

Tactical and Operational Planning of Scheduled Maintenance for Per-Seat, On-Demand Air Transportation

Gizem Keysan, George L. Nemhauser, and Martin W.P. Savelsbergh

May 27, 2009

1 Introduction

Advances in aviation technology including the development of relatively cheap, very light jets and the possibility of free-flight have led to the realization of a per-seat, on-demand (PSOD) air transportation business that operates without a published flight schedule. In this business model, travelers call a few days in advance to request transportation by providing their origin airport, destination airport, an earliest departure time from the origin and a latest arrival time at the destination. An online scheduling algorithm quickly determines whether a request can be accommodated. However, the flight schedules for a given day are not finalized until the night before when itineraries (sets of consecutive flight legs to be flown by a single jet) are constructed to accommodate all accepted requests using an off-line scheduling algorithm ([6, 7]). Thus, on any given day only the itineraries for that day are known with certainty.

An important aspect of daily operations for PSOD air transportation is managing the scheduled maintenance of the jets in the fleet. Scheduled maintenance is mandated by the Federal Aviation Administration (FAA) to ensure safety and has to be done periodically. Unscheduled maintenance due to breakdowns or other unforeseen events is not included in this category. There are several types of scheduled maintenance. Some of these take a short amount of time and can be done at any airport. Others are more time consuming and are required to be done at a maintenance facility. In this paper, we consider types of scheduled maintenance that require a visit to a maintenance facility and have to be performed frequently enough to pose planning challenges.

The types of scheduled maintenance considered here are driven by different attributes of the fleet such as accumulated flying hours and number of take-offs and landings since the last maintenance. For example, certain checks might have to be done every 300 hours while others have to take place after every 100 take-offs and landings. Since the analysis is the same for the different attributes, we present the methodology in terms of accumulated flying hours.

Due to physical space limitations or the availability of labor, there is an upper bound on the number of jets that can be maintained on a given day. This is referred to as the maintenance capacity. The tactical level decision making for scheduled maintenance planning determines the capacity at the maintenance facility. As jets are introduced into the fleet gradually over time to accommodate business growth, the maintenance capacity needs to increase over time as well. Thus, the tactical decisions related to scheduled maintenance focus on when and how much to increase the maintenance capacity. We model these decisions using an integer program. To enhance our

ability to solve realistic instances, we develop valid inequalities that strengthen the formulation. Furthermore, we develop an optimization-based local search to find good solutions quickly.

Operational level planning is concerned with assigning itineraries to jets and determining the specific jets that have to go to maintenance on a daily basis given a certain maintenance capacity. We develop a two-phase approach. In the first phase, a multi-commodity network flow model is used to assign itineraries to critical jets, i.e. jets with high accumulated flying hours, and to decide which of these critical jets to send to maintenance. In the second phase, an integer programming model is used to assign itineraries to non-critical jets in such a way that the number of critical jets on subsequent days does not vary too much.

There is a strong interaction between the operational level maintenance decisions and flight scheduling. Maintenance decisions affect flight scheduling as jets that undergo maintenance cannot be used to transport passengers. Conversely, flight scheduling impacts maintenance decisions as it may be necessary to revise preferred maintenance decisions if the set of accepted transportation requests can no longer be accommodated when the preferred maintenance decisions are implemented. In order to capture this interaction, we develop a framework in which operational maintenance planning and daily flight scheduling receive feedback from each other. The framework is tested in a simulation environment where travel requests are generated using an agent-based model, and flight scheduling and operational maintenance planning are performed in real-time.

Although motivated by the operations of PSOD air transportation, the models and solution approaches introduced in this paper can be used more generally for dynamic scheduling problems in which resources need to undergo periodic maintenance and the arrival and the length of the tasks that need to be performed by the resources are non-deterministic.

The remainder of the paper is organized as follows. In Section 2, we formally introduce the scheduled maintenance planning setting considered. In Section 3, we briefly describe related work. Tactical and operational level scheduled maintenance planning problems are discussed in Section 4 and Section 5, respectively. We describe the integration of operational maintenance planning with flight scheduling in Section 6. Finally, in Section 7, we present a case study based on the operations of DayJet Corporation, a former startup PSOD air transportation provider, in which we test all our solution approaches including the optimization-based local search for tactical capacity planning, and the integrated framework of operational maintenance planning with flight scheduling.

2 Problem Description

Ideally, a jet should be maintained after accumulating a target number of flying hours (H) since its last maintenance. However, in reality it is hard to achieve this in a dynamic environment where future itineraries are unknown and maintenance capacity limits the number of jets that can be maintained on a given day. Thus, there is an allowance on both sides of H , i.e.

- (R1) A jet should accumulate at least $H^{min} = H - w$ hours of flying time before its next maintenance, and
- (R2) A jet can accumulate at most $H^{max} = H + w$ hours of flying time before its next maintenance.

Over the lifetime of a jet, using only these two rules might not provide a maintenance schedule in accordance with the original intentions. If a jet is maintained always after accumulating H^{max} (H^{min}) hours, then in the long run it is maintained less (more) frequently than originally intended. In other words, if a jet is maintained early in one interval, we would like to maintain it later in the next interval to ensure that on average it accumulates close to the target number of flying hours before being maintained. Thus, we introduce another rule to determine the timing of maintenance activities over the lifetime of a jet:

- (R3) The n^{th} maintenance can only be done when a jet accumulates flying hours in the interval $[nH - w .. nH + w]$ (the integer interval between $nH - w$ and $nH + w$).

For example, consider the case where $H = 300$ and $w = 30$. The intervals of accumulated flying hours in which a jet can be maintained are shown in Figure 1. The first maintenance can be done when the jet accumulates flying hours in the interval $[270 .. 330]$. Suppose it is maintained when it accumulates 280 hours. Considering the minimum and maximum flying hours to be accumulated before maintenance, the next maintenance can be done when the jet accumulates flying hours between $280 + 270 = 550$ and $280 + 330 = 610$. However, that maintenance should also fall into the interval $[570 .. 630]$. Thus, the next maintenance for this jet can only be done when it accumulates flying hours in the interval $[570 .. 610]$ and the shaded area within the second maintenance interval as shown in Figure 1 becomes unavailable.

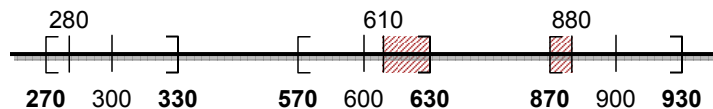


Figure 1: Allowable maintenance intervals for a jet

3 Related Work

Scheduled maintenance planning in which maintenance has to be performed at regular intervals is a periodic scheduling problem. Periodic scheduling is a well researched area with a broad range of applications. These include scheduling periodic real-time tasks on computer processors (Dhall and Liu [5]), data dissemination in teletext and wireless systems via broadcast disks (Kenyon et al. [11]), multi-item replenishment of stock (Anily et al. [1], Bar-Noy et al. [3]) and so on. However, most of the literature is on the operational planning problem for a given capacity while there are only a few studies that address the capacity planning problem. Furthermore, to the best of our knowledge, there is no study in the literature that addresses capacity planning for periodic scheduling where the number of tasks to be scheduled increases over time.

Starting with the early work of Wagner et al. ([15]), several studies address the operational scheduled maintenance planning problem. Some of these studies, such as Anily et al. ([1, 2]), Bar-Noy et al. ([3]), Grigoriev et al. ([9]) and Kenyon et al. ([11]) address a problem in which the

operating cost of a machine increases with time since its last maintenance. The aim is to minimize the long-run average cost per time. This problem differs from ours as there is no explicit time until the next maintenance.

Scheduled maintenance planning problems similar to ours have been studied in Mattila and Virtanen ([13]) for scheduling the periodic maintenance of a fleet of fighter aircraft, in Haghani and Shafahi ([10]) for scheduling bus maintenance activities, in Deris et al. ([4]) for ship maintenance scheduling, and in Kralj and Petrovic ([12]) for periodic maintenance scheduling of thermal units in electric power systems. None of these studies try to accumulate on average a target amount for the attributes that determine the timing of the maintenance as our problem does by considering maintenance rule (R3). Furthermore, the problems addressed in these studies do not have an additional assignment component to determine the daily use of vehicles/machines as it is constant and known. However, in our problem the flying time of the itineraries is not constant and thus, it is necessary to model the decisions pertaining to the assignment of itineraries to specific jets.

Gopalan and Talluri ([8]) review models and solution approaches for maintenance planning in schedule operated airlines. These models and solution approaches are quite different from ours. First, maintenance capacity is never considered to be a binding constraint in these studies. However, we try to allocate the lowest possible maintenance capacity and thus, capacity is a tight constraint most of the time. Furthermore, in schedule operated airline operations, the timing between successive maintenance activities is usually set to be 3 or 4 days and the solution approaches are geared towards finding a feasible routing into a maintenance facility within these days. These solution approaches are not applicable to our problem due to the uncertainty of the itineraries. Sriram and Haghani ([14]) presents a formulation based on accumulated flying hours for schedule operated airlines. However, they do not develop a solution procedure for solving the problem with this formulation as an exact solution cannot be obtained in reasonable computation time due to the large size of the problem.

4 Tactical Planning for Maintenance Capacity

Tactical planning for scheduled maintenance involves determining the daily capacities at the maintenance facility for a given planning horizon during which jets are introduced gradually into the fleet at specified points in time. The aim of tactical capacity planning is to ensure that sufficient maintenance capacity is available to perform the required scheduled maintenance for the jets, i.e. according to rules (R1) - (R3), in an environment where jets accumulate a number of flying hours on each day. At the tactical level planning, we assume that the jets accumulate the historical average number of daily flying hours, denoted by f , on each day of the planning horizon. This is a reasonable assumption, because at the operational level, we can assign itineraries to jets on each day so as to ensure that the true average number of flying hours per day is close to f .

The workload at the maintenance facility increases over time as new jets are introduced into the fleet. Since capacity installments are costly, any additional capacity installed to cope with increasing workload is never discarded and thus, capacity is monotone non-decreasing over time. Each unit of maintenance capacity also has a corresponding cost associated with the maintenance personnel. Thus, the objective of tactical maintenance capacity planning is to achieve the lowest possible total capacity over the planning horizon while ensuring that jets can be scheduled for maintenance in a timely fashion.

Before presenting the integer programming formulation for the General Tactical Capacity Planning Problem (GTCP), we introduce some notation. Let \mathcal{T} be the planning horizon and \mathcal{J} be the set of all jets introduced over the planning horizon. As mentioned before, jets are introduced gradually over time. Let d_j denote the day jet j is introduced into the fleet. Note that when jets are introduced into the fleet, they start with 0 accumulated flying hours. Let $\left\lceil \frac{H^{min}}{f} \right\rceil = n^{min}$ and $\left\lfloor \frac{H^{max}}{f} \right\rfloor = n^{max}$. Then, the set of days jet j can be maintained for the first time, denoted by \mathcal{M}_j^F , is $\{d_j + n^{min} - 1, \dots, d_j + n^{max} - 1\}$. Let \mathcal{M}_{jt}^N denote the set of days jet j can be maintained next given that it is maintained on day t , then $\mathcal{M}_{jt}^N = \{t + n^{min}, \dots, t + n^{max}\} \cap \{d_j - 1 + \left\lceil \frac{n'H - w}{f} \right\rceil, \dots, d_j - 1 + \left\lfloor \frac{n'H + w}{f} \right\rfloor\}$ where $n' = \min\{n \in \mathbb{Z}^+ : nH > (t - d_j + 1)f + H^{min}\}$. As can be seen, \mathcal{M}_{jt}^N is the intersection of two intervals where the first interval is obtained by considering maintenance rules (R1) and (R2), and the second interval is obtained by considering maintenance rule (R3).

There are two types of decision variables in the formulation. The binary variable x_{st}^j equals 1 if jet j is maintained on day s and next on day t , and 0 otherwise. The integer variable cap_t represents the capacity on day t . The integer programming formulation for GTCP is:

$$(GTCP) : \min z = \sum_{t \in \mathcal{T}} cap_t \quad (1)$$

s.t.

$$\sum_{t \in \mathcal{M}_j^F} x_{d_j-1,t}^j = 1 \quad \forall j \in \mathcal{J} \quad (2)$$

$$\sum_{s \in \mathcal{T} : t \in \mathcal{M}_{js}^N} x_{st}^j = \sum_{s \in \mathcal{M}_{jt}^N} x_{ts}^j \quad \forall j \in \mathcal{J}, \forall t \in \mathcal{T} \quad (3)$$

$$\sum_{j \in \mathcal{J}} \sum_{s \in \mathcal{T} : t \in \mathcal{M}_{js}^N} x_{st}^j \leq cap_t \quad \forall t \in \mathcal{T} \quad (4)$$

$$cap_t \leq cap_{t+1} \quad \forall t \in \mathcal{T} \quad (5)$$

$$x_{st}^j \in \{0, 1\} \quad \forall j \in \mathcal{J}, \forall s \in \mathcal{T}, \forall t \in \mathcal{M}_{js}^N \quad (6)$$

$$cap_t \geq 0, \text{ integer} \quad \forall t \in \mathcal{T}. \quad (7)$$

The objective is to minimize the total capacity over the planning horizon. Constraints (2) ensure that each jet is maintained once in its first maintenance interval. The periodic maintenance constraints are represented by constraints (3). Constraints (4) ensure that the number of jets maintained on each day is less than or equal to the capacity on that day. Finally, (5) are the monotonicity constraints on the capacity.

4.1 History-Independent Tactical Capacity Planning Problem

GTCP considers maintenance rules (R1) - (R3). To analyze the increase in the capacity due to (R3), which is not commonly used in applications, we consider a simpler model that is concerned with only (R1) and (R2). We call this model History-Independent Tactical Capacity Planning (HTCP).

Let \mathcal{M}_t^N denote the set of days on which the next maintenance for any jet can be done given that it is maintained on day t . $\mathcal{M}_t^N = \{t + n^{\min}, \dots, t + n^{\max}\}$ since the timing between successive maintenance activities is determined only by H^{\min} and H^{\max} . Let \mathcal{J}_t denote the set of jets introduced on day t . Variable z_{st} represents the number of jets that are maintained on day s and next on day t . HTCP is:

$$(HTCP) : \min z = \sum_{t \in \mathcal{T}} cap_t \quad (8)$$

s.t.

$$\sum_{s \in \mathcal{T} : t \in \mathcal{M}_s^N} z_{st} + |\mathcal{J}_{t+1}| = \sum_{s \in \mathcal{M}_t^N} z_{ts} \quad \forall t \in \mathcal{T} \cup \{0\} \quad (9)$$

$$\sum_{s \in \mathcal{T} : t \in \mathcal{M}_s^N} z_{st} \leq cap_t \quad \forall t \in \mathcal{T} \quad (10)$$

$$cap_t \leq cap_{t+1} \quad \forall t \in \mathcal{T} \quad (11)$$

$$z_{st} \geq 0, \text{ integer} \quad \forall s \in \mathcal{T}, \forall t \in \mathcal{M}_s^N \quad (12)$$

$$cap_t \geq 0, \text{ integer} \quad \forall t \in \mathcal{T}. \quad (13)$$

The periodic maintenance constraints are represented by constraints (9) and the maintenance capacity constraints are represented by constraints (10). All other constraints and the objective function are the same as the ones in GTCP. Note that this formulation considers the jets maintained on a given day in an aggregate fashion and does not model maintenance decisions for individual jets.

It is easy to see that a feasible solution for GTCP can be aggregated into a feasible solution for HTCP by summing the values of the maintenance decision variables x_{st}^j over the jets and keeping the values of the capacity decision variables the same. Furthermore, note that HTCP is an integer flow formulation with side constraints where nodes correspond to days in the planning horizon and the variable z_{st} corresponds to the flow on arc (s, t) . Thus, any feasible integer flow can be decomposed into paths corresponding to the flow of individual jets and these paths represent the maintenance schedule of the jets over the planning horizon. The decomposed solution over the paths together with the corresponding values of the capacity decision variables obtained while solving HTCP constitute a feasible solution for GTCP.

Proposition 1. *HTCP is equivalent to GTCP when rule (R3) is omitted.*

4.2 Solution Approach

An integral part of the solution approach is strengthening the formulations by including additional valid inequalities. The jets that have been introduced earlier than day t are referred to as *active jets* on that day and the set of active jets on day t is denoted by \mathcal{J}_t^A , i.e. $\mathcal{J}_t^A = \{j : d_j \leq t\}$. Let $\alpha_t = \lfloor \frac{|\mathcal{J}_t^A|}{n^{\max}} \rfloor$, and $\beta_t = |\mathcal{J}_t^A| \bmod n^{\max}$. Note that n^{\max} is the largest number of days a jet can go without maintenance for both GTCP and HTCP.

Proposition 2. For $t \in \mathcal{T}$,

$$\sum_{t'=t+n^{max}-\beta_t}^{t+n^{max}-1} cap_{t'} \geq \beta_t(\alpha_t + 1) \quad (14)$$

is a valid inequality for GTCP and HTCP. That is, for each day $t \in \mathcal{T}$, total capacity on the last β_t days of the interval $\{t, \dots, t + n^{max} - 1\}$ should be at least $\beta_t(\alpha_t + 1)$.

Proof. Suppose not. Then, $cap_{t+n^{max}-\beta_t} \leq \alpha_t$. Otherwise, we would have $cap_{t'} \geq \alpha_t + 1$, $\forall t' \geq t + n^{max} - \beta_t$ and (14) would hold.

Since $cap_{t+n^{max}-\beta_t} \leq \alpha_t$, $cap_{t'} \leq \alpha_t \quad \forall t' \leq t + n^{max} - \beta_t$ from the monotone non-decreasing property. Thus,

$$\sum_{t'=t}^{t+n^{max}-\beta_t-1} cap_{t'} \leq \alpha_t(n^{max} - \beta_t)$$

which implies

$$\sum_{t'=t}^{t+n^{max}-1} cap_{t'} < \beta_t(\alpha_t + 1) + \alpha_t(n^{max} - \beta_t) = \alpha_t n^{max} + \beta_t = |\mathcal{J}_t^A|.$$

Since the total capacity in the next n^{max} days starting from day t is less than the number of active jets on day t , at least one of these jets would not be able to be maintained without exceeding its maximum interval. Thus, we arrive at a contradiction and conclude that (14) is a valid inequality. \square

From now on we assume that GTCP and HTCP refer to the formulations in which inequalities (14) are added. HTCP is solved routinely by CPLEX. The relatively small size of this problem and the strengthening in the LP relaxation provided by the valid inequalities make it possible to solve it in seconds for realistic instances. For GTCP, where we model each jet separately in order to consider their maintenance histories, it is generally not possible to find a good solution in a reasonable amount of time even with the added valid inequalities. Finding good solutions in a reasonable amount of time even for planning purposes of this type is necessary to be able to conduct sensitivity analysis to study the impact of changing problem parameters. For example, we need to understand the effect of the fleet introduction schedule on the capacity to negotiate the best such schedule. Thus, we introduce an optimization-based local search algorithm to solve GTCP.

The optimization-based local search algorithm starts with an initial solution constructed trivially as follows. Suppose all maintenance activities are performed as early as possible considering maintenance rules (R1) - (R3). That is, $x_{d_j-1,t}^j = 1$ if t is the first day in M_j^F and given that jet j is maintained on day s , $x_{st}^j = 1$ if t is the first day in M_{js}^N . Furthermore, $cap_t = cap_{t-1}$ if the total number of jets maintained on day t is less than or equal to the number of jets maintained on day $t - 1$. Otherwise, cap_t is equal to the number of jets maintained on day t .

A neighborhood around a solution is constructed by fixing the maintenance decision variables corresponding to all but a subset of jets to their values in this solution. Let $\bar{\mathcal{S}} = \{\bar{x}_{st}^j, \bar{cap}_t, \forall j \in \mathcal{J}, \forall s, t \in \mathcal{T}\}$ be a feasible solution for GTCP and $\mathcal{J}_{\bar{\mathcal{S}}}$ be the subset of jets for which the

maintenance decisions are not fixed. The corresponding neighborhood, denoted by $N_{\bar{\mathcal{J}}, \mathcal{J}_{\bar{\mathcal{J}}}}$, is $\{x_{st}^j, cap_t : x_{st}^j, cap_t \text{ satisfy the constraints of GTCP, } x_{st}^j = \bar{x}_{st}^j, \forall j \in \mathcal{J} \setminus \mathcal{J}_{\bar{\mathcal{J}}}\}$.

$\mathcal{J}_{\bar{\mathcal{J}}}$ is chosen such that the optimization problem for the corresponding neighborhood is likely to lead to improvement and is efficiently solvable. The first type of neighborhood, referred to as the *primary neighborhood*, is obtained by choosing $\mathcal{J}_{\bar{\mathcal{J}}}$ among the set of jets that are maintained on the days where capacity increases, denoted by \mathcal{J}^{inc} . The aim of choosing $\mathcal{J}_{\bar{\mathcal{J}}}$ this way is to delay the days where capacity increases by changing the decisions for the jets that are maintained on these days, and thus decreasing the objective function value. In order to solve the corresponding optimization problem in the neighborhood efficiently, only a specified number, denoted by r_1 , of such jets are randomly chosen to be included in $\mathcal{J}_{\bar{\mathcal{J}}}$.

Our experiments indicated that if an improving solution cannot be obtained with the chosen primary neighborhood, exploring different neighborhoods of this type by selecting different subsets from \mathcal{J}^{inc} did not lead to an improvement for many iterations. Thus, a *secondary neighborhood*, is constructed to increase the likelihood of obtaining an improving solution in this situation. The secondary neighborhood is obtained by choosing $\mathcal{J}_{\bar{\mathcal{J}}}$ to include jets that are not in \mathcal{J}^{inc} as well as jets that are in \mathcal{J}^{inc} . Specifically, r_1 jets are chosen randomly among the ones in \mathcal{J}^{inc} and r_2 jets are chosen randomly among the ones that are not in \mathcal{J}^{inc} .

The optimization-based local search algorithm starts with a chosen primary neighborhood. The neighborhoods are explored by solving the corresponding optimization problems using CPLEX with an upper bound on the solution time for each neighborhood. If at any iteration, an improving solution cannot be obtained within the chosen primary neighborhood, the local search starts exploring secondary neighborhoods until an improving solution is obtained. After obtaining an improving solution, it goes back to exploring primary neighborhoods. The search stops when a specified time limit is reached.

4.3 Computational Results

We report the results of our solution approach for instances that represent 10 practical scenarios for the growth of the fleet over a two year planning horizon. The aim of the computational study is to analyze both the impact of the frequency with which jets are introduced into the fleet as well as the total number of jets required to meet customer demand. We consider two classes of instances: The first class assumes that at the end of the second year we have as many jets as the number of business days in the planning horizon, i.e. on average one new jet arriving per day and 480 jets at the end of the planning horizon. The second class assumes that projected customer demand warrants approximately half the number of jets over the planning horizon, i.e. on average one new jet arriving every two days and 288 jets at the end of the planning horizon. Finally, within each of these two classes we test five possibilities for the frequency with which jets arrive: every 1 – 2 days, 1 week, 2 weeks, 1 month, and 2 months. The arrival of new jets has a delayed impact on maintenance capacity as jets have to accumulate a minimum number of flying hours before maintenance. In order to account for the delay for jets introduced at the end of the second year, an additional six month period is included after two years.

Scheduled maintenance is required every 300 hours on average. An allowance of 10% is given on both sides of 300. Also, maintenance rule (R3) translates to maintenance having to fall into $300n \pm 30$ hours intervals where $n \in \mathbb{Z}^+$. Finally, the average flying time f is 10 hours. For

all computational results in the paper, the code is written in C++ and the IP models are built and solved with Concert 2.1 in ILOG CPLEX 9.1 using 2.4 GHz AMD 250 processors with 4 GB of RAM. Initial experiments indicated that the root node solved fastest when using the Barrier algorithm. We thus report results where the Barrier algorithm with crossovers is used to solve the root node and default CPLEX is used thereafter.

During the implementation of the optimization-based local search algorithm, we experimented with different values for the sizes of the subset of jets for which we optimize the maintenance decisions in the primary and secondary neighborhoods. The results of these experiments indicated that the following configuration gives the best quality solutions in a reasonable amount of time: primary neighborhoods including $r_1 = 20$ jets chosen from J^{inc} , the set of jets maintained on the days where capacity increases, and the secondary neighborhoods including $r_1 = 20$ jets chosen from J^{inc} and $r_2 = 20$ jets chosen among the ones that are not in J^{inc} . Furthermore, the time limit for exploring a neighborhood is set to be 120 seconds.

Figure 2 shows the progress of the optimization-based local search over time during the first 30 minutes in terms of the gap to the best lower bound obtained within 24 hours with CPLEX while Figure 3 shows the same during the time period between half an hour and two hours. The instances are named according to the number of jets they have at the end of the planning horizon and the time interval between jet arrivals. For example, $(480J, 1M)$ represents the instance in which a total of 480 jets are introduced where the time interval between jets arrivals is 1 month.

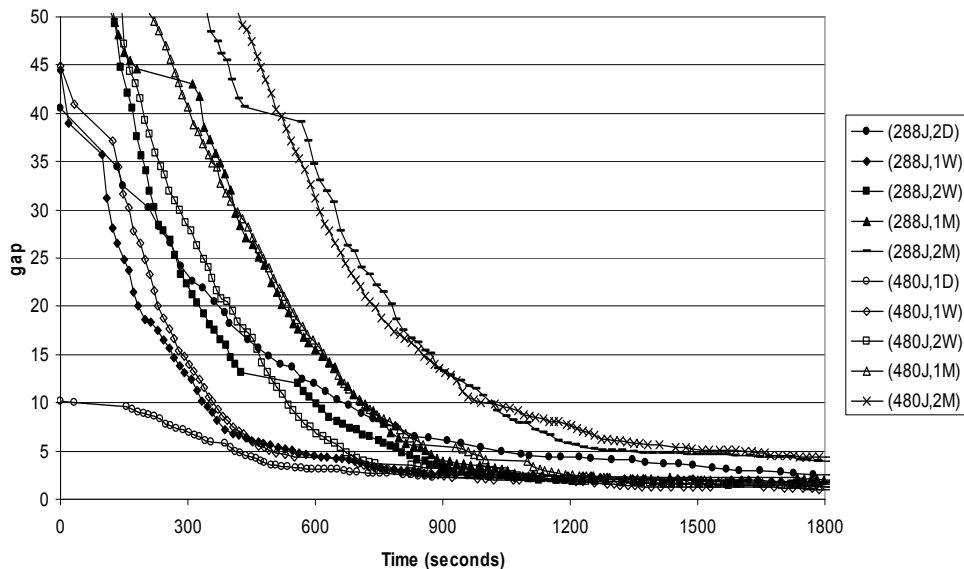


Figure 2: Optimization-based local search progress within half an hour

As can be seen from Figure 2, the optimization-based local search algorithm finds a solution with a gap of less than or very close to 5% within 10 minutes for half of the instances. Furthermore, after 30 minutes all instances have a gap of less than 5%. All instances are solved to around 1% of optimality and the rate of progress has become insignificant by the end of the two hours as seen in Figure 3.

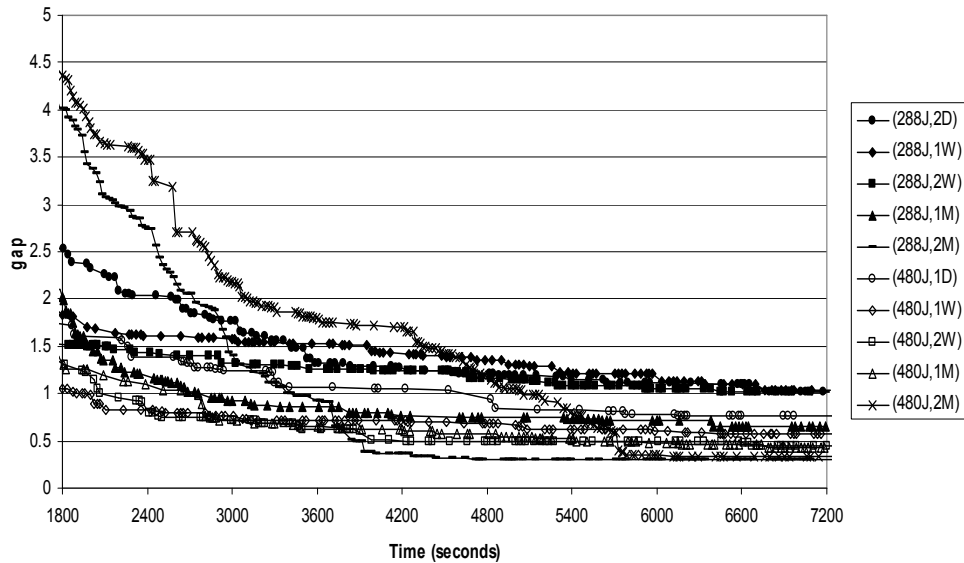


Figure 3: Optimization-based local search progress between half an hour and two hours

It is interesting to note that the optimization-based local search algorithm starts with a solution that has a very small gap, which is around 10%, only for the instance in which a total of 480 jets are introduced on average every day. On examining the total number of maintenances in the initial solution as a percentage of total capacity allocated, we note that 89% of the allocated capacity is already used which indicates that there is little room for improvement. For all other instances, only around 40 – 50% of the total allocated capacity is used in the initial solution. This implies that in these instances it is possible to decrease the total capacity by delaying the capacity increases while still complying with the maintenance rules.

We compare the results of our optimization-based local search algorithm to the results obtained by solving GTCP using CPLEX in Table 1. The first column corresponds to the instance names. The next two columns represent the best upper bound, denoted by *BEST UB*, and the best lower bound, denoted by *BEST LB*, obtained within 24 hours with CPLEX respectively. These runs are used as a benchmark to judge the quality of solutions obtained in shorter time. In the rest of the table, we present the objective values (*O.V.*) and the gap of these objective values to *BEST LB* (*gap*) within half an hour, one hour and two hours time limits for CPLEX and the optimization-based local search algorithm, respectively. Finally, the last column in Table 1 (*Optimal Without (R3)*) represents the optimal total capacity that would be allocated if maintenance rule (R3) was not considered.

As can be seen from Table 1, CPLEX found a feasible solution for none of the instances within half an hour whereas feasible solutions with on average 2% optimality gap are obtained with the optimization-based local search. Furthermore, less than half of the instances are solved to on average 4.68% and 2.22% of optimality with CPLEX within one hour and two hours, respectively while all instances are solved to on average 1% and 0.65% of optimality with optimization-based local search within one hour and two hours, respectively. These results clearly prove the benefits that

Table 1: Summary results for the tactical capacity planning problem

Instance	CPLEX						Optimization-based Local Search						Optimal Without (R3)		
	24 hours		30 min.		1 hour		2 hours		30 min.		1 hour			2 hours	
	Best UB	Best LB	O.V.	gap	O.V.	gap	O.V.	gap	O.V.	gap	O.V.	gap		O.V.	gap
(288J, 2D)	5075	5057.8	-	-	5394	6.23	5075	0.34	5189	2.52	5126	1.33	5110	1.02	4788
(288J, 1W)	5138	5076.8	-	-	5472	7.22	5402	6.02	5173	1.86	5156	1.53	5130	1.04	4810
(288J, 2W)	5273	5152	-	-	-	-	-	-	5232	1.51	5218	1.26	5205	1.02	4874
(288J, 1M)	5332	5322.8	-	-	-	-	5463	2.57	5433	1.99	5369	0.86	5358	0.66	5016
(288J, 2M)	6173	6171.8	-	-	6214	0.68	6173	0.02	6430	4.01	6230	0.93	6191	0.31	5531
(480J, 1D)	-	8335.4	-	-	-	-	-	-	8486	1.78	8425	1.07	8398	0.75	7710
(480J, 1W)	-	8394.2	-	-	-	-	-	-	8483	1.05	8455	0.72	8442	0.57	7799
(480J, 2W)	-	8547	-	-	-	-	-	-	8668	1.4	8601	0.63	8585	0.44	7947
(480J, 1M)	-	8746.2	-	-	-	-	-	-	8865	1.34	8804	0.66	8784	0.43	8224
(480J, 2M)	10106	10051	-	-	-	-	-	-	10513	4.39	10235	1.8	10084	0.33	9026

can be achieved with optimization-based local search, especially for the larger 480 jets instances as CPLEX finds a feasible solution for none of these instances within two hours. In addition, from the 24 hour CPLEX runs, we can further conclude that although most 288 jets instances end up with a better objective value than the objective value of the solutions found with the optimization-based local search algorithm in two hours, CPLEX found feasible solutions for almost none of the 480 jets instances.

For a fixed level of jets at the end of the planning horizon, it is expected that the total capacity would be lower for the instances in which a smaller number of jets are introduced more frequently since there would be more opportunities for spreading the maintenance over time. The results in Table 1 substantiate this expectation. However, the rate of change in total capacity diminishes as the introductions become more frequent since the length of the extra interval to spread the maintenance of jets gets smaller.

Finally, if we compare the solution values that are obtained with the optimization-based local search algorithm within two hours and the optimal total capacity that would be achieved if maintenance rule (R3) was not considered, we see that the total capacity increases by around 7% which is a significant investment to accommodate this rule. This additional investment can be considered as the hedge against potential costs of increased frequency of maintenance or breakdowns if this rule is not followed.

5 Operational Planning for Scheduled Maintenance

Operational planning for scheduled maintenance involves assigning itineraries to jets and determining the specific jets to be maintained on a daily basis subject to capacity limitations. Our main objective here is to ensure that jets accumulate flying hours as close as possible to the target H between successive maintenance activities. The possibility of choosing which itineraries to assign to jets with different accumulated flying hours gives us the opportunity to optimize our maintenance decisions. However, care must be taken to avoid situations in which the number of jets maintained on the current day is much less than the capacity whereas the number of jets in need of maintenance at some future point in time is likely to exceed the capacity.

5.1 Solution Approach

We employ a look-ahead approach which involves solving a k -day problem for the current day in order to capture the effect of current maintenance decisions on the future. It might be necessary to maintain some jets earlier if the capacity would be insufficient to accommodate all the jets requiring maintenance in the future. The look-ahead approach captures that by considering information about the next $k - 1$ days in addition to the current day. As the itineraries for a given day are constructed the night before, only the itineraries of the current day are known with certainty. For the other $k - 1$ days, it is assumed that each jet will accumulate the average flying hours f on each day. Thus, solving a k -day problem on day $t \in \mathcal{T}$ refers to making itinerary assignments to jets on day t , determining the jets to be maintained on each of the k days, and implementing these maintenance decisions for day t .

Maintenance rules (R1) and (R3) together determine a lower bound on the flying hours to be accumulated until the next maintenance for the jets. Thus, certain jets cannot be maintained during the k days starting from day t since their accumulated flying hours would not reach the lower bound. These jets are referred to as the *non-critical jets*. All other jets can potentially be maintained during the k days and are referred to as the *critical jets*. Exploiting the fact that non-critical jets have no impact on maintenance decisions over the k days, we decompose our problem and solve it in two phases. The first phase, which is referred to as the *maintenance decision phase*, determines the itinerary assignments to the critical jets on day t and the jets to be maintained on each of the k days. The second phase is concerned with the assignment of the remaining unassigned itineraries to the non-critical jets on day t .

Although non-critical jets play no part in maintenance decision making over the k days, assigning itineraries arbitrarily could in the long run impact the flexibility available to the maintenance decision phase to meet the target accumulated flying hours H . Even worse, it could lead to infeasibility with respect to maintenance capacity in the future. Consider the example given in Figure 4 where $k = 3$. Here we arrive at a situation in which although the maintenance schedule seems feasible for the next k days, the number of jets that are approximately $k + 1$ days away from maintenance well exceeds the maintenance capacity. As a result, some jets may have to be maintained earlier or later than their intended maintenance day resulting not only in poor usage of flying hours but also possibly increasing the future workload at the maintenance facility and leading to infeasibilities. To avoid such situations, our objective in assigning itineraries to non-critical jets is to ensure that the number of jets with varying levels of accumulated flying hours is as evenly distributed as possible. We call this the *smoothing phase*.

5.1.1 Maintenance Decision Phase

The Maintenance Decision Phase (MDP) is modeled as a multicommodity network flow problem on a time-expanded network denoted by $D = (\mathcal{N}, \mathcal{A})$ where \mathcal{N} is the set of nodes and \mathcal{A} is the set of arcs. Let \mathcal{T}_t denote the set of days MDP is solved for on day t and \mathcal{J}_t^C denote the set of critical jets on day t . Each node $(j, t', h) \in \mathcal{N}$, corresponds to a critical jet $j \in \mathcal{J}_t^C$ on a given day $t' \in \mathcal{T}_t$ with h accumulated flying hours since its last maintenance. Let \mathcal{I}_t denote the set of itineraries on day t and g_i denote the flying time of itinerary i .

Suppose critical jet j starts day t with \bar{h} accumulated flying hours. The part of the network corresponding to jet j is constructed as follows. First of all, a node (j, t, \bar{h}) is included in the

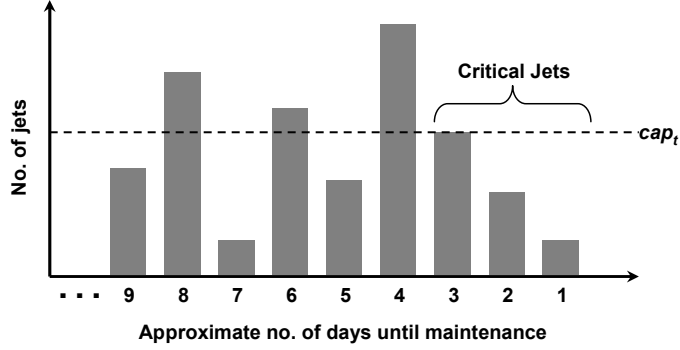


Figure 4: Impact of large fluctuations in the number of jets with different accumulated flying hours

network to represent the starting condition of jet j . Then, a node $(j, t', 0)$ is included for each $t' > t$ to represent the jet being maintained on day $t' - 1$.

For day t and for each $i \in \mathcal{I}_t$,

- a node $(j, t + 1, \bar{h} + g_i)$ is included in the network.
- an arc between nodes (j, t, \bar{h}) and $(j, t + 1, \bar{h} + g_i)$ is included in the network representing the assignment of itinerary i to jet j on day t without being sent to maintenance.
- an arc between nodes (j, t, \bar{h}) and $(j, t + 1, 0)$ is included in the network representing the assignment of itinerary i to jet j on day t after which the jet goes to maintenance (this arc is referred to as a_j^i).

Next, for day $t' > t$ and for each node (j, t', h') ,

- a node $(j, t' + 1, h' + f)$ is included in the network.
- an arc between nodes (j, t', h') and $(j, t' + 1, h' + f)$ is included in the network representing the assignment of an average length itinerary (with length f) to jet j on day t' without being sent to maintenance.
- an arc between nodes (j, t', h') and $(j, t' + 1, 0)$ is included in the network representing the assignment of an average length itinerary (with length f) to jet j on day t' after which the jet goes to maintenance.

Complying with the maintenance rules and not exceeding the capacity might not always be possible due to the uncertainty introduced into the problem by unknown itineraries. Thus, the network is constructed such that the lower bound on the flying hours to be accumulated is explicitly considered while the upper bound is relaxed. That is, we do not add arcs that violate the lower bound but we include arcs that violate the upper bound on the flying hours to be accumulated since the last maintenance.

The objective is to maintain jets with accumulated flying hours as close to target H as possible. That is, we would like to minimize the deviation of the accumulated flying hours from the target H when jets are maintained. However, it is also important to have a schedule in which the deviations from the target is approximately uniform among the fleet. To achieve this, it is necessary to use a nonlinear penalty function. Specifically, we set the penalty of being maintained with $H \pm q$ accumulated flying hours to be q^2 .

The arcs that represent the jet being maintained are referred to as the *maintenance arcs*. The cost of arc $a \in \mathcal{A}$ is denoted by c_a and is given by:

- $c_a = 0$ if a is not a maintenance arc and the accumulated flying hours of the jet after flying the itinerary associated with the arc do not exceed the upper limit;
- $c_a = q^2$ if a is a maintenance arc and the accumulated flying hours $H \pm q$ of the jet after flying the itinerary associated with the arc do not exceed the upper limit;
- $c_a = C$ if the accumulated flying hours of the jet after flying the itinerary associated with the arc do exceed the upper limit (where C is chosen large enough to ensure that this arc is never used by the jet if a feasible alternative is available).

Note that the costs considered so far are incurred when a jet is maintained or exceeds the upper limit on the accumulated flying hours. However, it is necessary to include a cost component to measure the effect of not maintaining a jet within \mathcal{T}_t as it will have to be maintained at some point in the near future. To this end, we consider an additional time period following \mathcal{T}_t , denoted by $\bar{\mathcal{T}}$, that is long enough to ensure that all critical jets will have to be maintained by the end of this time period in order not to exceed the upper limit on the accumulated flying hours. For each critical jet that is not maintained within \mathcal{T}_t , we want to find a day in $\bar{\mathcal{T}}$ on which it will be maintained and an approximate value for the penalty that will be incurred for this maintenance. In order to achieve this, we extend the time-expanded network D . For each day $\bar{t} \in \bar{\mathcal{T}}$, we include a node $u_{\bar{t}}$. For each node $(j, t + k, h)$ corresponding to a situation in which jet j has not been maintained within \mathcal{T}_t , we add arcs to nodes $u_{\bar{t}}$ for all $\bar{t} \in \bar{\mathcal{T}}$. Such an arc represents jet j being maintained on day \bar{t} and the cost of this arc is equal to the penalty that will be incurred if jet j is maintained after accumulating $h + f(\bar{t} - (t + k))$ hours.

An example of the part of network D corresponding to a single jet is shown in Figure 5, where dotted arcs represent the jet being maintained. The number of days in \mathcal{T}_t and $\bar{\mathcal{T}}$ is three. The jet starts day t with 275 hours accumulated since its last maintenance. It is assumed that H , H^{min} , and H^{max} are 300, 270, and 330 and that f is 8. Furthermore, there are two possible itineraries i_1 and i_2 for the jet on day t with flying time 6 and 10 hours, respectively.

Let \mathcal{A}^i represent the set of arcs associated with itinerary $i \in \mathcal{I}_t$ and $\mathcal{A}_{t'}^M$ represent the set of arcs representing a jet being maintained on day t' . Decision variable y_a corresponds to the jet flow on arc $a \in \mathcal{A}$. MDP can be written as the integer program:

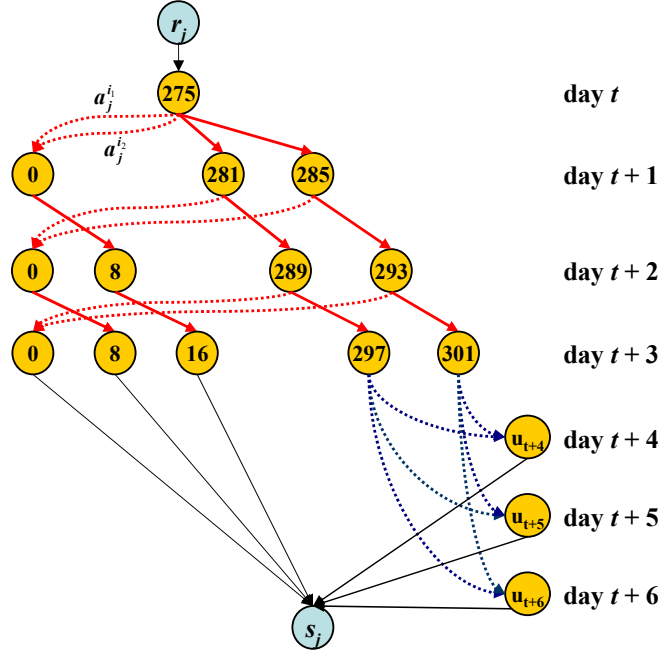


Figure 5: Part of the network for a single jet

$$(MDP) : \min z = \sum_{a \in \mathcal{A}} c_a y_a \quad (15)$$

s.t.

$$\sum_{a \in \delta^{out}(n)} y_a - \sum_{a \in \delta^{in}(n)} y_a = \begin{cases} 1 & \text{if } n = r_j \text{ for } j \in J_t^C; \\ -1 & \text{if } n = s_j \text{ for } j \in J_t^C; \forall n \in \mathcal{N} \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

$$\sum_{a \in \mathcal{A}^i} y_a \leq 1 \quad i \in \mathcal{I}_t \quad (17)$$

$$\sum_{a \in \mathcal{A}_t^M} y_a \leq cap_t \quad \forall t' \in \mathcal{I}_t \cup \bar{\mathcal{T}} \quad (18)$$

$$y_a \geq 0, \text{ integer} \quad \forall a \in \mathcal{A}. \quad (19)$$

The objective is to minimize the total cost of flow of jets on the arcs. Constraints (16) are the flow balance constraints for the critical jets. Constraints (17) ensure that each itinerary is assigned to at most one critical jet. Constraints (18) are the maintenance capacity constraints. Note that the maintenance capacities are parameters here as they already have been determined in the tactical level.

5.1.2 Smoothing phase

In the Smoothing Phase (SP), the itineraries of day t that were not assigned to a critical jet while solving MDP are assigned to non-critical jets with the objective of distributing the number of jets with varying levels of accumulated flying hours evenly. We divide the range of possible accumulated flying hours into buckets where the length of each bucket is equal to the average daily flying time f . Thus, a bucket also corresponds to a certain number of days until the next maintenance and on each day a jet typically moves from one bucket to the next, each time getting a day closer to maintenance. Having approximately the same number of jets in each bucket, in particular buckets of high accumulated flying hours (i.e. few days away from maintenance), should provide a smooth workflow into the maintenance facility giving MDP ample flexibility to ensure feasibility with respect to capacity.

Let B denote the set of accumulated hours buckets and n_b denote the number of days until the next maintenance corresponding to bucket $b \in B$. In order to have approximately the same number of jets in each bucket while considering the capacity at the maintenance facility, we penalize deviations of the number of jets in bucket b from cap_{t+n_b} , the capacity available on the day these jets will likely be maintained. Note that it might not be possible to distribute the number of jets evenly among the buckets of low accumulated flying hours right after the introduction of new jets. Furthermore, smoothing out an uneven distribution among the buckets of high accumulated flying hours is harder as these jets are only a few days away from maintenance. Thus, the penalty of deviating from the maintenance capacity gets larger as jets get closer to maintenance.

Let the parameter $h_{ijb} = 1$ if assigning itinerary i to jet j takes it to bucket $b \in B$ and 0 otherwise. Let p_b denote the penalty for the deviation of the number of jets in bucket b from cap_{t+n_b} . Finally, let the set of critical jets that are in bucket b be represented by J_b^C . There are two types of decision variables. The binary decision variable w_{ij} equals 1 if itinerary $i \in I_t^U$, where I_t^U represents the set of unassigned itineraries of day t , is assigned to non-critical jet $j \in J_t^N$, where J_t^N represents the set of non-critical jets on day t , and 0 otherwise. The variable D_b corresponds to the absolute deviation of the number of jets in bucket b from cap_{t+n_b} . The integer program solved in the smoothing phase is given as:

$$(SP) : \min z = \sum_{b \in B} p_b D_b \quad (20)$$

s.t.

$$\sum_{i \in I_t^U} w_{ij} = 1 \quad \forall j \in J_t^N \quad (21)$$

$$\sum_{j \in J_t^N} w_{ij} = 1 \quad \forall i \in I_t^U \quad (22)$$

$$D_b \geq |J_b^C| + \sum_{i \in I_t^U} \sum_{j \in J_t^N} h_{ijb} w_{ij} - cap_{t+n_b} \quad \forall b \in B \quad (23)$$

$$D_b \geq cap_{t+n_b} - |J_b^C| - \sum_{i \in I_t^U} \sum_{j \in J_t^N} h_{ijb} w_{ij} \quad \forall b \in B \quad (24)$$

$$w_{ij} \in \{0, 1\} \quad \forall i \in I_t^U, \forall j \in J_t^N \quad (25)$$

$$D_b \geq 0, \text{ integer} \quad \forall b \in B. \quad (26)$$

The objective is to minimize the total deviation of the number of jets in the accumulated hours buckets from maintenance capacity. Constraints (21) ensure that each non-critical jet is assigned exactly one itinerary while constraints (22) ensure that each unassigned itinerary of day t is assigned to exactly one non-critical jet. Finally, constraints (23) and (24) determine the deviation of the number of jets in bucket b from cap_{t+n_b} .

5.2 Computational Results

We report the results of our solution approach for the 10 instances introduced in Section 4.3. The computations are done over a rolling horizon where the problem for the current day t in the planning horizon is solved for k days and only the decisions made for day t are implemented. When we roll forward one day in the planning horizon, the problem is resolved for k days starting from day $t+1$ with the newly available information about the itineraries on day $t+1$.

The experimental settings used during the computations are as follows. We assume that all itineraries have a flight time equal to a random integer between 8 and 12 hours. If a jet is scheduled for maintenance on day t , its total flying time on days t and $t+1$ are incremented by the flying time to the maintenance facility and the flying time from the maintenance facility, respectively. Initially, the average flying time that is used for unknown itineraries in the future is set to be 10 hours. As we move further in the planning horizon, this average is updated with a moving average with exponential smoothing. The problem parameters related to maintenance are the same as before, i.e. H , H^{min} and H^{max} are equal to 300, 270 and 330, respectively. The maintenance capacities presented in Section 4.3 that were obtained with the optimization-based local search algorithm within 2 hours are used during the experiments. The cost of accumulating more than the upper bound on the accumulated flying hours before the next maintenance, C , is set to be larger than $30^2|J|$, which is the largest total penalty for feasible maintenance. Finally, p_b , which is used in the smoothing phase to penalize the deviations of the number of jets in accumulated flying hours bucket b from the capacity is set to be $2^{\left(\left\lceil \frac{H}{F} \right\rceil - n_b\right)}$ where n_b is the number of days until maintenance

for a jet in bucket b . After experimenting with different values for the look-ahead length k while solving MDP, $k = 10$ was found to be the best choice. Also, choosing $|\overline{T}| = 10$, which is the length of the additional time period considered while solving MDP to capture the cost of not maintaining a jet, is enough to ensure all critical jets will be maintained feasibly within the given time period.

Figure 6 shows the number of jets maintained on each day for instances $(288J, 2D)$ and $(288J, 2M)$. The area shown by the lighter color represents the maintenance capacity while the darker area to the front gives the number of jets maintained on each day. As can be seen from Figure 6, the number of jets maintained on each day closely follows the maintenance capacity for instance $(288J, 2D)$ while there is unused capacity during the first year for instance $(288J, 2M)$. This is expected since the maintenance capacity needs to be increased to higher levels earlier when jets are introduced in larger batches. Capacity usage rate is 99% and 86% for instances $(288J, 2D)$ and $(288J, 2M)$ respectively which is large enough to indicate a good match between the requirements of operational planning and the output of tactical planning.

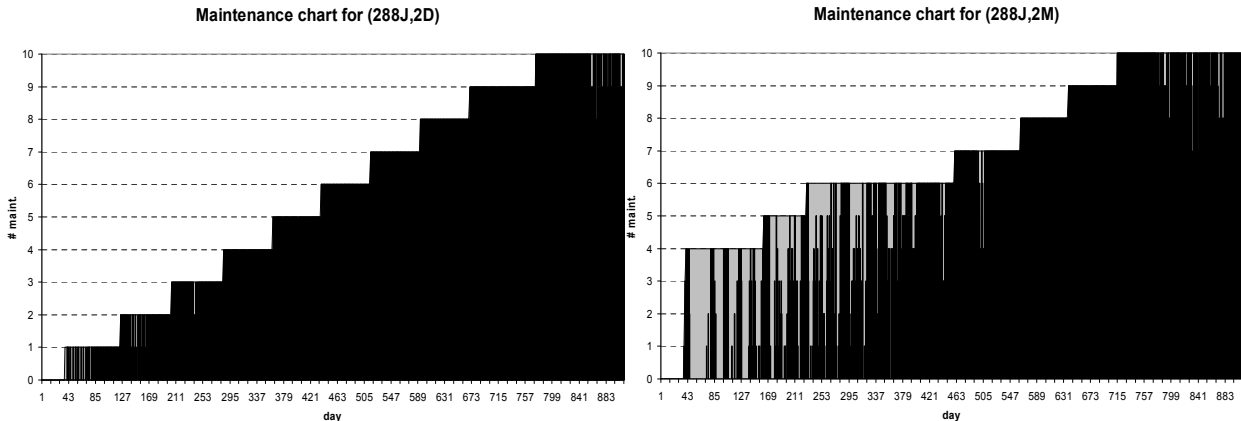


Figure 6: Maintenance charts for instances $(288J, 2D)$ and $(288J, 2M)$

Figure 7 shows the histograms of the accumulated flying hours between successive maintenance activities for the two instances. As can be seen from the figure, the majority of the time jets are scheduled for maintenance within the target accumulated flying hours. It can also be observed that the distribution of the deviations around the target is more uniform for instance $(288J, 2M)$ while most of the deviations for instance $(288J, 2D)$ are above the target. This result is due to the fact that maintenance capacity is very tight for instance $(288J, 2D)$ and thus, it is almost impossible to maintain some jets earlier using the unused available capacity when it is obvious that some other jets are likely to be maintained late.

Table 2 summarizes the results for all 10 instances. The first column shows the total number of times maintenance is performed whereas the second column represents the percentage of total capacity that is used for maintenance during the planning horizon. The third and fourth columns show the average and the standard deviation of the accumulated flying hours between maintenance activities respectively. The fifth column corresponds to the percentage of maintenance activities that violate the maintenance rules. Finally, the last column represents the average time it takes to solve the operational maintenance planning problem.

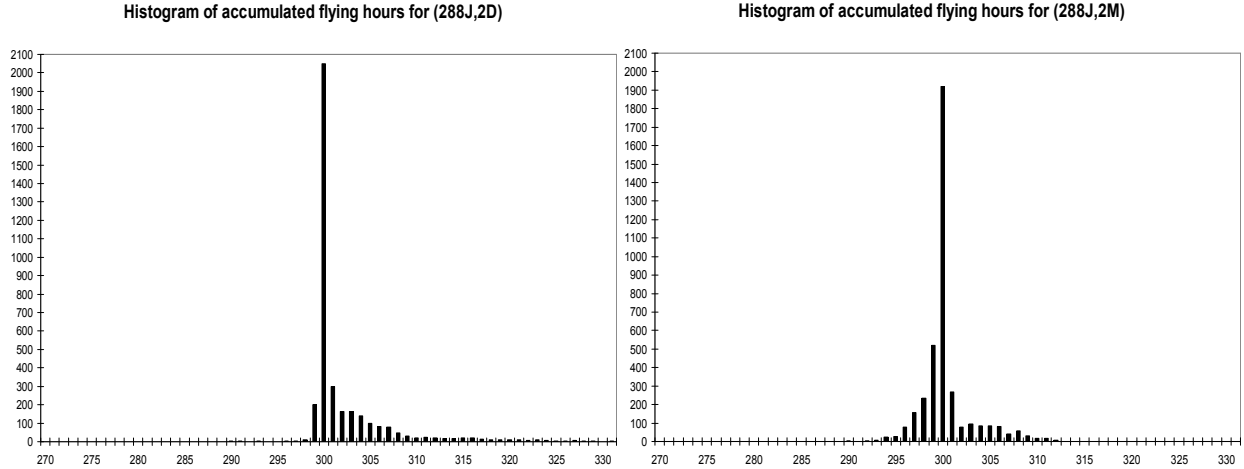


Figure 7: Histograms of accumulated flying hours for instances (288J,2D) and (288J,2M)

Table 2: Summary results for the operational maintenance planning problem

Instance	No. of maint.	% cap. used	Avg. acc. hrs.	Std. dev. of acc. hrs.	% infeasible maint.	Avg. time (sec)
(288J, 2D)	3618	98.97	302.09	5.11	0.53	5.14
(288J, 1W)	3634	98.91	302.05	5.03	0.33	5.43
(288J, 2W)	3661	98.23	301.87	4.75	0.11	5.5
(288J, 1M)	3725	97.08	301.12	3.44	0	5.87
(288J, 2M)	3841	86.68	300.34	2.7	0	5.93
(480J, 1D)	6000	99.92	301.87	3.44	0	11.72
(480J, 1W)	6037	99.83	301.82	3.37	0	11.78
(480J, 2W)	6086	99.07	301.68	3.36	0	11.84
(480J, 1M)	6178	98.28	301.03	3.15	0	11.94
(480J, 2M)	6373	88.34	300.56	2.41	0	12.02

From Table 2, it can be seen that capacity usage rate is above 97% for almost all the instances. As mentioned before, the capacity usage rate for instances in which jets are introduced with the largest batches and least frequently is lower since maintenance capacity has to be increased quickly to larger levels in order to accommodate the early maintenance requirements of the batches arriving together. The high capacity usage rate together with a rate of compliance with the rules above 99% indicate a good match between the output of tactical planning and the requirements of operational planning. It can also be observed that with the given capacity we can achieve our maintenance objectives since average accumulated flying hours between successive maintenance activities is very close to the target 300 hours for all instances while the standard deviation is at most approximately 5 hours. Furthermore, the average time it takes to solve the operational maintenance planning problem is quite small even for the large instances with 480 jets.

The average and the standard deviation of accumulated flying hours between successive maintenance activities improve as jets are introduced in larger batches and less frequently. This is due to the fact that capacity is increased earlier for these instances and with more capacity available, some jets can be maintained a little earlier to avoid maintaining other jets much later. Another advan-

tage of capacity increasing earlier for these instances is in the lower rate of maintenance activities violating the rules. However, these improvements are not large enough to justify the increase in the total number of maintenance activities resulting in larger costs. Thus, introducing jets into the fleet in smaller batches and more frequently seems to be better for operational planning purposes as well.

6 Integration with Flight Scheduling

There is a strong interaction between the operational level maintenance decisions and flight scheduling. Maintenance decisions affect flight scheduling as jets that undergo maintenance cannot be used to transport passengers. Conversely, flight scheduling impacts maintenance decisions as it may be necessary to revise preferred maintenance decisions if the set of accepted transportation requests can no longer be accommodated when the preferred maintenance decisions are implemented. In this section, we present a framework for capturing the interaction between operational maintenance decisions and flight scheduling.

Our approach to making operational maintenance decisions needs to be slightly modified to handle the interaction with flight scheduling.

Making maintenance decisions in advance. Typically, transportation requests are received a few days in advance and it is the task of the online flight scheduling algorithm to decide whether such a request can be accommodated. Therefore, if we try to make maintenance decisions for a given day only when the flight schedule for that day has been determined by the off-line flight scheduling algorithm, it might not be possible to implement the preferred maintenance decisions while still accommodating all transportation requests for that day. As a consequence, especially if this happens frequently, we may not be able to satisfy the maintenance mandates. To prevent this from happening, we make maintenance decisions a few days in advance. That is, on a given day t we make maintenance decisions for day $t + l$, where l is a parameter specifying how many days in advance maintenance decisions are made. (Note that the jets that will be maintained on days $t, \dots, t + l - 1$ have already been determined and these decisions are not changed.) By making maintenance decision a few days in advance, the online flight scheduling algorithm is aware of any reduced capacity due to maintenance decisions.

Updating maintenance decisions: Even if maintenance decisions are made a few days in advance, it may happen that preferred maintenance decisions for day $t + l$ cannot be implemented because a large number of travel requests has already been accepted for day $t + l$ (and these commitments have to be honored). In this situation, it is necessary to adjust the decisions for the jets that cannot be maintained on day $t + l$. These jets are scheduled for maintenance as early as possible after day $t + l$ (while ensuring that accepted requests on a day can always be accommodated). Note that by adjusting the day of maintenance for these jets forward in time maintenance capacity becomes available on day $t + l$. Thus, it may be a good idea to maintain some other jets on day $t + l$. To evaluate that possibility, we resolve MDP (and continue to iterate this process as long as maintenance decisions are adjusted due to the requirements of flight scheduling). Once all the maintenance decisions for day $t + l$ can be feasibly implemented, we move to the smoothing phase.

Thus, the daily early morning decision process on each day t is as follows. First, itineraries are constructed for all accepted travel requests for day t using the off-line flight scheduling algorithm. After the itineraries are constructed, MDP is solved to determine which jets are to be maintained

on day $t + l$ as well as to assign itineraries to the critical jets on day t . Let the set of jets to be maintained on day $t + l$ be denoted by \mathcal{J}^M . For each jet in \mathcal{J}^M , we check whether it can be maintained while still accommodating all requests that have already been accepted for day $t + l$. If the jet can be maintained feasibly, the decision for this jet is finalized and the online flight scheduling algorithm is made aware of that maintenance. If the jet cannot be feasibly maintained, the earliest day after $t + l$ is found where the jet can be maintained and the decision to maintain the jet on that day is finalized. If any of the maintenance decision for the jets in \mathcal{J}^M is adjusted, MDP is solved again to see if the maintenance capacity on day $t + l$ that has become available can be used effectively. The process continues until all maintenance decisions for day $t + l$ can feasibly be implemented. After that the smoothing algorithm is called to assign itineraries to the non-critical jets.

7 Case Study

In this section, we describe a case study based on simulated operations at DayJet Corporation, a startup PSOD air transportation provider. The emphasis of this case study is on evaluating the operational maintenance decisions in the presence of flight scheduling. The scenario in the case study represents an environment in which the travel demand is highly variable from day to day in order to stress test the models and solution approaches. A total of 174 jets are introduced into the fleet over a two year planning horizon with an average of 1 month between jet arrivals according to the contract with the jet manufacturer. The average number of jets arriving per month is lower over the first year. Furthermore, the growth rate of travel demand is also lower during the first year and gradually increases during the second year.

At the tactical level, we determine the capacity at the maintenance facility using our optimization-based local search algorithm with a time limit of 2 hours. The values of the local search parameters determining the neighborhood sizes are the same as the ones given in Section 4.3. The average flying time, f , gradually increases from 5 hours to 10 hours during the planning horizon in accordance with the scenario. The integrated framework of operational maintenance planning and flight scheduling is implemented as explained in Section 6 on each day of the planning horizon. The framework is tested in a simulation environment where travel requests are generated using an agent-based model developed by DayJet Corporation. Furthermore, online and off-line flight scheduling are performed using algorithms and decision support tools developed by DayJet as well. The values for the parameters of operational maintenance planning such as the look-ahead length k and the penalties for MDP and SP are chosen to be the same as the ones given in Section 5.2. Historical data regarding the travel requests indicates that requests arrive on average two days ahead of their travel date. Thus, accordingly, we set the parameter $l = 2$ corresponding to the number of days maintenance decisions are made in advance.

Figure 8 shows the number of jets maintained on each day and the histogram of the accumulated flying hours before maintenance. The number of jets maintained on each day still closely follows the maintenance capacity, especially during the second year. Since the growth rate of the fleet is small during the first year and the jets need to accumulate a minimum number of flying hours before maintenance, there are days where no jet can be maintained. The histogram shows that maintenance is still mostly done after accumulating 300 flying hours which is the target value. The standard deviation of the accumulated flying hours is larger compared to the results presented for operational maintenance planning in Section 5.2. One of the reasons for the larger standard

deviation is that travel demand is highly variable from day to day. While planning for maintenance two days in advance, the *anticipated* (average) flying times might seem to take a jet very close to the target H before maintenance although this target might be more likely to be missed with the *actual* realized flying times due to the high variation in travel demand. Furthermore, sometimes maintenance has to be postponed in order to accommodate the requests that have already been accepted, which in turn results in jets accumulating more flying hours than planned.

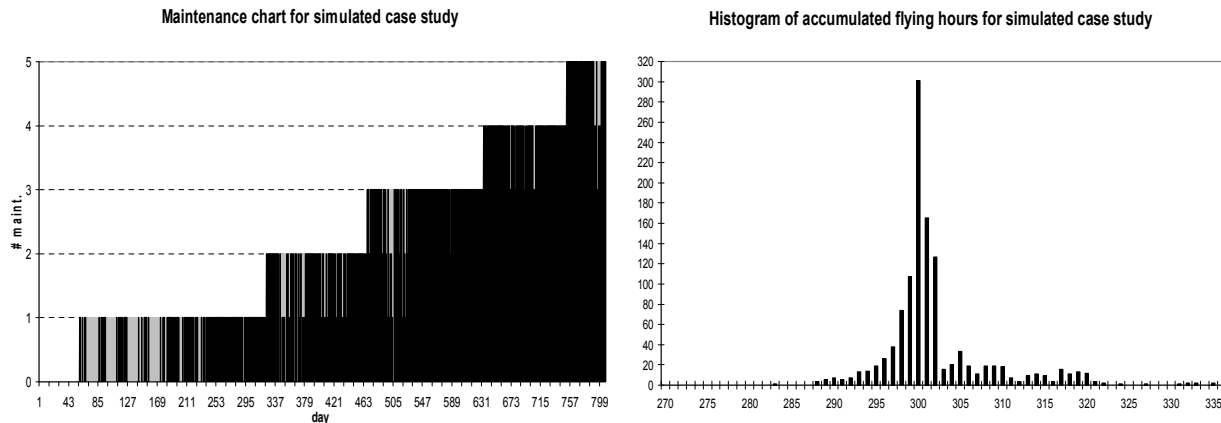


Figure 8: Maintenance chart and histogram for the simulated case study

Table 3 summarizes the results for the simulated case study. The first row shows the total number of times maintenance is performed during two years. The second row represents the percentage of total capacity used for maintenance during the planning horizon. The average and the standard deviation of the accumulated flying hours before maintenance are given in the third and the fourth rows, respectively. The fifth row corresponds to the percentage of maintenance activities that violate the maintenance rules. Finally, the last two rows give the percentage of maintenance activities that have to be postponed due to the impact of flight scheduling and the average number of days they are postponed for.

Table 3: Summary results for the simulated case study

No. of maint.	1175
% cap. used	89.46
Avg. acc. hrs.	301.95
Std. dev. of acc. hrs.	6.53
% infeasible maint.	1.96
% maint. postponed	6.04
Avg. no. of days postponed	1.26

The results in Table 3 demonstrate that we obtain very good results using the proposed framework of the integration of operational maintenance planning with flight scheduling for the operations of DayJet. A capacity usage rate of 90% and less than 2% violations of the maintenance rules indicates a good fit between the maintenance capacity and the requirements of operational planning.

The slight increase in the infeasibility rate, over the earlier experiments in Section 5.2, can be attributed to the need to postpone a portion of maintenance activities in order to accommodate accepted travel requests. However, as can be seen from Table 3, only 6% of the maintenance activities have to be postponed due to the impact of flight scheduling and on average they can be maintained the next day. Furthermore, we can still achieve our maintenance objectives since the average of the accumulated flying hours before maintenance is very close to the target 300 hours and the standard deviation is around 7 hours. Thus, our solution approaches are robust enough to handle high variability in travel demand.

8 Conclusions

We have presented models and solution approaches for tactical and operational planning of scheduled maintenance for per-seat, on-demand air transportation. An optimization-based local search algorithm has been developed to solve the tactical capacity planning problem. The solution approach for operational maintenance planning uses a look-ahead approach together with a smoothing phase to schedule the daily maintenance while determining the itinerary assignments to the jets. Finally, operational maintenance planning and flight scheduling have been integrated within a framework that has been tested with a simulated case study for the operations of a per-seat, on-demand air transportation provider, DayJet Corporation. All of our computational experiments for tactical planning, operational planning and the proposed integrated framework have demonstrated the efficacy of our solution approaches.

References

- [1] S. Anily, C.A. Glass and R. Hassin. The Scheduling of Maintenance Service. *Discrete Applied Mathematics*, 82:27-42, 1998.
- [2] S. Anily, C.A. Glass and R. Hassin. Scheduling Maintenance Services to Three Machines. *Annals of Operations Research*, 86:375-391, 1999.
- [3] A. Bar-Noy, R. Bhatia, J. Naor and B. Schieber. Minimizing Service and Operation Costs of Periodic Scheduling. *Mathematics of Operations Research*, 27:518-544, 2002.
- [4] S. Deris, S. Omatu, H. Ohta, Lt. C.S. Kutar and P.A. Samat. Ship Maintenance Scheduling by Genetic Algorithm and Constraint-based Reasoning. *European Journal of Operational Research*, 112:489-502, 1999.
- [5] S.K. Dhall and C.L. Liu. On a Real-Time Scheduling Problem. *Operations Research*, 26:127-140, 1978.
- [6] D. Espinoza, R. Garcia, M. Goycoolea, G.L. Nemhauser and M.W.P. Savelsbergh. Per-Seat, On-Demand Air Transportation Part I: Problem Description and an Integer Multi-Commodity Flow Model. *Transportation Science*, 42:263-278, 2008.
- [7] D. Espinoza, R. Garcia, M. Goycoolea, G.L. Nemhauser and M.W.P. Savelsbergh. Per-Seat, On-Demand Air Transportation Part II: Parallel Local Search. *Transportation Science*, 42:279-291, 2008.

- [8] R. Gopalan and K.L. Talluri. Mathematical Models in Airline Schedule Planning: A Survey. *Annals of Operational Research*, 76:155-185, 1998.
- [9] A. Grigoriev, J. van de Klundert and F.C.R. Spijksma. Modeling and Solving the Periodic Maintenance Problem. *European Journal of Operational Research*, 172:783-797, 2006.
- [10] A. Haghani and Y. Shafahi. Bus Maintenance Systems and Maintenance Scheduling: Model Formulations and Solutions. *Transportation Research Part A*, 36:453-482, 2002.
- [11] C. Kenyon, N. Schabanel and N Young. Polynomial-Time Approximation Scheme for Data Broadcast. *European Journal of Operational Research*, 172:783-797, 2006.
- [12] B. Kralj and R. Petrovic. A Multiobjective Optimization Approach to Thermal Generating Units Maintenance Scheduling. *European Journal of Operational Research*, 84:481-493, 1995.
- [13] V. Mattila and K. Virtanen. A Simulation-based Optimization Model to Schedule Periodic Maintenance of a Fleet of Aircraft. *Proc. 2005 European Simulation and Modelling Conference*, 479-483, 2005.
- [14] C. Sriram and A. Haghani. An Optimization Model for Aircraft Maintenance Scheduling and Re-assignment. *Transportation Research Part A*, 37:29-48, 2003.
- [15] H.M. Wagner, R.J. Giglio and R.G. Glaser. Preventive Maintenance Scheduling by Mathematical Programming. *Management Science*, 10:316-334, 1964.