2012* (2012).

[53] Y. Pleumpirom and S. Amornsawadwatana, *Multiobjective optimization of aircraft maintenance in thailand using goal programming: A decision-support model,* Advances in Decision Sciences **2012** (2012).

[54] K. Deb, *Multi-objective optimization using evolutionary algorithms: an introduction,* Multi-objective evolutionary optimisation for product design and manufacturing , 1 (2011).

[55] S. Meeran and M. Morshed, *A hybrid genetic tabu search algorithm for solving job shop scheduling problems: a case study,* Journal of intelligent manufacturing **23**, 1063 (2012).

[56] W. Xia and Z. Wu, *An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems,* Computers & Industrial Engineering **48**, 409 (2005).

[57] E. G. Coffman Jr, J. Csirik, G. Galambos, S. Martello, and D. Vigo, *Bin packing approximation algorithms: survey and classification,* in *Handbook of combinatorial optimization* (Springer, 2013) pp. 455–531.

[58] L. Epstein, L. M. Favrholdt, and J. S. Kohrt, *Comparing online algorithms for bin packing problems,* Journal of Scheduling **15**, 13 (2012).

[59] D. S. Johnson, *Fast algorithms for bin packing,* Journal of Computer and System Sciences **8**, 272 (1974).

[60] B. S. Baker and E. G. Coffman, Jr, *A tight asymptotic bound for next-fit-decreasing bin-packing,* SIAM Journal on Algebraic Discrete Methods **2**, 147 (1981).

[61] D. S. Johnson, A. Demers, J. D. Ullman, M. R. Garey, and R. L. Graham, *Worst-case performance bounds for simple one-dimensional packing algorithms,* SIAM Journal on Computing **3**, 299 (1974).

[62] B. S. Baker, *A new proof for the first-fit decreasing bin-packing algorithm,* Journal of Algorithms **6**, 49 (1985).

[63] M. Yue, *A simple proof of the inequality ffd (l)≤ 11/9 opt (l)+ 1,∀l for the ffd bin-packing algorithm,* Acta mathematicae applicatae sinica **7**, 321 (1991).

[64] D. S. Johnson and M. R. Garey, *A 7160 theorem for bin packing,* Journal of Complexity **1**, 65 (1985).

[65] J. P. Vielma, *Mixed integer linear programming formulation techniques,* Siam Review **57**, 3 (2015).

[66] J. T. Linderoth and T. K. Ralphs, *Noncommercial software for mixed-integer linear programming,* in *Integer Programming* (CRC Press, 2005) pp. 269–320.

[67] J. T. Linderoth and A. Lodi, *Milp software,* Wiley encyclopedia of operations research and management science (2010).

[68] S. Perez-Canto and J. C. Rubio-Romero, *A model for the preventive maintenance scheduling of power plants including wind farms,* Reliability Engineering & System Safety **119**, 67 (2013).

[69] G. Chen, W. He, L. C. Leung, T. Lan, and Y. Han, *Assigning licenced technicians to maintenance tasks at aircraft maintenance base: a bi-objective approach and a chinese airline application,* International Journal of Production Research **55**, 5550 (2017).

[70] L. Davis, *Handbook of genetic algorithms,* Van Nostrad Reinhold, New York (1991).

[71] S. Malik and S. Wadhwa, *Preventing premature convergence in genetic algorithm using dgca and elitist technique,* Int J Adv Res Comput Sci Soft Eng **4** (2014).

[72] H. Pirim, E. Bayraktar, and B. Eksioglu, *Tabu search: a comparative study,* in *Tabu Search* (InTech, 2008).

[73] J. F. Gonçalves, J. J. Mendes, and M. G. Resende, *A genetic algorithm for the resource constrained multi-project scheduling problem,* European Journal of Operational Research **189**, 1171 (2008).

[74] H. Zhaodong, C. Wenbing, X. Yiyong, and L. Rui, *Optimizing human resources allocation on aircraft maintenance with predefined sequence,* in *Logistics Systems and Intelligent Management, 2010 International Conference on*, Vol. 2 (IEEE, 2010) pp. 1018–1022.

[75] P. Kulkarni and P. Joshi, *Artificial intelligence: building intelligent systems* (PHI Learning Pvt. Ltd., 2015).

[76] D. Ambrosino, D. Anghinolfi, M. Paolucci, and A. Sciomachen, *An experimental comparison of different heuristics for the master bay plan problem,* in *International Symposium on Experimental Algorithms* (Springer, 2010) pp. 314–325.

[77] J. F. í lvarez Serrano, *A heuristic for vessel planning in a reach stacker terminal,* Journal of Maritime Research **3**, 3 (2006).

[78] V. Selvi and D. R. Umarani, *Comparative analysis of ant colony and particle swarm optimization techniques,* International Journal of Computer Applications (0975–8887) **5** (2010).

[79] S. Khalouli, R. Benmansour, and S. Hanafi, *An ant colony algorithm based on opportunities for scheduling the preventive railway maintenance,* in *Control, Decision and Information Technologies (CoDIT), 2016 International Conference on* (IEEE, 2016) pp. 594–599.

[80] S. Rahim, S. A. Khan, N. Javaid, N. Shaheen, Z. Iqbal, and G. Rehman, *Towards multiple knapsack problem approach for home energy management in smart grid,* in *Network-Based Information Systems (NBiS), 2015 18th International Conference on* (IEEE, 2015) pp. 48–52.

[81] W. Chen, Y.-j. Shi, H.-f. Teng, X.-p. Lan, and L.-c. Hu, *An efficient hybrid algorithm for resource-constrained project scheduling,* Information Sciences **180**, 1031 (2010).

[82] M. T. Jonsson, *Power plant maintenance scheduling using dependency structure matrix and evolutionary optimization,* in *Proceedings of the World Congress on Engineering and Computer Science*, Vol. 2 (2015).

[83] K. Deb, *Multi-objective optimization using evolutionary algorithms*, Vol. 16 (John Wiley & Sons, 2001).

[84] D. Fouskakis and D. Draper, *Stochastic optimization: a review,* International Statistical Review **70**, 315 (2002).

[85] F. Busetti, *Simulated annealing overview,* published on the World Wide Web URL www.geocities.com/francorbusetti/saweb.pdf [Online; accessed 23-June-2018] (2003).

[86] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, *A simulated annealing-based multiobjective optimization algorithm: Amosa,* IEEE transactions on evolutionary computation **12**, 269 (2008).

[87] F. Rossi, P. Van Beek, and T. Walsh, *Handbook of constraint programming* (Elsevier, 2006).

[88] S. I. Gass, *A process for determining priorities and weights for large-scale linear goal programmes,* Journal of the Operational Research Society **37**, 779 (1986).

[89] M. Larbani and B. Aouni, *On the pareto optimality in goal programming,* in *ASAC*, Vol. 28 (2007).

[90] Q. Deng, *A practical dynamic programming based methodology for aircraft maintenance check scheduling optimization (submitted),* (2019).

[91] G. J. Woeginger, *Exact algorithms for np-hard problems: A survey,* in *Combinatorial Optimization—Eureka, You Shrink!* (Springer, 2003) pp. 185–207.

[92] M. Reopel, *Smarter mro-5 strategies for increasing speed, improving reliability, and reducing costs—all at the same time,* (2012).