Erasmus University Rotterdam

Erasmus School of Economics

Master Thesis Operations Research and Quantitative Logistics

# Network Seat Inventory Control

Finding Effective Solution Methods for Network Seat Inventory Control Problem incorporating Aircraft Configuration Selection

**Abstract**

This thesis aims to find an effective method to maximize the revenue of airlines operating a large-scale network using network seat inventory control. The problem consists of allocating flight leg capacities to customer requests under network effects and fixed fare classes. Compared to literature, this thesis incorporates aircraft configuration selection during the optimization and finds a control mechanism that is flight leg based, instead of the traditional flight leg-cabin based. We propose three variants of an approximation method based on decomposition by a mathematical programming model and solving independent single-leg problems using a dynamic programming model. Bid prices are obtained as a control mechanism, which are dynamic, being a function of both the remaining time and remaining capacity in the cabins. Applying the approaches to the network of a major European airline showed that two of the three approaches still work within a reasonable amount of time. Moreover, a Monte Carlo simulation of the booking process is performed using eight smaller subnetworks of the network of the airline. The simulation showed that the proposed approaches can, on average, obtain revenue increases of up to 0.8% compared to two well-known benchmark solution methods, corresponding to an average gap of 2.3% for the introduced approximate upper bound. However, sensitivity analysis showed that the relative performance of the proposed approaches is sensitive to the characteristics of the network. That is, when demand factors increase, the relative performance increase seems to decrease slightly and when the network contains more connecting flow, the relative performance increase even becomes negative.

Keywords: Network seat inventory control; Network problem; Quantity-based revenue management; Yield management; Airlines; Bid price control; Aircraft configuration selection.

*Name student:*
Julian van 't Hooft

*Student ID number:*
471932

*Supervisor:*
Remy Spliet

*Second assessor:*
Lisanne van Rijn

*Date final version*: February 28, 2023

# Contents

## List of Tables



## List of Figures

# Acronyms

**CEC** Certainty equivalent control. 20

**DAVN** Displacement-adjusted virtual nesting. 12

**DD-DP** Dual-based decomposition - dynamic programming. 2

**DLP** Deterministic linear programming. 18

**DMILP** Deterministic mixed integer linear programming. 19

**DPD** Dynamic programming decomposition. 13

**EDLP** Extended deterministic linear programming. 18

**EMSR** Expected marginal seat revenue. 8

**MPC** Mathematical programming control. 28

**OCC** Opportunity cost control. 29

**OCD-DP** Opportunity cost-based decomposition - dynamic programming. 2

**OD** Origin-destination. 4

**ODF** Origin-destination and fare class. 4

**PHMILP** Perfect hindsight mixed integer linear programming. 29

**PODS** Passenger origin and destination simulator. 9

**ROCD-DP** Relaxed opportunity cost-based decomposition - dynamic programming. 2

# 1 Introduction

After the deregulation of the U.S. airline industry, revenue management systems became indispensable for airline capacity control, having resulted in four to ten percent increase in revenue (Fuchs, 1987). Simply put, American Airlines described in its 1987 annual report the objective of revenue management as to sell "the right seats to the right customer at the right price". Price and seat inventory control are major options among the techniques used in revenue management. Airlines practice price discrimination by offering a wide variety of fare products, which are uniquely defined by price, service and restrictions, to capture demands from different market segments and different times of the season. However, due to keen competition, price options are limited by what other airlines offer in similar markets. Individual carriers do, however, have full control over the seat inventory control component. Therefore, managing the mix of fares, in contrast to managing the fares itself, is often the most effective and important part of revenue management systems.

Seat inventory control is about controlling the number of seats available for each fare product at any given time. This type of quantity-based revenue management is, next to the airline industry, also used in different industries such as the hotel, cruise line, car rental, entertainment and media industries. This thesis will specifically focus on the airline industry and the research will be conducted for a major airline in Europe operating a large-scale network. Essentially, the problem of seat inventory control within the airline industry can be seen as finding a trade-off between revenue realized by greater demand for discounted seats against revenue lost due to denying full-fare requests by having already accepted prior discounted seat sales. The difficulty here is that discounted-fare requests are often observed before full-fare requests, which is caused by the nature of the customer of the respective fare classes, namely business travelers for the full fares and leisure travelers for the discounted fares.

Moreover, airlines operating a large-scale network also need to consider network impacts on the traffic flows, such as connecting and through traffic. This is because maximizing single-leg revenues is not necessarily the same as maximizing total revenues in a network of flight legs. Indeed, Talluri and Van Ryzin (2004b) cite multiple simulation studies of airlines operating a large-scale network where network methods show significant revenue benefits over single-leg methods of 1.5% up to 3% for higher load factors. Therefore, airlines operating a large-scale network need to manage the capacities of a group of connecting flights within a network, where the individual flight legs can have a mix of local and connecting traffic. This problem is known in literature as the network problem and in the airline industry called the network seat inventory control problem.

Seat inventory control is the final step in the three-phase marketing process, consisting of pricing, scheduling and inventory control. In theory, this three-phase process should be optimized simultaneously, however, due to the size and complexity of the problem the three processes are solved sequentially. By being the last step in the optimization, the flight schedule and pricing structure are fixed inputs for the seat inventory control problem (Williamson, 1992).

In this thesis, we try to find an effective method to maximize revenue by network seat inventory control. Compared to literature, this thesis focuses on incorporating aircraft configuration selection during the optimization and, thereby, obtaining a control mechanism that is flight leg based, instead of the traditional flight leg-cabin based. Aircraft configuration selection is of importance as many airlines use a mobile cabin divider to partition the economy and business cabins for short and medium-haul flights and, thereby, have freedom in selecting the aircraft configuration before departure. To segment the market better and to provide more comfort and privacy to the business traveler, many airlines leave the aisle or middle seat open when having rows of two or three chairs, respectively. Therefore, the number of seats open for sale differs per configuration and also per cabin and, hence, must be accounted for in the optimization.

Some airlines currently deal with aircraft configuration selection by fixing the aircraft configuration at the beginning of the optimization and obtaining a control mechanism that is flight leg-cabin based. That is, the different cabins are treated to be independent 'flight legs' and the control mechanisms are optimized separately for the different cabins. This thesis aims to incorporate the aircraft configuration selection during the optimization and, thereby, obtain a control mechanism that is flight leg based and, therefore, a function of the remaining capacities in the different cabins. In this way, the control mechanism incorporates the flexibility to change the aircraft configuration at all times.

The network problem can relatively easily be formulated as a stochastic dynamic programming model. However, solving this model becomes problematic, as the model suffers from the curse of dimensionality. In this thesis, we propose three variants of an approximation method based on decomposition by a mathematical programming model and solving independent single-leg problems using a dynamic programming model. The first variant is called the Opportunity Cost based Decomposition - Dynamic Programming (OCD-DP) approach, the second variant is called the Relaxed Opportunity Cost based Decomposition - Dynamic Programming (ROCD-DP) approach and the third variant is called the Dual based Decomposition - Dynamic Programming (DD-DP) approach. The use of the dynamic programming model allows us to solve the single-leg problems to optimality and incorporates the aircraft configuration selection. The obtained bid prices, which are used as a control mechanism, are dynamic in both the remaining time and the remaining capacity in the different cabins.

To measure the potential of increased revenues for the different approaches, a booking process simulation was developed. The proposed approaches are evaluated using a large-scale network and eight smaller subnetworks generated from the large-scale network. Here, the large-scale network is acquired from the major European airline for which this research is conducted. The entire large-scale network is used to demonstrate whether the proposed approaches still work for a real, large network of a major airline. The smaller subnetworks allow for the application of a Monte Carlo simulation to derive interesting summary statistics. The results are benchmarked against two well-known solution methods and an approximate upper bound.

The case study on the entire-large scale network showed the impracticality of the OCD-DP approach, namely with current implementation, the computation time of the optimization of a single booking period was approximately 4.7 hours, which was approximately 4 and 7 times longer than the ROCD-DP and DD-DP approaches, respectively. Using the eight subnetworks, the Monte Carlo simulation of the booking process showed that the remaining two approaches can, with lower final load factors, on average, obtain revenue increases of up to 0.8% compared to the benchmark solution approaches. This corresponded to an average gap of 2.3% for the approximate upper bound. However, sensitivity analysis showed that this relative performance can be subjective to the characteristics of the network. That is, the relative performance increase of the proposed approaches seems to decrease slightly when demand factors increase and even becomes negative when the considered network contains more connecting flow.

All considered approaches are fairly robust when demand estimates are heavily perturbed, which can be explained by the daily revision of the booking limits. Nonetheless, when demand estimates are heavily perturbed, the performance of the ROCD-DP and DD-DP approaches compared to one of the benchmark solution approaches becomes fairly similar. Moreover, for all experiments, the

ROCD-DP approach outperformed the DD-DP approach, however, the differences were minimal. As the ROCD-DP approach has a longer computation time, the choice between the two approaches must be made depending on the interest of the airline.

The remainder of the thesis is organized as follows: in Chapter 2 a detailed problem description is presented. In Chapter 3, we provide a literature review of the seat inventory control problem. The methodology is presented in Chapter 4. In Chapter 5, we present numerical experiments together with the results for the different approaches. Lastly, in Chapter 6 conclusions and topics for future research are provided.

## 2 Problem Description

In this section, we will describe the network seat inventory control problem in which we incorporate aircraft configuration selection. Throughout the thesis, we will denote vectors and matrices in bold. Moreover, let $\mathbb{1}\{E\}$ denote the indicator function of an event $E$ and let $x^+ = \max\{0, x\}$.

We consider an airline operating a network, which is described by $m$ flight legs, and the airline sells $n$ different products to its customers. Flight legs $i$ are numbered by 1 through $m$ and products $j$ are numbered by 1 through $n$. A flight leg is part of a flight involving a take-off and landing. A product can be seen as an origin-destination and fare class combination (ODF) itinerary. Here, the fare class consists of the product's price, purchase terms and restrictions. Therefore, a product can be seen as a bundle of the $m$ flight legs, which is sold for a given price with certain purchase terms and restrictions. Lastly, a set of fare classes is defined for each origin-destination (OD) itinerary.

The state of the network is defined by the remaining capacities on the $m$ flight legs. Here, the remaining capacity of a flight leg is given by the initial capacity diminished by the seats sold on that flight leg. To segment the market better, airlines partition the aircraft into different cabins. For the remainder of the thesis, we assume that there are at most two cabins, namely the economy (M) and business (C) cabins. However, one must note that the introduced concepts are not limited to using only two cabins. To incorporate the use of two cabins, we split the capacity for any flight leg into two distinct, cabin-specific, flight leg capacities. That is, we introduce the integer $2m$-vector $\mathbf{x} = \left[x_1^M, x_1^C, ..., x_m^M, x_m^C\right]^\intercal$, where $x_i^M$ and $x_i^C$ equal the capacity of flight leg $i$ and cabin M and C, respectively. For the remainder of the thesis, flight leg capacities are always cabin-specific and, therefore, flight leg capacities $i$ are numbered by 1 through $2m$. Moreover, let the non-negative integer $2m-$vector $\mathbf{X} = \left[X_1^M, X_1^C, ..., X_m^M, X_m^C\right]^\intercal$ denote the initial flight leg capacities, that is, $X_i^M$ and $X_i^C$ equal the initial capacity of flight leg $i$ and cabin M and C, respectively.

For short and medium-haul flights, we have the freedom to change the aircraft configuration by reallocating the mobile cabin divider, which partitions economy and business cabins. In this way, the remaining flight leg capacities open for sale can differ per chosen aircraft configuration. To incorporate aircraft configuration selection, we introduce the non-negative integer $m$-vector $\mathbf{z} = \left[z_1, ..., z_m\right]^\intercal$, where $z_i$ denotes the aircraft configuration of flight leg $i$. Here, $z_i = 0$ represents the configuration with the lowest number of business passengers, $z_i = 1$ represents the configuration with the second-lowest number of business passengers, and so on. Both the remaining and initial flight leg capacity vectors $\mathbf{x}$ and $\mathbf{X}$ are specified for the case when the aircraft configuration is fixed to the zero-configuration, that is $\mathbf{z} = \left[0, ..., 0\right]^\intercal$.

The mobile cabin divider does, however, only allow specific configurations. Namely, the cabin divider always segregates entire rows and the cabin divider can only be positioned in the front part of the aircraft. Moreover, in the business cabin, the aisle seat or the middle seat is often left empty when having rows of two or three chairs, respectively. We introduce a non-negative integer $m$-vector $\mathbf{c} = \left[c_1, ..., c_m\right]^\intercal$, where $c_i$ equals the number of different configurations on flight leg $i$. Moreover, we introduce the integer $(m \times 2m)$-matrix $\mathbf{B}$, where the $i^{th}$ row of $\mathbf{B}$ represents the changes in the remaining flight leg capacities when reallocating the mobile cabin divider of flight leg $i$ by one row to allow for more business passengers. More specifically, $\mathbf{B}_j^i$, which is the $j^{th}$ element of the $i^{th}$ row of $\mathbf{B}$, represents the change in the $j^{th}$ flight leg capacity of the vector $\mathbf{x}$ when reallocating the mobile cabin divider of flight leg $i$ by one row to allow for more business passengers. The change in remaining flight leg capacities open for sale by choosing a different configuration other than the zero-configuration can now be given by $\mathbf{B}^\intercal\mathbf{z}$, with $\mathbf{z} \in \mathbb{Z}_{0+}^m$ and $\mathbf{z} < \mathbf{c}$. Here, $\mathbb{Z}_{0+}^m$ denotes the $m$-dimensional set of non-negative integers.[1]

An airline also has the possibility to up- or downgrade customers, which previously purchased a certain product, to the other cabin. This gives more freedom in selecting different aircraft configurations and, therefore, more freedom in accepting requests for higher-revenue products and, hence, must be accounted for. We introduce the integer $m$-vector $\mathbf{g} = \left[g_1, ..., g_m\right]^\intercal$, where $g_i$ equals the number of upgrades from the economy to the business cabin on flight leg $i$. For business reasons, the number of up- and downgrades is limited. Let the maximum number of down- and upgrades on each of the flight legs be given by the non-negative integer $m$-vectors $\mathbf{d} = \left[d_1, ..., d_m\right]^\intercal$ and $\mathbf{f} = \left[f_1, ..., f_m\right]^\intercal$, respectively. That is, $d_i$ and $f_i$ equal the maximum number of down- and upgrades on flight leg $i$, respectively. We introduce the $(m \times 2m)$-matrix $\mathbf{C}$, where the $i^{th}$ row of $\mathbf{C}$ represents the changes in the remaining flight leg capacities when we upgrade a passenger from the economy to the business cabin on flight leg $i$. More specifically, the $j^{th}$ element of the $i^{th}$ row of $\mathbf{C}$, given by $\mathbf{C}_j^i$, represents the change in the $j^{th}$ flight leg capacity of the vector $\mathbf{x}$ when upgrading a passenger from the economy to the business cabin on flight leg $i$. By definition, matrix $\mathbf{C}$ is given by:

$$\mathbf{C} = \begin{bmatrix} 1\text{ -1} & 0\;0 & \dots & 0\;0 \\ 0\;0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0\;0 \\ 0\;0 & \dots & 0\;0 & 1\text{ -1} \end{bmatrix}$$

The change in remaining flight leg capacities open for sale by up- and downgrading customers can now be given by $\mathbf{C}^\intercal\mathbf{g}$, with $\mathbf{g} \in \mathbb{Z}^m$ and $-\mathbf{d} \leq \mathbf{g} \leq \mathbf{f}$. Here, $\mathbb{Z}^m$ denotes the m-dimensional set of integers. Using previous definitions, we define the effective remaining flight leg capacities as the vector $\mathbf{x}' = \mathbf{x} + \mathbf{B}^\intercal\mathbf{z} + \mathbf{C}^\intercal\mathbf{g}$. That is, the effective remaining flight leg capacities $\mathbf{x}'$ define for each flight leg the remaining number of seats open for sale. Since we cannot oversell flight legs, we have that all elements of the vector $\mathbf{x}'$ are non-negative integer variables, therefore $\mathbf{x}' = \mathbf{x} + \mathbf{B}^\intercal\mathbf{z} + \mathbf{C}^\intercal\mathbf{g} \geq 0$. Moreover, we can restrict the aircraft configurations $\mathbf{z}$ to the set $\mathcal{Z}(\mathbf{x})$, which is defined by Equation 1. Lastly, we can restrict the remaining flight leg capacities $\mathbf{x}$ to set $\mathcal{X}$, which is defined by Equation 2.

$$\mathcal{Z}(\mathbf{x}) = \left\{\mathbf{g} \in \mathbb{Z}^m, \mathbf{z} \in \mathbb{Z}_{0+}^m : \mathbf{z} < \mathbf{c}, -\mathbf{d} \leq \mathbf{g} \leq \mathbf{f}, \mathbf{x} + \mathbf{B}^\intercal\mathbf{z} + \mathbf{C}^\intercal\mathbf{g} \geq 0\right\} \tag{1}$$

$$\mathcal{X} = \left\{\mathbf{x} \in \mathbb{Z}^{2m}, \mathbf{g} \in \mathbb{Z}^m, \mathbf{z} \in \mathbb{Z}_{0+}^m : \mathbf{x} \leq \mathbf{X}, \mathbf{z} < \mathbf{c}, -\mathbf{d} \leq \mathbf{g} \leq \mathbf{f}, \mathbf{x} + \mathbf{B}^\intercal\mathbf{z} + \mathbf{C}^\intercal\mathbf{g} \geq 0\right\} \tag{2}$$

To clarify the previously mentioned notation, consider Example 2.1 of a simple network of a single flight leg.

**Example 2.1.** *Consider a single flight leg network to which an aircraft is assigned with an initial capacity of 130 seats in the economy cabin and 8 seats in the business cabin. That is,* $\mathbf{X} = \left[X_1^M, X_1^C\right]^\intercal = \left[130, 8\right]^\intercal$. *Moreover, the aircraft type allows us to reallocate the mobile cabin divider,*

[1] Note that by using this notation, we assume constant changes in capacity. However, variable changes are possible using corrective extra terms.

*and therefore to increase the capacity of the business cabin. The aircraft has rows of three seats and in total 4 different configurations. The middle seat is blocked in the business cabin. Also, we allow upgrading of in total 3 passengers from the economy to the business cabin. Downgrades are prohibited. Consequently, we have that* $\mathbf{c} = \begin{bmatrix} c_1 \end{bmatrix} = \begin{bmatrix} 4 \end{bmatrix}$ *and, therefore,* $\mathbf{z} = \begin{bmatrix} z_1 \end{bmatrix}$, *where* $z_1 \in \{0, ..., 3\}$. *Moreover, we have that* $\mathbf{d} = \begin{bmatrix} d_1 \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix}$ *and* $\mathbf{f} = \begin{bmatrix} f_1 \end{bmatrix} = \begin{bmatrix} 3 \end{bmatrix}$ *and, therefore,* $\mathbf{g} = \begin{bmatrix} g_1 \end{bmatrix}$, *where* $g_1 \in \{0, ..., 3\}$. *In addition, matrix* $\mathbf{B}$ *is given by* $\mathbf{B} = \begin{bmatrix} -6 & 4 \end{bmatrix}$. *That is, if we reallocate the mobile cabin divider by one row to allow for more business passengers, the first (and only) row of matrix* $\mathbf{B}$ *shows that this reallocation results in a respective decrease of six and an increase of four seats for the effective remaining economy and business flight leg capacities* $x'^M_1$ *and* $x'^C_1$. *Lastly, matrix* $\mathbf{C}$ *is given by* $\mathbf{C} = \begin{bmatrix} 1 & -1 \end{bmatrix}$.

*Now, at a certain moment in time, in total 120 seats are sold in the economy cabin and 13 seats are sold in the business cabin. Therefore, the remaining capacity vector* $\mathbf{x}$ *is given by Equation 3.*

$$\mathbf{x} = \begin{bmatrix} x^M_1 \\ x^C_1 \end{bmatrix} = \begin{bmatrix} X^M_1 \\ X^C_1 \end{bmatrix} - \begin{bmatrix} \textit{Seats sold cabin } M \\ \textit{Seats sold cabin } C \end{bmatrix} = \begin{bmatrix} 130 \\ 8 \end{bmatrix} - \begin{bmatrix} 120 \\ 13 \end{bmatrix} = \begin{bmatrix} 10 \\ -5 \end{bmatrix} \tag{3}$$

*Moreover, for this remaining capacity, the only possible aircraft configuration is* $z_1 \in \mathcal{Z}(\mathbf{x}) = \{2\}$ *and, therefore,* $\mathbf{z} = \begin{bmatrix} z_1 \end{bmatrix} = \begin{bmatrix} 2 \end{bmatrix}$. *Moreover, this configuration requires upgrading two economy passengers to the business cabin to make sure that we do not have a negative effective remaining capacity in the economy cabin. That is,* $\mathbf{g} = \begin{bmatrix} g_1 \end{bmatrix} = \begin{bmatrix} 2 \end{bmatrix}$. *The effective remaining flight leg capacities are then given by Equation 4. That is, we have still one remaining capacity for the business cabin open for sale.*

$$\mathbf{x}' = \mathbf{x} + \mathbf{B}^\intercal \mathbf{z} + \mathbf{C}^\intercal \mathbf{g} = \begin{bmatrix} 10 \\ -5 \end{bmatrix} + \begin{bmatrix} -6 \\ 4 \end{bmatrix} \begin{bmatrix} 2 \end{bmatrix} + \begin{bmatrix} 1 \\ -1 \end{bmatrix} \begin{bmatrix} 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{4}$$

We define the $(2m \times n)$-incidence matrix $\mathbf{A} = \begin{bmatrix} a_{ij} \end{bmatrix}$, where $a_{ij} = 1$ if flight leg capacity $i$ is used by product $j$ and $a_{ij} = 0$ otherwise. Consequently, $\mathbf{A}^i$, which denotes the $i^{th}$ row of $\mathbf{A}$, has entry one in column $j$ if product $j$ uses flight leg capacity $i$. Likewise, $\mathbf{A}_j$ denotes the $j^{th}$ column of $\mathbf{A}$, and is the incidence vector for product $j$. Matrix $\mathbf{A}$ may contain repeated columns for each fare class on a given OD itinerary. Moreover, we define the set of flight leg capacities used by product $j$ by $A_j$. That is, if a flight leg capacity $i$ is used by product $j$, we indicate it by $i \in A_j$. If product $j$ is sold, the remaining flight leg capacity vector is updated to $\mathbf{x} - \mathbf{A}_j$.

To model the booking process, we assume that time is discrete with a finite booking horizon of length $T$. The index $t$ denotes the remaining time, so time is counted backward. That is, $t = T$ is the beginning of the booking horizon and $t = 0$ is the time of service. Moreover, we assume sufficiently fine discretization of time, such that at most one request can arrive in a single time-period. We, therefore, do not allow group bookings. Let the $n$-vector $\mathbf{r} = \begin{bmatrix} r_1, ..., r_n \end{bmatrix}^\intercal$ be the revenues associated with the $n$ network products, that is, $r_j$ equals the revenue of product $j$. For analytical purposes, we model the demand for the $n$ products at time $t$ as a realization of a single random vector $\mathbf{R}(t) = \begin{bmatrix} R^t_1, ..., R^t_n \end{bmatrix}^\intercal$. Here, $R^t_j = r_j > 0$ if a request for product $j$ occurred with associated revenue $r_j$ and $R^t_j = 0$ if no request occurred for product $j$. Consequently, if no requests arrived at time $t$, we have a realization of $\mathbf{R}(t) = \mathbf{0}$. Moreover, as an example, if we have $n = 4$ products and a request for product 2 arrived at time $t$ with associated revenue $r_2 = 600$, the realization of $\mathbf{R}(t)$ would be given by $\mathbf{R}(t) = \left[0, 600, 0, 0\right]^\intercal$. Using the definition of $\mathbf{R}(t)$, the aggregate demand process at time $t$, $\mathbf{D}(t)$, can be represented as $\mathbf{D}(t) = \left[\sum_{k=1}^{t} \mathbb{1}\left\{R_1^k > 0\right\}, ..., \sum_{k=1}^{t} \mathbb{1}\left\{R_n^k > 0\right\}\right]^\intercal$. In addition, let $\boldsymbol{\mu}(t) = \left[\mu_1^t, ..., \mu_n^t\right]^\intercal$ denote the vector of mean aggregate demand to come at time $t$, that is, $\mu_j^t$ denotes the mean aggregate demand for product $j$ at time $t$. Then, $\boldsymbol{\mu}(t)$ can be given by $\boldsymbol{\mu}(t) = \left[\mu_1^t, ..., \mu_n^t\right]^\intercal = \left[E\left[\sum_{k=1}^{t} \mathbb{1}\left\{R_1^k > 0\right\}\right], ..., E\left[\sum_{k=1}^{t} \mathbb{1}\left\{R_n^k > 0\right\}\right]\right]^\intercal$. Lastly, let the associated probability of a request for product $j$ at time $t$ be given by $p_j^t$ and let $p_0^t$ denote the probability of no request at time $t$. As we have at most one request per time-period, we have that $p_0^t = 1 - \sum_{j=1}^{n} p_j^t$.

We assume statistical independence of demand for different fare classes. That is, it is assumed that the demand for a given fare class is independent of the used capacity control and, consequently, independent of whether other fare classes are open or not. Therefore, the sequence $\{\mathbf{R}(t); t \leq T\}$ is assumed to be independent across time $t$. This assumption is obviously somewhat unrealistic, as it is, for example, very plausible that a customer will buy a higher fare when the discounted fares are closed. This phenomenon is called buy-up, and from the firm's point of view, it is often referred to as sell-up. However, we assume that these dependencies are dealt with in the demand forecasts and, therefore, we will not consider any form of customer choice behavior. Moreover, we assume that the joint distribution is known at each period $t$.

We ignore overbooking and, thereby, ignore cancellations and no-shows at this stage. Here, overbooking is the widely used concept of selling more seats than the capacity of an aircraft to compensate for losses of cancellations and to benefit from no-shows. With overbooking, an airline defines per flight a maximum number of seats the airline is willing to risk to deny boarding and this number determines how many extra seats may be sold. Lastly, we also ignore code-sharing at this stage, which is a marketing agreement by airlines in which an airline sells seats of other airlines as their own.

Network seat inventory control now simplifies to deciding whether to accept an incoming request for product $j$ at time $t$ given the current remaining capacities $\mathbf{x}$. Here, the objective is to maximize the expected revenue to go. That is, we maximize the total expected revenue of the $m$ flight legs over the $T$ periods to come, where demand is uncertain. For this, we introduce the length $n$ decision vector $\mathbf{u}(t, \mathbf{x}, \mathbf{r}) = \left[u_1(t, \mathbf{x}, r_1), ..., u_n(t, \mathbf{x}, r_n)\right]^\intercal$, where:

$$u_j(t, \mathbf{x}, r_j) = \begin{cases} 1 & \text{if we accept a request for product } j \text{ with corresponding revenue } r_j \\ & \text{in period } t \\ 0 & \text{otherwise.} \end{cases}$$

Since we have at most one request in any period and we cannot oversell flight legs, we can restrict the vector $\mathbf{u}(t, \mathbf{x}, \mathbf{r})$ to the set $\mathcal{U}(\mathbf{x})$, which is defined by Equation 5. Here, $E^n$ denotes the set of unit $n$-vectors, that is $E^n = \{e_0, e_1, ..., e_n\}$, with $e_0$ the zero $n$-vector and $e_j$ the $j$-th unit $n$-vector.

$$\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \in E^n, \mathbf{z} \in \mathbb{Z}_{0+}^m, \mathbf{g} \in \mathbb{Z}^m : \mathbf{z} < \mathbf{c}, -\mathbf{d} \leq \mathbf{g} \leq \mathbf{f}, \mathbf{A}\mathbf{u} \leq \mathbf{x} + \mathbf{B}^\intercal\mathbf{z} + \mathbf{C}^\intercal\mathbf{g}\} \tag{5}$$

Consequently, solving the network seat inventory control problem to optimality boils down to determining at any time $t$, given the remaining flight leg capacities $\mathbf{x}$, the optimal decisions $\mathbf{u}^*(t, \mathbf{x}, \mathbf{r})$.

# 3 Literature Review

Within seat inventory control, we can distinguish between two different scopes, namely single-leg capacity control and network capacity control. The first, within the airline industry known as the single-leg problem, looks at a single flight leg that is sold to differentiated demand classes. For an airline that is operating a point-to-point network with isolated, non-stop flights, a flight leg approach is sufficient for seat inventory control (Williamson, 1992). Network capacity control tries to find the same capacity controls, however, now in a setting where products require multiple flight legs. Nowadays, many quantity-based revenue management problems are network problems, however, in practice, they are still commonly solved as a collection of single-leg problems (Birbil et al., 2014). Also, many heuristics for solving the network problem use single-leg models. Therefore, first, the single-leg problem will be discussed in Section 3.1 and second, the network problem is discussed in Section 3.2. In Section 3.3, we discuss two commonly used control mechanisms for network capacity control. Lastly, in Section 3.4, the different optimization algorithms and control methodologies for the network capacity control problem are weighed against each other.

## 3.1 Single-leg problem

The single-leg problem is well-studied in the academic literature. In the past, a variety of exact and heuristic methods have been introduced. We refer the interested reader to the book by Talluri and Van Ryzin (2004b) for an extensive discussion on the single-leg problem. The models can generally be grouped into either solving the so-called static or dynamic version of the problem. In short, most static models assume the following assumptions: (1) sequential arrival of demand of different fare classes; (2) low-before-high fare bookings; (3) no batch bookings, or if there are batch bookings, they can be partially accepted; (4) statistical independence of demand for different classes; (5) no network effects taken into account; and, (6) no no-shows and cancellations and, hence, no overbooking. Dynamic models model the demand for each class as a stochastic process and allow for an arbitrary order of arrival and, therefore, relax assumptions (1) and (2). Dynamic models are, however, not a strict generalization of the static models, as dynamic models need to assume Markovian arrivals to remain tractable. While for the static model inventory control can be implemented by putting booking limits on the fare classes, for the dynamic model this acceptance of booking requests is now based on the remaining time to departure and the number of unsold seats. The Markovian assumption limits modeling different levels of variability in demand. Moreover, we require a so-called booking curve, which is an estimate of the pattern of arrival over time. Assumptions (3)-(6) are still retained for the dynamic model. In the end, choosing between static and dynamic models boils down to choosing which set of approximations is more reasonable and is dependent on the type and availability of the data (Talluri and Van Ryzin, 2004b). In Section 3.1.1 and 3.1.2, we provide relevant literature for the static and dynamic models, respectively. Lastly, in Section 3.1.3, we discuss an extension that explicitly models customer-choice behavior.

### 3.1.1 Static models

Littlewood (1972) was the first to contribute to the single-leg problem, more specifically he looked at a single-leg with two fare classes. In his model, he equated the expected marginal revenues of the two fare classes to determine whether to accept or reject a request for a specific fare for a flight leg. Bhatia and Parekh (1973) and Richter (1982) found equivalent formulations of Littlewood's rule and through these extensions, Littlewood's rule was proven to be optimal for the nested, two-fare class, single-leg problem. Belobaba (1987) proposed the Expected Marginal Seat Revenue (EMSR) heuristic for the nested, multiple-fare class, single-leg problem. This heuristic only produces optimal booking limits in the two-fare case, however, it is easy to implement and gives reasonable approximations for typical airline demand distributions. Belobaba (1989) later refines the EMSR heuristic, called EMSRb, which has been widely implemented in the airline industry. Assuming independent demands, Brumelle and McGill (1993), Curry (1990) and Wollmer (1992) independently found the optimal booking limits for the nested, multiple fare-class, single-leg problem. All solutions mentioned so far, require previously stated assumptions (1)-(6). Robinson (1995) was able to find optimal booking limits while relaxing assumption (2), that is, the low-before-high fare assumption is relaxed.

### 3.1.2 Dynamic models

Due to the nature of the dynamic problem, optimal control policies are often determined by using dynamic programming approaches. The earliest work on the dynamic problem was by T. C. Lee and Hersh (1993), which formulated the problem as a Markov decision process. Their work is refined by integrating important practical issues, such as cancellations, no-shows and overbooking (e.g. Lautenbacher and Stidham Jr, 1999; Liang, 1999; Subramanian et al., 1999). Gerchak et al. (1985) were the first to provide dynamic structural results for this type of problem.

### 3.1.3 Discrete-choice models

A key assumption of all so-far mentioned models is that demand for a given class is independent of the used capacity controls, more specifically, it is independent of whether other classes are open or not. Obviously, this assumption is somewhat unrealistic, making it a weakness in the design of the models so far. For instance, it is very plausible that the likelihood of a customer buying a full-fare ticket depends on whether a discount fare is available at the same time. The passenger origin and destination simulator (PODS) studies by Belobaba and Hopperstad (1999) showed the significance of customer-choice models on the performance of revenue management systems. Therefore, this topic gained momentum lately in literature. R. Phillips (1994) introduced a method that adjusts the controls based on demand forecast, thereby incorporating the current controls in place. Moreover, Belobaba (1987) adjusted his already introduced EMSR heuristic to incorporate buy-up probabilities. Here, buy-up is the phenomenon of purchasing a higher fare class when the lower fare classes are already closed. While the method was conceptually attractive, it had several methodological difficulties. For example, the method only holds for the two-fare class model and estimating the probabilities is hard since one cannot observe buy-ups directly. Talluri and Van Ryzin (2004a) provide an exact analysis of the problem under a general discrete choice model of demand. Other main references on this topic are Vulcano et al. (2010) and Osadchiy and Vulcano (2010). More recent papers use deep reinforcement learning to incorporate customer choice-based models, like the one of Alamdari and Savard (2021).

## 3.2 Network problem

Similar to the single-leg problem, the network problem can be expressed as one large dynamic program. However, exact network optimization is practically impossible as a result of the large dimensionality of the resulting dynamic programming model. Therefore, various approximation methods have been developed, which are often either based on mathematical programming models or based on the decomposition of the network problem into a collection of single-leg problems. More recently, also approximations based on approximate dynamic programming are introduced. In the end, all these approximation methods are different approximations of the so-called optimal value function. Here, the optimal value function, which we will denote by $V_t(x)$, represents the optimal expected revenue as a function of the remaining capacity $x$ and time $t$. The approximation method may be static, however, then frequent re-optimization is needed in practice to account for changes in capacity $x$ and remaining time $t$. Next to producing an estimate of the value function, the approximation method must, more importantly, provide a good estimate of the displacement cost or bid prices. The definition of bid prices will be discussed later in this section. In Sections 3.2.1 and 3.2.2, we provide the relevant literature for, respectively, the mathematical programming and approximate dynamic programming models. Moreover, we discuss the decomposition models in Section 3.2.3 and provide extensions on previously introduced models in Section 3.2.4.

### 3.2.1 Mathematical programming models

The first mathematical programming approach to the network problem was given by Glover et al. (1982). In their work, they formulate the network problem as a large network flow problem using special side constraints. The problem could be solved by linear programming, however, they only considered deterministic demand and the model was only suitable if customers are indifferent between different routes for the same OD itinerary. Demand stochasticity was incorporated by Wollmer (1986), which proposes a binary programming formulation. Here, the binary decision variable represents a seat on a flight leg for an ODF itinerary and is given by $x_{ODF,i}$. The objective function consists of the sum of products of the decision variable $x_{ODF,i}$ and the expected marginal revenue of potentially selling the $i$th seat to the ODF itinerary. The resulting formulation leads to an enormous large problem, as we need to introduce a variable $x_{ODF,i}$ for every seat available to every ODF itinerary on every flight leg in the network. Nonetheless, Wollmer (1986) showed that the formulation can be solved relatively efficiently as a series of longest-path problems, though still impractical for large-scale networks. D'Sylva (1982) tries to make the LP formulation more manageable by using a piecewise linear approximation of each ODF's expected revenue curve. Dror et al. (1988) also formulated the network problem as a network flow problem, but now also incorporating cancellations and no-shows. When no "switch-over" or connections are allowed, the problem can easily be solved. However, incorporating connecting passengers between flight legs leads to additional constraints, making the model for large hub operations impractical to solve at that time.

A main disadvantage of the previously mentioned mathematical programming formulations is that they produce partitioned seat allocations and, therefore, cannot be used for day-by-day seat inventory control. However, they are still used in practice, as the dual variables of the capacity constraints can be used for bid price approaches, which we will discuss below further.

Williamson (1992) proposed and investigated linear and nonlinear programming approximations of the mathematical programming methods. Williamson (1992) also provided a detailed analysis of controls commonly used in practice. Quite surprisingly, she showed that the partitioned network seat allocations from deterministic optimization models consistently outperform those from more advanced probabilistic models when used as a basis of some sort of nested control strategy. This is caused by the fact that the partitioned probabilistic network solutions tend to 'overprotect' seats for higher fare class ODF itineraries. De Boer et al. (2002) argue that this phenomenon is caused by the fact that the booking process used in practice includes nesting of the fare classes, while this is ignored in the modeling phase. The probabilistic methods suffer more from this negligence than the deterministic methods. Curry (1990) proposed an optimization approach that combines a mathematical programming and nested allocation formulation, where fare classes within the same OD itinerary are nested. However, the capacity is partitioned among the OD itineraries, which makes his approach impractical for large networks, as for such networks we have a small number of seats allocated to each OD itinerary.

Using these network optimizations, booking limits can be implemented by using so-called network bid prices. For each flight leg, a bid price is calculated, which is the shadow price of the capacity constraint. Therefore, a bid price can be seen as a "cutoff" value for each flight leg, which can be used to decide whether to accept or reject an ODF itinerary request. Simpson (1989) was the first to introduce this concept, using a linear programming model of the network problem and using its optimal dual variables, which represent the bid prices. Williamson (1992) also analyzed bid price control in her research paper. Talluri and Van Ryzin (1998) recognized the lack of rigorous analysis of bid price control in previous work and studied the bid price control in depth. In their work, they proved that bid price control is not optimal in general, however, it performs well asymptotically. After their detailed analysis of bid prices, Talluri and Van Ryzin (1999) introduce a randomized linear programming model for the computation of network bid prices. Here, they introduce randomness into the deterministic approximation of Simpson (1989) and Williamson (1992) by using samples of ODF itinerary requests instead of using their expected value.

### 3.2.2 Approximate dynamic programming models

So far the bid prices are determined by solving static network models. The bid prices can also be determined by approximating the large dynamic program. Bertsimas and Popescu (2003) introduced such an algorithm to capture the dynamics of the booking process. In this algorithm, for each OD itinerary, the marginal value of the corresponding flight leg capacities is determined by taking the finite difference in the value functions. Adelman (2007) complements the research by Bertsimas and Popescu (2003) and tries to find time-dependent bid prices using a linear program by making an affine functional approximation to the optimal dynamic programming value function. Tong and Topaloglu (2014) improved the computational performance of the model of Adelman (2007) by proposing an approximate linear programming approach. Topaloglu (2009) takes another approach and computes dynamic bid prices depending on remaining capacity using Lagrangian relaxation. Both the model of Adelman (2007) and Topaloglu (2009) give tighter upper bounds on the optimal solution value than the deterministic linear programming model. Models giving tighter bounds are of great interest as empirical studies and practical experience show that these models generate more profitable controls. Talluri (2008) orders multiple upper bounds already established in the literature and also proposes new bounds. Topaloglu (2008) presents a stochastic approximation method in which the total expected revenue is visualized as a function of the bid prices. A good set of bid prices is found by using sample path-based derivatives.

### 3.2.3 Decomposition models

As already mentioned earlier, another way to tackle the network problem is to use an approximation method based on decomposing the problem into a collection of single-leg problems. These single-leg problems are solved independently, however, they typically incorporate some network information. An advantage of such a method over other approximation methods is that, as they solve the problem as independent single-leg problems, the resulting bid prices and displacement costs are typically dynamic in the sense that they are a function of remaining capacity and time. This is in contrast to network models, which should be solved repeatedly to determine the effects of changes in remaining capacity and time. Moreover, the simpler single-leg models allow for more realistic assumptions, such as stochastic dynamic demand, discrete demand and capacity and sequential decision-making. The main disadvantage of decomposition methods is that when decomposing the network problem into a collection of single-leg problems, we may lose valuable network information.

One of the first decomposition approaches is the virtual nesting approach developed by Smith et al. (1992) at American airlines. Virtual nesting uses single-leg nested-allocation controls, in which the classes used in the nested allocations are so-called virtual classes. These virtual classes group together sets of ODF itineraries using a certain flight leg through a process known as indexing. This process attempts to cluster ODF itineraries based on similar 'network value' and can be updated periodically to incorporate changes in network demand patterns. Based on the calculated nested protection levels for the virtual classes, a request for an ODF itinerary is accepted if and only if its corresponding virtual classes on each of the flight legs it requires are open. The main advantage of virtual nesting is that it incorporates network effects in the process of indexing, yet it preserves class-based booking controls of reservation systems at that time. Therefore, no major investments in IT infrastructure were needed, making it a quite popular method in practice. Nonetheless, over the years reservation systems and revenue management systems were upgraded, therefore also allowing bid price control. Talluri and Van Ryzin (2004b) state that in industry bid price control as means of network control is preferred as it is a simpler, more intuitive and powerful control method.

Bertsimas and De Boer (2005) viewed the virtual classes and the nested protection levels as a class of control strategies, parameterized by the protection levels on the network. They proposed a general framework to optimize an initial set of protection levels using simulation-based optimization and approximate dynamic programming methods. Using simulation, one can very accurately estimate the true network effects of changing protection levels. This is in contrast with traditional virtual nesting methods, which, at best, can only heuristically approximate network effects. In the optimization, Bertsimas and De Boer (2005) use a discrete-capacity, discrete-demand model of the network problem. However, their model can be computationally too intensive for large networks and the method lacks good convergence properties. Van Ryzin and Vulcano (2008) revisit the simulation-based optimization method described by Bertsimas and De Boer (2005) and propose a continuous capacity and demand model and use a faster and locally convergent algorithm.

Another decomposition method is the displacement-adjusted virtual nesting (DAVN) method introduced by Talluri and Van Ryzin (2004b). This approach applies the virtual nesting approach described previously. First, they solve a network heuristic model to obtain network static bid prices. Examples of a network heuristic model are the deterministic linear programming model (Glover et al. (1982); Wollmer (1986); Williamson (1992)), the probabilistic nonlinear programming model (Williamson (1992)) and the randomized linear programming model (Talluri and Van Ryzin (1999)). For the process of indexing, they calculate for all ODF itineraries using a specific flight leg the so-called displacement-adjusted revenue. That is, the revenue of the ODF itinerary on the specific flight leg is decreased by the static bid prices of the other flight legs used by this ODF itinerary. This is also known as prorating, for which more schemes exist, and where the goal is to approximate the net benefit of accepting a request for an ODF itinerary on a flight leg that is used by the ODF itinerary.

The performance of DAVN is, however, sensitive to the selection of the indexing method. Vinod (1989) introduced a DP-based indexing method at American Airlines, which significantly (about 0.5%) increased performance relative to their previous indexing scheme, which was based upon simple equal-revenue-width partitions. Moreover, when solving independent single-leg problems, one must note that the revenue-order assumption of the standard static single-leg model is likely to be violated. That is, it is very unlikely that demand for the virtual classes arrives in low-to-high displacement-adjusted revenue order, even if the original demand arrives in low-to-high revenue order.

Dynamic programming decomposition (DPD) also uses displacement-adjusted revenues to decompose the network, however, this approach leaves demand and revenue disaggregated. The single-leg problems are now solved as dynamic programs, therefore we do not have to worry about violating the revenue-order assumption of static single-leg models. As this approach is so similar to DAVN, the decision of which approach to use is often dictated by the desired control strategy. That is, if one implements virtual nesting control, then aggregating and indexing performed in DAVN closely match the control implementation. However, if one implements bid price control, then aggregation and indexing are not needed, and Talluri and Van Ryzin (2004b) show that DPD tends to give more accurate bid price approximations. Moreover, DPD has as output a bid price table, which is based on remaining time and capacity, therefore allowing better control. Zhang (2011) proposes a variant of the DPD incorporating customer choice. Zhang (2011) approximates the value function of the dynamic programming formulation of the network problem using a functional approximation that is nonlinear and non-separable.

Another closely related method to DAVN is the so-called iterative decomposition method. Both Williamson (1992) and R. L. Phillips (1994) propose variations to this idea. For consistency of the DAVN approach, the marginal costs produced by DAVN should match the original static bid price. Iterative DAVN feeds the estimated marginal cost back as new static bid prices if the two do not match and resolves the DAVN model. Convergence of the method is not guaranteed, as it is not known whether the mapping of the bid price vector onto itself is a contraction mapping. Iterative prorated EMSR is very similar to iterated DAVN, however, it uses another proration scheme. Bratu (1998) shows that, when demand is left disaggregated and the EMSR-b heuristic is used, the generated mapping is a contraction mapping, therefore convergence of this algorithm is guaranteed. Pseudo-code of both approaches is given by Talluri and Van Ryzin (2004b).

Birbil et al. (2014) propose an approximation method based on decomposition by origins and destinations. Their two-step approach consists of first determining optimal allocations of network capacities to OD itinerary pairs and, secondly, finding booking controls for different fare classes within each OD itinerary pair by solving a single-leg problem. In their computational study, they show that DPD still obtains higher average revenues than all considered strategies. Nonetheless, the work of Birbil et al. (2014) is more general, allowing for extensions such as customer choice models or robust optimization, and does not suffer from the computational burden of DPD. As a byproduct, Birbil et al. (2014) find an equivalent formulation of the famous deterministic linear programming model with considerably fewer variables. This formulation leads to a decrease in computation time and can, therefore, be interesting when frequently reoptimizing the deterministic linear programming model.

### 3.2.4 Extensions

More recent literature looks at robust optimization and customer choice models within network revenue management methods. For example, see the paper of An et al. (2021), which introduces a linear programming approach for robust network revenue management. In addition, recent literature focuses on the impact of strategic policies and business models on revenue management (e.g. (Lin et al., 2017) and (Alderighi et al., 2019)). Moreover, Calmon et al. (2021) analyze repeated customer interactions between the platform and a set of customers over a number of periods. Lastly, Chen et al. (2017) and Stein et al. (2020) consider advanced reservations.

## 3.3 Types of controls

Within seat inventory control, we can identify two underlying components. Namely, first, the seat allocation needs to be determined using an optimization algorithm and, second, these allocations need to be realized by the use of booking limits in such a way that the potential for increased revenues is secured. In this context, over the years a variety of combinations of optimization algorithms and control methodologies have been developed. In previous sections, we already shortly described some control mechanisms. In this section, we will more elaborately discuss the two most popular network controls, which are virtual nesting and bid price control. The overall robustness of the control scheme, the technological constraints imposed by the distribution system and the revenue performance achievable by the method play an important role in deciding which control to use.

American Airlines developed virtual nesting control as a hybrid of single-leg and network capacity control circa 1983. For each flight leg in the network, it uses single-leg nested-allocation control. The nested allocations consist of virtual classes, which group together sets of ODF itineraries that use a given flight leg. The process of grouping is known as indexing where one essentially uses a generated table to map every ODF itinerary to a virtual class on each flight leg. Here, the table can be generated periodically to account for changes in network demand patterns. The used booking policy is to accept a request for an ODF itinerary if all virtual classes on each flight leg corresponding to the ODF itinerary are available and to deny the request if any one of the virtual classes is closed.

The major advantage of this type of control was that it preserves the booking-class control logic of the used Global Distribution Systems (GDSs) at that time while allowing revenue management systems to incorporate some network information. That is, one can use the existing infrastructure of single-leg capacity control, yet incorporate network information. It has proven to be quite effective and virtual nesting control is, therefore, also popular in practice (Talluri and Van Ryzin, 2004b). Nonetheless, virtual nesting also has several noteworthy disadvantages. First of all, indexing introduces noise into the data and demand forecasts, specifically if the data is collected at the virtual class level. Namely, when reindexing, the virtual class demand statistics may shift dramatically, even when the underlying ODF itinerary demand is unchanged. In addition, the process of indexing brings additional complexity to the network problem. Lastly, virtual class demand can confuse analysts as it is not easily interpretable.

Bid price control provides a more intuitive, simpler and more powerful means of network capacity control. It was already popular in industries that were not using these legacy distribution systems. However, with current upgrades of GDS and revenue management systems, it has also gained popularity in the airline industry. With bid price control, we set a so-called bid price, which is a threshold value, for each flight leg in the network. It can be seen as an estimate of the marginal cost to the network of consuming the next incremental unit of the flight leg's capacity. The used booking policy is to only accept a request for an ODF itinerary if its corresponding revenue exceeds the sum of the bid prices of each flight leg required by the itinerary.

The first advantage of bid price control is its simple structure, namely only a single bid price is needed for each flight leg and the booking policy only requires a simple comparison of revenue to the sum of bid prices for the requested flight legs. It must, however, be noted that bid prices must be updated with changes in time and capacity to work properly. A second advantage is its intuitive functioning and its natural economic interpretation, namely the marginal cost to the network of the next incremental unit of flight leg capacity. Lastly, bid price control, if implemented correctly, has very good revenue performance and it can be proved to be near-optimal under certain conditions (Talluri and Van Ryzin, 2004b). Nonetheless, bid price control is by some criticized to be 'unsafe', due to its open/closed control philosophy (Williamson, 1992). Namely, if the revenue of a given ODF itinerary exceeds the bid price criteria, then the revenue management system will sell an unlimited amount of capacity for this ODF itinerary. However, this problem can be resolved by making the bid price a function of the remaining capacity. For a truly optimal system, revisions, requiring both reoptimization and reforecasting of demand, would be necessary on a real-time basis.

## 3.4 Overall comparison

Bid price control has become more and more the standard way to control the availability of capacity in a network environment due to current upgrades of GDS and revenue management systems. That is, it provides a simple, intuitive and powerful means of network capacity control. Nonetheless, due to the open/closed control philosophy of bid price control, the bid prices need to be dynamic, that is, it needs to be a function of remaining capacity. Therefore, only using a mathematical programming model, as described in Section 3.2.1, to optimize seat allocations, is less suitable as this only provides static bid prices. Although dynamic bid prices can be obtained theoretically by resolving the network problem repeatedly, computation issues make it unable to do so. Therefore, the remaining options to solve the network problem are approximate dynamic programming models and decomposition models.

A decomposition model seems to be in favor as it allows for more realistic assumptions and has proven to work well in practice and to be a good balance between industry standards and innovation. By solving the network problem as a collection of independent single-leg problems, the obtained bid prices are typically dynamic in the sense that they are a function of remaining capacity and or time. We can, therefore, counter the disadvantage of the open/closed control philosophy of static bid price control. Also, when solving the single-leg problems using a dynamic program, we do not have to violate the revenue-order assumption of the static single-leg models. Moreover, one can more easily incorporate demand variability in the single-leg problems. Lastly, when using a dynamic program to solve the single-leg problems, aircraft configuration selection can be accounted for. This is essential, as the obtained bid prices should factor in that in the business cabin the aisle seat or middle seat is left open when having rows of two or three chairs, respectively.

# 4 Methodology

First, in Section 4.1, we show that we can derive an optimal network seat inventory control by modeling the network problem as a stochastic dynamic programming model. However, solving this model is impractical due to the curse of dimensionality. For this reason, we need to find an approximation method to solve the network problem. We introduce four variants of a decomposition model as an approximation method in Section 4.2.

## 4.1 Optimal network seat inventory control

To determine the optimal decisions $\mathbf{u}^*(t, \mathbf{x}, \mathbf{r})$, we formulate the network seat inventory control problem as a stochastic dynamic program. A stochastic dynamic programming model can find an optimal decision vector by evaluating the entire state space and deciding at each point in time whether to accept a request for product $j$. This decision is represented by the decision vector $\mathbf{u}(t, \mathbf{x}, \mathbf{r})$. The states $\mathbf{S}$ of the system are defined by the remaining capacity $\mathbf{x}$, when the remaining time is $t$. Consequently, we represent the states by $\mathbf{S} = (t, \mathbf{x})$. We introduce $V_t(\mathbf{x})$, which is the maximum expected revenue to go given the remaining capacity $\mathbf{x}$, when the remaining time is $t$. Then $V_t(\mathbf{x})$ can be computed via the Bellman equation, which is given by Equation (6).

$$V_t(\mathbf{x}) = \max_{\mathbf{u}(t,\mathbf{x},\mathbf{r}) \in \mathcal{U}(\mathbf{x})} E\left[\mathbf{R}(t)^\intercal \mathbf{u}(t, \mathbf{x}, \mathbf{r}) + V_{t-1}(\mathbf{x} - \mathbf{A}^\intercal \mathbf{u}\,(t, \mathbf{x}, \mathbf{r}))\right] \tag{6}$$

for all $\mathbf{x} \in \mathcal{X}$, $t \leq T$, and with the boundary condition given by Equation (7).

$$V_0(\mathbf{x}) = 0 \text{ if } \mathbf{x} \in \mathcal{X} \tag{7}$$

Here, it is crucial to note that we assume sufficiently fine discretization of time, such that at most one request can arrive in a single time-period. Moreover, the boundary condition corresponds to the fact that when the aircraft departs and, therefore, when the remaining time $t = 0$, the remaining capacity is worth nothing as we cannot sell products anymore. Proposition 4.1 establishes both the existence and form of an optimal control policy.

**Proposition 4.1.** *If $R_j^t$ has a finite mean for all $t \leq T$ and products $j$, then $V_t(\mathbf{x})$ is finite for all $\mathbf{x} \in \mathcal{X}$, and an optimal control decision $\mathbf{u}^*(t, \mathbf{x}, \mathbf{r})$ exists and satisfies:*

$$u_j^*(t, \mathbf{x}, r_j) = \begin{cases} 1 & \textit{if } r_j \geq V_{t-1}(\mathbf{x}) - V_{t-1}(\mathbf{x} - \mathbf{A}_j) \textit{ and } \mathbf{x} - \mathbf{A}_j \in \mathcal{X}, \\ 0 & \textit{otherwise.} \end{cases} \tag{8}$$

*Proof.* $\mathbf{u}^*(t, \mathbf{x}, \mathbf{r})$ as defined by Equation (8) maximizes:

$$\mathbf{R}(t)^\intercal \mathbf{u}(t, \mathbf{x}, \mathbf{r}) + V_{t-1}(\mathbf{x} - \mathbf{A}^\intercal \mathbf{u}\,(t, \mathbf{x}, \mathbf{r}))$$

with $\mathbf{u}(t, \mathbf{x}, \mathbf{r}) \in \mathcal{U}(\mathbf{x})$. Consequently, we have that:

$$E\left[\mathbf{R}(t)^\intercal \mathbf{u}^*(t, \mathbf{x}, \mathbf{r}) + V_{t-1}(\mathbf{x} - \mathbf{A}\mathbf{u}^*\,(t, \mathbf{x}, \mathbf{r}))\right] \geq \max_{\mathbf{u}(t,\mathbf{x},\mathbf{r}) \in \mathcal{U}(\mathbf{x})} E\left[\mathbf{R}(t)^\intercal \mathbf{u}(t, \mathbf{x}, \mathbf{r}) + V_{t-1}(\mathbf{x} - \mathbf{A}\mathbf{u}\,(t, \mathbf{x}, \mathbf{r}))\right] \tag{9}$$

Therefore, if we can prove that the expectation of the left-hand side of Equation (9) exists, we know that $\mathbf{u}^*(t,\mathbf{x},\mathbf{r})$ satisfies the Bellman equation and, thereby, is an optimal control. We prove this by induction.

First, note that applying $\mathbf{u}^*(t,\mathbf{x},\mathbf{r})$ to the expectation of the left-hand side of Equation (9), gives:

$$\begin{aligned} &E\left[\mathbf{R}(t)^\intercal \mathbf{u}^*(t,\mathbf{x},\mathbf{r}) + V_{t-1}(\mathbf{x} - \mathbf{A}\mathbf{u}^*(t,\mathbf{x},\mathbf{r}))\right] \\ &= V_{t-1}(\mathbf{x}) + \sum_{j:(\mathbf{x}-\mathbf{A}_j)\in\mathcal{X}} E\left[R_j^t - V_{t-1}(\mathbf{x}) + V_{t-1}(\mathbf{x}-\mathbf{A}_j)\right]^+ \end{aligned} \tag{10}$$

We apply induction on $t$. Note that $V_0(\mathbf{x}) = 0$ for all $\mathbf{x} \in \mathcal{X}$ and together with the assumption of finite mean of $R_j^t$ for all $t \leq T$ and products $j$ and the fact that $E\left[X - a\right]^+ \leq E\left[X\right] + |a|$, we know that Equation (10) is finite for the base case.

As induction hypothesis, let $V_{t-1}(\mathbf{x})$ be finite for all $\mathbf{x} \in \mathcal{X}$. Therefore, the term $-V_{t-1}(\mathbf{x}) + V_{t-1}(\mathbf{x} - \mathbf{A}_j)$ is finite and together with the assumption of finite mean of $R_j^t$ for all $t \leq T$ and products $j$ and the fact that $E\left[X - a\right]^+ \leq E\left[X\right] + |a|$, we know that Equation (10) is finite for time $t$.

Consequently, by the principle of induction on $t$, we know that the expectation of the left-hand side of Equation (9) exists for all $\mathbf{x} \in \mathcal{X}$ and $t \leq T$ and, thereby, that $\mathbf{u}^*(t,\mathbf{x},\mathbf{r})$ exists and is of the form given by Equation (8). □

The optimal control policy given by Equation (8) states that we should accept a request for product $j$ with corresponding revenue $r_j$ if and only if we have sufficient remaining capacity and the revenue is greater than or equal to the opportunity cost of accepting the request for product $j$. Here, the opportunity cost $OC_t^j(\mathbf{x})$ of a product $j$ at time $t$ is defined as Equation (11).

$$OC_t^j(\mathbf{x}) = V_{t-1}(\mathbf{x}) - V_{t-1}(\mathbf{x} - \mathbf{A}_j) \tag{11}$$

Moreover, aircraft configuration selection is implicitly taken into account by evaluating all $\mathbf{x} \in \mathcal{X}$, and thereby evaluating all possible configurations, and only accepting a request for product $j$ if and only if $\mathbf{x} - \mathbf{A}_j \in \mathcal{X}$.

Unfortunately, solving this stochastic dynamic programming model exactly is practically impossible due to the curse of dimensionality. Even if aircraft configuration selection is not considered, the state space becomes the Cartesian product of the capacities in the network. Therefore, if we have a relatively small network with $m = 25$ flight legs and a capacity of 142 seats with fixed configuration on each flight leg, we have a state space of $142^{25}$ states. Due to this reason, various approximation methods have been developed, which are often based on decomposition or mathematical programming methods. Also, algorithms based on approximate dynamic programming are introduced recently. Here, the balance between efficiency and quality of the approximation is key.

## 4.2 Decomposition model

A decomposition model is chosen as it has been proven to work well in practice and as it is a good balance between industry standards and innovation. The decomposition model consists of two parts, namely, first, we need to decompose the network problem into a collection of single-leg problems. Here, the goal is to incorporate network information into the single-leg problems. Then, second, we need to solve the independent single-leg problems and implement a way to control the availability of capacity. In Section 4.2.1, we introduce a deterministic mixed integer linear programming model, which can be used to decompose the network problem. Moreover, in Section 4.2.2, we propose three alternatives to decompose the network problem. In Section 4.2.3, we introduce a dynamic programming model to solve the single-leg problems and introduce a dynamic bid price control. Lastly, in Section 4.2.4, we summarize the four proposed approaches.

### 4.2.1 Decomposition by deterministic mixed integer linear programming model

Most decomposition models introduced in literature use network static bid prices to decompose the network problem into a set of independent single-leg problems. These network static bid prices can be obtained by various network heuristic models, such as the ones described in Section 3.2.1. The network static bid prices take into account congestion within the network and are used to approximate the net benefit of a product $j$ on a flight leg $i$. This process is also known as prorating in literature.

Quite surprisingly, Williamson (1992) showed that a simpler deterministic optimization model, which is based on average demands, consistently outperforms the more advanced probabilistic models when used as a basis of some sort of nested control strategy. We propose an extension to the deterministic linear programming (DLP) model to incorporate aircraft configuration selection. Using the variables and definitions proposed in Chapter 2, the original DLP model is given by Equation (12)-(15).

Deterministic linear programming (dlp) model

$$V^{DLP}(\mathbf{x}, \boldsymbol{\mu}(t)) = \max \mathbf{r}^T \mathbf{y} \tag{12}$$

$$\text{s.t. } \mathbf{A}^T \mathbf{y} \leq \mathbf{x} \tag{13}$$

$$\mathbf{0} \leq \mathbf{y} \leq \boldsymbol{\mu}(t) \tag{14}$$

$$\mathbf{y} \in \mathbb{R}^n \tag{15}$$

Here, the decision variables $\mathbf{y} = \left[y_1, ..., y_n\right]^\intercal$ represent the partitioned fractional allocation of capacity for each of the $n$ products. The result of the approximation is an optimal partitioned fractional allocation in the case when demand was deterministic and equal to its mean $\boldsymbol{\mu}(t)$.

We will now extend the DLP model to incorporate up- and downgrading of customers. As the vector of mean aggregate demand to come at time $t$, $\boldsymbol{\mu}(t)$, and the partitioned fractional allocation of capacity, $\mathbf{y}$, is continuous, we also relax the upgrade vector $\mathbf{g}$. The resulting model is named the extended deterministic linear programming (EDLP) model and is given by Equation (16)-(20).

Extended deterministic linear programming (edlp) model

$$
\begin{aligned}
V^{EDLP}(\mathbf{x}, \boldsymbol{\mu}(t)) = \quad & \max \mathbf{r}^T\mathbf{y} && (16)\\
& \text{s.t. } \mathbf{A}^T\mathbf{y} - \mathbf{C}^T\mathbf{g} \leq \mathbf{x} && (17)\\
& \mathbf{0} \leq \mathbf{y} \leq \boldsymbol{\mu}(t) && (18)\\
& -\mathbf{d} \leq \mathbf{g} \leq \mathbf{f} && (19)\\
& \mathbf{y} \in \mathbb{R}^n, \mathbf{g} \in \mathbb{R}^m && (20)
\end{aligned}
$$

Again, the decision variables $\mathbf{y} = \left[y_1, ..., y_n\right]^\intercal$ represent the partitioned fractional allocation of capacity for each of the $n$ products. Moreover, the decision variables $\mathbf{g} = \left[g_1, ..., g_m\right]^\intercal$ describe the number of upgrades for each of the $m$ flight legs. To incorporate aircraft configuration selection, we can extend the EDLP model. Now, the resulting model becomes a deterministic mixed integer linear programming (DMILP) model, given by Equation (21)-(26).

Deterministic mixed integer linear programming (dmilp) model

$$
\begin{aligned}
V^{DMILP}(\mathbf{x}, \boldsymbol{\mu}(t)) = \quad & \max \mathbf{r}^T\mathbf{y} && (21)\\
& \text{s.t. } \mathbf{A}^T\mathbf{y} - \mathbf{C}^T\mathbf{g} \leq \mathbf{x} + \mathbf{B}^T\mathbf{z} && (22)\\
& \mathbf{0} \leq \mathbf{y} \leq \boldsymbol{\mu}(t) && (23)\\
& -\mathbf{d} \leq \mathbf{g} \leq \mathbf{f} && (24)\\
& \mathbf{z} < \mathbf{c} && (25)\\
& \mathbf{y} \in \mathbb{R}^n, \mathbf{g} \in \mathbb{R}^m, \mathbf{z} \in \mathbb{Z}^m_{0+} && (26)
\end{aligned}
$$

Decision variables $\mathbf{y} = \left[y_1, ..., y_n\right]^\intercal$ and $\mathbf{g} = \left[g_1, ..., g_m\right]^\intercal$ have the same interpretation as for the EDLP model. The decision variables $\mathbf{z} = \left[z_1, ..., z_m\right]^\intercal$ now describe the aircraft configuration on each of the $m$ flight legs. Note that the DMILP model can easily be extended to include cancellations and no-shows in the underlying linear programming formulation.

The optimal partitioned fractional allocation of the DMILP model can, however, not be used directly as seat inventory control. For example, the partition is fractional and, therefore, it is undefined what to do with the fractional part. Moreover, many carriers use a nested inventory structure instead of a partitioned fare class inventory structure. That is, for a nested inventory structure, each low-fare class is nested within the next higher-fare class at the flight leg level. In this way, as long as seats are available, higher revenue requests will not be denied. Williamson (1992) showed that the expected revenue of a nested inventory is greater than or equal to the expected revenue of a partitioned inventory. It is also easy to reason that it will not be profitable to reject higher revenue requests when seats are originally located to lower revenue requests. The problem with the found optimal partitioned fractional allocations is that these allocations are generally not optimal seat allocations for a nested structure. Moreover, the network environment makes nesting complicated, for example, the highest revenue ODF itinerary may not be the most desirable itinerary for the network. Namely, if there is high local demand, generally the local ODF itineraries are more desirable than the higher revenue multi-leg ODF itineraries.

A way to tackle this problem traditionally is to solve the DLP model and use the optimal dual variables associated with Constraints (13), as network bid prices. However, now, we need to solve a mixed integer linear programming model for which linear programming duality does not hold. Nonetheless, one must note that the optimal dual variables are the shadow prices of the capacity constraints and, thereby, an approximation of the opportunity cost of the corresponding flight leg-cabin capacities. We, therefore, propose to calculate the opportunity cost of a product (and thereby of a set of flight leg-cabin capacities) directly. These opportunity costs can then be used as network bid prices. An additional advantage is that these opportunity costs can also be used to support other decision-making, such as pricing and capacity planning.

The most noticeable work that is related to this approach is given by Bertsimas and Popescu (2003). Bertsimas and Popescu (2003) introduced a new control policy, which they called certainty equivalent control (CEC). They showed that CEC leads to higher revenue and more robust performance than an additive bid price control. Within CEC, the opportunity cost of a cabin-specific OD itinerary is calculated by the finite differences in the value functions. Here, the value function of the stochastic dynamic program is approximated by the value of the DLP model. A request for an ODF request is accepted if and only if the fare exceeds the current opportunity cost estimate of the cabin-specific OD itinerary and we have sufficient remaining capacity.

Bertsimas and Popescu (2003) noted two main disadvantages of using leg-based additive bid prices when obtained from solving the linear programming formulation of the problem. First, the bid prices are not well defined if there are multiple dual solutions. That is, the bid prices are not unique if the optimal solution is degenerate. This can lead to inconsistencies between different optimization runs. Secondly, the bid prices may not be completely additive due to 'bundle effects'. That is, we may have basis changes in the dual LP for multi-leg or group OD requests. CEC does not have an additive structure and is well-defined when having multiple dual solutions. Example 4.1, which is based on an example of Talluri and Van Ryzin (2004b), illustrates the two main disadvantages of bid price control.

**Example 4.1.** *Consider a network with two flight legs ($m = 2$) and the airline sells three products ($n = 3$). Each flight leg only has a single cabin. There are two time-periods ($T$=2) and there is a capacity of one on each of the flight legs. The problem data is given in Table 1.*

Table 1: Problem data of a small network of two flight legs and three products as a counterexample for bid price control.

| Period $t$ | Product $j$ | $\mathbf{A}_j$ | Associated revenue (€) | Probability $p_j^t$ |
|---|---|---|---|---|
| 2 | 1 | $(1,0)^T$ | 500 | 0.3 |
| | 2 | $(0,1)^T$ | 500 | 0.3 |
| | 3 | $(1,1)^T$ | 1000 | 0.4 |
| 1 | 3 | $(1,1)^T$ | 1000 | 0.8 |
| | No arrival | | | 0.2 |

*We have at most one request per time-period, so requests in each period are mutually exclusive. It is easy to notice by inspection that an optimal control would deny requests for products 1 and 2 and accept a request for product 3 in both periods. Namely, by accepting a request for product 1 or 2, we incur an opportunity cost of €800 (0.8×€1000) by not being able to accept a request for product 3 in period 1 anymore. This is greater than the incurred revenue of accepting a request for product 1* *or 2 in period 2, namely €500, therefore, we will not accept requests for product 1 or 2 in period 2. We also want to accept a request for product 3 in period 2, as we never can capture more revenue in period 1.*

*We, therefore, need that the bid prices* $\pi_1$ *and* $\pi_2$ *for flight legs 1 and 2, respectively, need to satisfy that* $\pi_1 > 500$, $\pi_2 > 500$ *and* $\pi_1 + \pi_2 \leq 1000$, *which is, obviously, impossible. Therefore, no bid price control can produce an optimal control.*

*It is not hard to see that the best a bid price control can do, is to deny all requests in period 2 and to accept a request for product 3 in period 1. This yields an expected revenue of €800 (0.8×€1000). This is in contrast with an optimal control, which has an expected revenue of €880 (0.4×€1000 + 0.6×0.8 ×€1000). Therefore, an optimal control can obtain on average 10% more revenue than the best bid price control.*

So in Example 4.1, we observe that the opportunity cost of using both flight legs simultaneously is exactly equal to the opportunity cost of using a single flight leg. This can be explained by the fact that the expected revenue to go is determined by the minimum available capacity of the two flight legs. Therefore, the revenue to go is highly nonlinear with displaced capacities, something that bid price control fails to capture. This is analogous to the degeneracy problem explained previously. Also, Example 4.1 reflects the problem that bid prices may not be completely additive due to 'bundle effects'. That is, in general, large relative changes in remaining capacity on several flight legs simultaneously cannot be expected to give the same revenue outcomes as the sum of the individual changes.

In our case, we can determine the opportunity cost of flight leg-cabin combination $i$ and the opportunity cost of product $j$ by Equation (27) and (28), respectively. In Equation (27), $\mathbf{e}_i$ denotes the unit vector with entry one for the flight leg-cabin combination $i$. Essentially, Equation (27) is a special case of Equation (28), namely the opportunity cost of a flight leg-cabin combination is the same as the opportunity cost of a product using only this single flight leg and cabin. However, for notation convenience, the two are given separately.

$$OC_i(\mathbf{x}) = V^{DMILP}(\mathbf{x}, \boldsymbol{\mu}(t)) - V^{DMILP}(\mathbf{x} - \mathbf{e}_i, \boldsymbol{\mu}(t)) \tag{27}$$

$$OC_j(\mathbf{x}) = V^{DMILP}(\mathbf{x}, \boldsymbol{\mu}(t)) - V^{DMILP}(\mathbf{x} - \mathbf{A}_j, \boldsymbol{\mu}(t)) \tag{28}$$

Example 4.2 continues Example 4.1 and shows that this framework does not suffer from not completely additive bid prices due to 'bundle effects' and degeneracy problems.

**Example 4.2.** *As we only have one cabin and fixed aircraft configuration for this example, one can use the DLP model instead of the DMILP model and, thereby, find the opportunity cost of product* $j$ *by Equation* (29).

$$OC_j(\mathbf{x}) = V^{DLP}(\mathbf{x}, \boldsymbol{\mu}(t)) - V^{DLP}(\mathbf{x} - \mathbf{A}_j, \boldsymbol{\mu}(t)) \tag{29}$$

*Using the probabilities given in Table 1, the vector of mean aggregate demand to come at remaining time* $t = 2$ *can be given by* $\boldsymbol{\mu}(2) = \begin{bmatrix}0.3, 0.3, 1.2\end{bmatrix}^\intercal$. *The DLP model at remaining time* $t = 2$ *can now be written as Equation* (30)-(36).

$$V^{DLP}(\mathbf{x}, \boldsymbol{\mu}(t)) = \quad max\ 500y_1 + 500y_2 + 1000y_3 \tag{30}$$
$$s.t.\ y_1 + y_3 \leq 1 \tag{31}$$
$$y_2 + y_3 \leq 1 \tag{32}$$
$$0 \leq y_1 \leq 0.3 \tag{33}$$
$$0 \leq y_2 \leq 0.3 \tag{34}$$
$$0 \leq y_3 \leq 1.2 \tag{35}$$
$$\mathbf{y} \in \mathbb{R}^3 \tag{36}$$

*Solving the DLP model gives the optimal dual values of Constraint (31) and (32) of both €500. Therefore, when using the optimal dual variables as network bid prices, the bid prices $\pi_1$ and $\pi_2$ at remaining time $t = 2$ will be given by $\pi_1 = \pi_2 =$ €500. This results in a control that accepts all incoming requests in period 2 and is, therefore, non-optimal. The expected revenue of this control is equal to €700 (0.3×€500 + 0.3×€500 + 0.4×€1000). Calculating the opportunity costs of products $j$ by Equation* (29) *results in an opportunity cost of €850, €850 and €1000 for products 1, 2 and 3, respectively. By only accepting an incoming request with an associated revenue equal to or higher than the opportunity cost of this incoming request, a request for product 1 or 2 will be rejected and a request for product 3 will be accepted. The expected revenue will, consequently, be equal to €880 (0.4×€1000 + 0.6×0.8 ×€1000), which is optimal and 25.7% more than the expected revenue of the control using bid prices based on the optimal dual variables of the capacity constraints. Here, as product 1 uses only flight leg 1 and product 2 uses only flight leg 2, the opportunity cost of flight legs 1 and 2 is the same as the opportunity cost of products 1 and 2. Clearly, the opportunity cost of using both flight legs (€1000) is not equal to the sum of the opportunity cost of flight leg 1 and flight leg 2 (€850 + €850 = €1700). This shows that optimal bid prices are not completely additive due to 'bundle effects'.*

Within CEC, the opportunity cost of a product $j$ is calculated on a real-time basis. That is, if a request for a product $j$ comes in, the opportunity cost of product $j$ is calculated given the remaining capacities and a product is sold if and only if the fare of the product exceeds the opportunity cost and we have sufficient remaining capacity. However, the deterministic mixed integer linear programming model is computationally too expensive to compute on a real-time basis. Moreover, using CEC requires airlines to make major adjustments in their revenue management systems, which are often based on bid price control.

We, therefore, propose to calculate the opportunity costs of all products and flight leg-cabin combinations nightly given the remaining capacity at that time. In this way, the computation time may not be a limiting factor. We then decompose the network problem by prorating the revenues of the products using the opportunity costs, thereby taking into account network effects. The goal of proration is to have a measure of the contribution of a multi-leg ODF itinerary to each flight leg the multi-leg ODF itinerary uses. Then, using the prorated revenues, we can solve the network problem as a collection of independent single-leg problems. An example of such a proration is to calculate for each flight leg-cabin combination $i$ used by product $j$ the prorated revenue $\bar{r}_{ij}$ of product $j$ by Equation (37).

$$\bar{r}_{ij} = r_j \frac{OC_i}{OC_j}, i \in A_j. \tag{37}$$

It must be noted that one can prorate the revenues also in other manners using the calculated opportunity costs. Finding the best way to prorate is, however, a subtle task that goes beyond the scope of this work. Also, note that the overall revenue is not preserved for the given proration. That is, if for a product $j$ we have that $OC_j \neq \sum_{i \in A_j} OC_i$, then $r_j \neq \sum_{i \in A_j} \bar{r}_{ij}$. Nonetheless, this is as expected, as if the opportunity cost of a product $j$ is smaller (greater) than the sum of the opportunity costs of the used flight legs, it means that the multi-leg product $j$ is (not) preferred over the single-leg products. Nonetheless, as the overall revenue is not preserved, it is important to use the prorated revenues within the single-leg models when deciding to accept or reject a request for an ODF itinerary, as these prorated revenues are also used to determine the bid prices.

It also must be noted that the opportunity costs of products $j$ using the same cabin-specific OD itinerary are identical. Also, if a product $j$ uses only a single flight leg, the opportunity cost of the cabin-specific product equals the opportunity cost of the cabin-specific flight leg capacity. Therefore, in the end, the opportunity cost only needs to be calculated for all cabin-specific OD itineraries used by the products $j$. Also, by 'warm starting' the integer linear programming solver with the previously determined solution for $\mathbf{x}$, one can reduce computation time significantly. Lastly, computation time could be reduced by computing the opportunity costs in parallel.

#### 4.2.2 Decomposition by relaxed deterministic mixed integer linear programming model

To reduce the computation time significantly of the previously introduced framework, one could relax the aircraft configuration $\mathbf{z}$ in the DMILP model. The relaxed DMILP model then becomes a linear programming model, which can be solved significantly faster than a mixed integer linear programming model. This also allows us to use the optimal dual variables associated with Constraints (22) directly as network bid prices. In this way, one needs to compute the relaxed DMILP model only once, which can be beneficial if computation time is limited.

When the optimal dual variables are used directly as network bid prices, a similar proration as Equation (37) can be performed. However, we now use leg-based additive bid prices instead of opportunity costs to prorate. The proration scheme is given by Equation (38). That is, the prorated revenue $\bar{r}_{ij}$ of product $j$ on each flight leg-cabin combination $i$ used by product $j$ is calculated by Equation (38). Here, $\pi_i$ denotes the bid price of flight leg-cabin combination $i$.

$$\bar{r}_{ij} = r_j \frac{\pi_i}{\sum_{l \in A_j} \pi_l}, i \in A_j. \tag{38}$$

Relaxing the aircraft configuration $\mathbf{z}$ can be justified by the fact that the network bid prices are only used for the proration. The purpose of the proration step to incorporate network information into the single-leg problems is still assured. Moreover, by incorporating aircraft configuration selection within the dynamic programming model, which is used to solve the single-flight legs, one can still account for the fact that the cabin divider only segregates entire rows. Additionally, when we did not sell many products yet and the remaining flight leg capacities $\mathbf{x}$ are still large, we can expect that the aircraft configuration selection will not be a limiting factor. Therefore, it might also be an option to only relax the aircraft configuration $\mathbf{z}$ when the remaining time $t$ is large, as we then expect that the flight leg capacities $\mathbf{x}$ are still large.

#### 4.2.3 Solving independent single-leg problems using dynamic programming model

We formulate the single-leg problem as a stochastic dynamic program, similar to the stochastic dynamic program of the network seat inventory control problem in Section 4.1. A dynamic program formulation has been chosen as it allows us to use more realistic assumptions. That is, compared to static models, we can relax the revenue-order assumption. This assumption is likely to be violated as it is very unlikely that demand arrives in low-to-high prorated revenue order, even if the original demand arrives in low-to-high revenue order. However, more importantly, by modeling the single-leg problem as a dynamic program, we can account for aircraft configuration selection in our optimization. Then, the decomposition model can become flight leg based instead of flight leg-cabin based. That is, we can obtain dynamic bid prices which are a function of the cabin-specific flight leg capacities.

Dynamic models do, however, require the assumption of Markovian arrivals, such as Poisson arrivals, to make them tractable. Nonetheless, Williamson (1992) shows that modeling the arrival of demand by a Poisson process is a valid assumption, as this process can model the underlying pattern of airline demand realistically. That is, the Poisson distribution is naturally truncated at zero with significant positive skewness for low mean demands. Also, the Poisson process is a discrete distribution, thereby being more realistic than continuous distributions, such as the normal and gamma distribution, which also tend to fit statistical demand data well. An additional advantage of the Poisson process is the property that the sum of two independent Poisson processes remains Poisson distributed, which is handy for simulation purposes. Moreover, the mean and variance of a Poisson process are equal to each other. Research by A. O. Lee (1990) shows that this property is a reasonable assumption for modeling the arrival of demand. Lastly, a dynamic model requires an estimate of the pattern of arrivals over time, which is called the booking curve. We assume a linear booking curve, that is, the probability of an arrival of a request for an ODF itinerary is constant over time.

Similar to Section 4.1, the stochastic dynamic programming model can find an optimal decision vector by evaluating the entire state space and deciding at each point in time whether to accept a request for product $j$. However, now, as we are only considering a single flight leg, the state space becomes manageable. The states $\mathbf{S}$ of the system are defined by the remaining capacity $\mathbf{x}$, when the remaining time is $t$. Consequently, we represent the states by $\mathbf{S} = (t, \mathbf{x})$. As we only look at a single flight leg, the remaining capacity vector $\mathbf{x}$ is given by $\mathbf{x} = [x^M, x^C]^T$. Time $t$ must be discretized properly to satisfy the assumption of at most one arrival per time-period. That is, it must hold that $\sum_{j=1}^{n} p_j^t \leq 1 \ \forall \ t \leq T$. Moreover, let $V_t(\mathbf{x})$ now denote the maximum expected prorated revenue to go given the remaining capacity $\mathbf{x}$, when the remaining time is $t$. Then $V_t(\mathbf{x})$ can be computed via the Bellman equation given by Equation (39), where we used the optimal control policy found in Section 4.1.

$$\begin{aligned} V_t(\mathbf{x}) &= V_{t-1}(\mathbf{x}) + \sum_{j:(\mathbf{x}-\mathbf{A}_j)\in\mathcal{X}} E\left[R_j^t - V_{t-1}(\mathbf{x}) + V_{t-1}(\mathbf{x}-\mathbf{A}_j)\right]^+ \\ &= V_{t-1}(\mathbf{x}) + \sum_{j:(\mathbf{x}-\mathbf{A}_j)\in\mathcal{X}} p_j^t \left\{\bar{r}_j - V_{t-1}(\mathbf{x}) + V_{t-1}(\mathbf{x}-\mathbf{A}_j)\right\}^+ \\ &= V_{t-1}(\mathbf{x}) + \sum_{j:(\mathbf{x}-\mathbf{A}_j)\in\mathcal{X}} p_j^t \left\{\bar{r}_j - OC_t^j(\mathbf{x})\right\}^+ \end{aligned} \tag{39}$$

for all $\mathbf{x} \in \mathcal{X}$, $t \leq T$, and with the boundary condition given by Equation (40)

$$V_0(\mathbf{x}) = 0 \text{ if } \mathbf{x} \in \mathcal{X} \tag{40}$$

and where the opportunity cost of product $j$ at time $t$, $OC_t^j(\mathbf{x})$, is defined by Equation (41).

$$OC_t^j(\mathbf{x}) = V_{t-1}(\mathbf{x}) - V_{t-1}(\mathbf{x} - \mathbf{A}_j) \tag{41}$$

Here, the prorated revenue of product $j$, $\bar{r}_j$, is equal to $\bar{r}_{ij}$ if we consider flight leg $i$. Moreover, in the single-leg variant, product $j$ either consumes an economy or a business seat. Therefore, for each state, we only need to calculate the opportunity cost of an economy or a business seat, which is given by Equations 42 and 43, respectively.

$$OC_t^M(\mathbf{x}) = V_{t-1}(\mathbf{x}) - V_{t-1}\left(\mathbf{x} - \begin{bmatrix}1\\0\end{bmatrix}\right) \tag{42}$$

$$OC_t^C(\mathbf{x}) = V_{t-1}(\mathbf{x}) - V_{t-1}\left(\mathbf{x} - \begin{bmatrix}0\\1\end{bmatrix}\right) \tag{43}$$

Now, we have that $OC_t^j(\mathbf{x}) = OC_t^M(\mathbf{x})$ if product $j$ uses an economy seat and $OC_t^j(\mathbf{x}) = OC_t^C(\mathbf{x})$ if product $j$ uses a business seat. Aircraft configuration selection is again implicitly taken into account by evaluating all $\mathbf{x} \in \mathcal{X}$, thereby evaluating all possible configurations, and only accepting a request for product $j$ if and only if $\mathbf{x} - \mathbf{A}_j \in \mathcal{X}$. The optimal control decision $\mathbf{u}^*(t, \mathbf{x}, \mathbf{r})$ should satisfy:

$$u_j^*(t, \mathbf{x}, \bar{r}_j) = \begin{cases} 1 & \text{if } \bar{r}_j \geq OC_t^j(\mathbf{x}) \text{ and } \mathbf{x} - \mathbf{A}_j \in \mathcal{X}, \\ 0 & \text{otherwise.} \end{cases} \tag{44}$$

In other words, an optimal control policy should accept a request for product $j$ with corresponding prorated revenue $\bar{r}_j$ if and only if this revenue is greater than or equal to the opportunity cost of accepting the request for product $j$ and we have sufficient remaining capacity. This optimal control policy can be implemented using a bid price control, namely, we set the cabin-specific bid price equal to the cabin-specific opportunity cost. That is, the bid price of the economy and business cabin seat at remaining capacity $\mathbf{x}$ is given by Equations 45 and 46, respectively.

$$\pi_t^M(\mathbf{x}) = OC_t^M(\mathbf{x}) = V_{t-1}(\mathbf{x}) - V_{t-1}\left(\mathbf{x} - \begin{bmatrix}1\\0\end{bmatrix}\right) \tag{45}$$

$$\pi_t^C(\mathbf{x}) = OC_t^C(\mathbf{x}) = V_{t-1}(\mathbf{x}) - V_{t-1}\left(\mathbf{x} - \begin{bmatrix}0\\1\end{bmatrix}\right) \tag{46}$$

Note that the shown nonoptimality of a bid price control by Example 4.1 in Section 4.2.1 does not hold here, as in Example 4.1 we look at a network of flight legs instead of one single flight leg. Moreover, note that for the optimal control policy, we obtain dynamic bid prices, that is, they are a function of both remaining time $t$ and remaining capacity $\mathbf{x}$.

As stated previously, to solve the single-leg problem and to obtain the dynamic bid prices, we evaluate the entire state space. We start at the boundary condition given by Equation (40) and use the recursion given by Equation (39) to proceed backward in time $t$. For each stage $t$, we first construct set $\mathcal{X}$ as defined by Equation (2) in Chapter 2 for the specific aircraft type assigned to the flight leg and given the remaining capacity vector $\mathbf{x}$. Here, set $\mathcal{X}$ defines all feasible remaining flight leg capacities $\mathbf{x}$. The size of set $\mathcal{X}$ is $\mathcal{O}(C^2)$, where $C$ represents the total flight leg capacity, that is $C = X^M + X^C$. If a request for product $j$ results in infeasible remaining flight leg capacities $\mathbf{x}_{new} = \mathbf{x} - \mathbf{A}_j$, that is $\mathbf{x}_{new} \notin \mathcal{X}$, then $OC_t^{cabin}(\mathbf{x})$ is set to positive infinity for all $t$.

The set $\mathcal{X}$ for the specific remaining flight leg capacities $\mathbf{x}$ can be reused for different optimization runs and flight legs using the same aircraft type. This is, therefore, a one-time calculation and is consequently not taken into account for the complexity analysis of the dynamic programming algorithm. For each stage $t$, we have $\mathcal{O}(nC^2)$ computations, therefore, the overall computational complexity of the dynamic programming model is $\mathcal{O}(nC^2T)$. Moreover, as the periods are chosen such that we have $\mathcal{O}(1)$ requests per period and for most flight legs, the total expected demand is of the same magnitude as the total capacity $C$, we can approximate $T$ by $\mathcal{O}(C)$. Therefore, the overall computational complexity can also be represented as $\mathcal{O}(nC^3)$, which is pseudopolynomial in the input size.

Lastly, note that the independent nature of the single-leg problems makes computation very suitable for parallel computation. That is, in theory, if the hardware allows us to use this many threads, the single-leg problems can be solved at the same time as one single-leg problem.

#### 4.2.4 Overall summary of the different approaches

As previously stated, the decomposition model consists of two parts, namely the decomposition step and solving the independent single-leg problems. For the decomposition step, we have suggested four different approaches. The first approach uses the framework as described in Section 4.2.1 as the decomposition step. We call this approach the Opportunity Cost based Decomposition - Dynamic Programming (OCD-DP) approach. Then, for the second approach, we relax the aircraft configuration $\mathbf{z}$ in the DMILP model and for the remainder use the same framework as for the first approach. We call this approach the Relaxed Opportunity Cost based Decomposition - Dynamic Programming (ROCD-DP) approach. For the third approach, we also relax the aircraft configuration $\mathbf{z}$ in the DMILP model, however, we now use the optimal dual variables associated with Constraint (22) directly as network bid prices. Therefore, this approach is called the Dual based Decomposition - Dynamic Programming (DD-DP) approach. Depending on the results of the first, second and third approaches, it might be beneficial to look at a fourth approach where the aircraft configuration $\mathbf{z}$ is only relaxed when the remaining time $t$ is large. Depending on the results of the second and third approaches, the fourth approach will use the opportunity cost or the optimal dual variables corresponding to Constraint (22) as network bid prices.

Note that all four different approaches use the same dynamic programming model to solve the independent single-leg problems. Therefore, all four approaches obtain dynamic bid prices, which are a function of both the remaining time $t$ and remaining capacity $\mathbf{x}$. Lastly, due to the dependencies of the fourth approach on the results of the first three approaches, we consider the fourth approach as a topic for further research. Therefore, in the next chapter, we will test the first three approaches.

# 5 Results

First, in Section 5.1, a simulation of the booking process is introduced, which is used to evaluate the newly proposed approaches. Then, in Section 5.2, we introduce two benchmark solution methods and an approximate upper bound on the realized revenue, which we use to show whether our proposed approaches yield competitive results. Next, in Section 5.3, we introduce several instances and experiments and we show our computational results.

## 5.1 Simulation of the booking process

To evaluate a newly proposed method, it is important to look at implementation difficulties within the current reservation system. Moreover, one needs to consider the complexities of the actual optimization and the control mechanism. In addition, it is important to understand the ins and outs of the method and the underlying assumptions on which the method is built.

Nonetheless, the driving force behind developing a new method is to capture additional revenue. Therefore, a measure is needed to quantify the potential increase in total revenue for the airline. A simulation of the booking process of the airline can provide us with such a measure. Moreover, it allows us to separate the impact of the many factors which play a role in network seat inventory control, such as competition, control policies and the usage of different optimization algorithms. Lastly, a simulation that realistically can model the booking process, allows us to refine the method before implementing it in the airline's reservation system.

The implemented booking process simulation is a so-called Monte Carlo simulation. The simulation has as input the definition of the network, that is, we have $m$ flight legs and we sell $n$ different products. This includes the cabin capacities, the possible aircraft configurations and the allowed number of upgrades on the flight legs. Lastly, the remaining booking horizon length $T$ and the mean aggregate demand to come at time $T$ for each ODF itinerary, given by $\boldsymbol{\mu}(T)$, are given as input. As airlines make adjustments to booking limits regularly, the implemented simulation is a multi-stage process. We assume that the airline updates its booking limits daily, consequently, we implement daily scheduled revision points in our simulation. Moreover, a booking period is defined to be the period between two revision points. As we assume a linear booking curve, we adjust the demand forecast in every booking period based on the remaining time. That is, we approximate the mean aggregate demand to come at time $t$ for each ODF itinerary, given by $\boldsymbol{\mu}(t)$, by the relation given by Equation 47.

$$\boldsymbol{\mu}(t) = \frac{t}{T} \cdot \boldsymbol{\mu}(T) \tag{47}$$

Moreover, as we assume a linear booking curve, the probability of a request for product $j$ at time $t$, given by $p_j^t$, can be calculated by Equation 48.

$$p_j^t = \frac{\mu_j^t}{t} = \frac{\mu_j^T}{T} \tag{48}$$

The arrival of demand for a product is modeled as a Poisson process, which is described by a single parameter, namely the average arrival rate $\lambda$. That is, let $X_1, X_2, ...$ be a sequence of independent exponentially distributed random variables with rate $\lambda$ (and mean $\frac{1}{\lambda}$). Moreover, let $S_0 = 0$ and $S_n = \sum_{k=1}^{n} X_k$, for $n = 1, 2, ...$ and let $N(t) = \max\{n : S_n \leq t\}$, for all $t \geq 0$. Then $\{N(t), t \geq 0\}$

is said to be a Poisson process with rate $\lambda$. Here, the realization of $X_n$ is the time between the $(n-1)$th and $n$th event (arrival), the realization of $S_n$ is the arrival time of the $n$th event and the realization of $N(t)$ is the number of arrivals in $(0, t]$. The mean and variance of $N(t)$ are given by $E[N(t)] = V[N(t)] = \lambda t$.

Given the assumption of a linear booking curve, the arrival rate vector $\boldsymbol{\lambda}$ is given by $\boldsymbol{\mu}(T)/T$. For each product, we simulate the Poisson process for $T$ time units. Here, we use the fact that inter-arrival times $X_1, X_2, ...$ for a Poisson process are exponentially distributed with rate $\lambda$. When the arrivals of all products have been simulated, the arrivals are combined in one list and sorted on arrival time. Using this list, we can start the simulation. A schematic overview of the simulation is given in Figure 1. We start at remaining time $t = T$ and iterate over the booking periods. For every booking period, we update the demand forecasts, for which we use the assumption of a linear booking curve. Then, based on the updated demand forecasts and most recent remaining capacities $\mathbf{x}$, we update the booking limits. That is, we perform the decomposition model as described in Section 4.2. Arrivals of requests within the booking period are then accepted or rejected based on the updated booking limits. If the request is accepted, we update the remaining capacities $\mathbf{x}$.

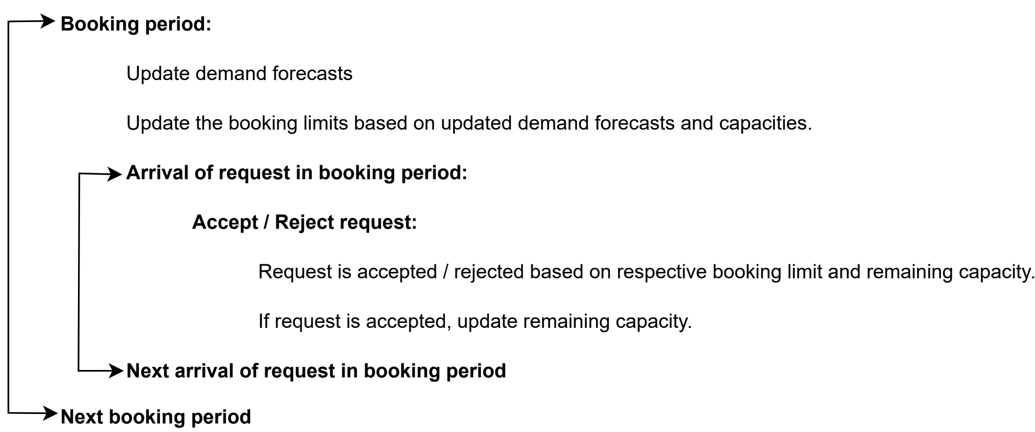


Figure 1: Schematic overview of the simulation process.

This process is repeated a large number of times and after doing these iterations, one can derive several summary statistics. Note that the number of times the process is repeated is called the sample size of the Monte Carlo simulation for the remainder of this thesis. Obviously, the most important summary statistic is given by the average realized revenue of the network. Other interesting summary statistics are the average final load factor of the network, the number of times a configuration is chosen and the average number of upgrades that are needed. Here, we define the load factor as the total number of seats sold for a flight leg over the total capacity of the flight leg. Lastly, the average computation time is also a key statistic.

## 5.2 Benchmark solution methods and approximate upper bound

We compare the performance of our approaches with 2 benchmark methods. The first method uses the optimal dual variables of Constraints (17) of the EDLP model as network bid prices and these network bid prices are used directly as a control mechanism. For this, we first must solve the DMILP model and fix the aircraft configuration $\mathbf{z}$. If a request for an ODF itinerary arrives using multiple flight legs, the summation of the bid prices of the flight legs used by the itinerary is used as bid price. We call this type of control the mathematical programming control (MPC). Also, please note the resemblance of this approach with the famous deterministic linear programming method used in literature. That is, the deterministic linear programming method in literature uses the optimal dual variables of Constraints (13) of the DLP model as bid prices, however, this method does not consider up- and downgrading and aircraft configuration selection. The second benchmark method uses the calculated opportunity costs of the $n$ products, which is given by Equation 28, directly as control mechanism. This method comes closest to the approach of Bertsimas and Popescu (2003), however, Bertsimas and Popescu (2003) calculate the opportunity cost on a real-time basis. We call this type of control the opportunity cost control (OCC). Lastly, please note that both methods can be argued to be unsafe, due to the open/closed control philosophy.

The realized revenue of the (benchmark) approaches can also be compared using the perfect-hindsight solution. That is, an upper bound on the maximum realized revenue of a single run of the Monte Carlo simulation can be obtained by solving the perfect hindsight mixed integer linear programming (PHMILP) model given by Equation (49)-(54). This model uses perfect information on the realized demand, given by $\bar{\boldsymbol{\mu}}$. Therefore, given the realized demand, this model essentially finds an optimal allocation of the available capacity. This upper bound might, however, be loose as we use perfect information on the realization of demand which is clearly too optimistic for real-time control.

Perfect hindsight mixed integer linear programming (phmilp) model

$$V^{PHMILP}(\mathbf{x}, \bar{\boldsymbol{\mu}}) = \max \mathbf{r}^T \mathbf{y} \tag{49}$$

$$\text{s.t. } \mathbf{A}^T \mathbf{y} - \mathbf{C}^T \mathbf{g} \leq \mathbf{x} + \mathbf{B}^T \mathbf{z} \tag{50}$$

$$\mathbf{0} \leq \mathbf{y} \leq \bar{\boldsymbol{\mu}} \tag{51}$$

$$-\mathbf{d} \leq \mathbf{g} \leq \mathbf{f} \tag{52}$$

$$\mathbf{z} < \mathbf{c} \tag{53}$$

$$\mathbf{y} \in \mathbb{Z}^n_{0+}, \mathbf{g} \in \mathbb{Z}^m, \mathbf{z} \in \mathbb{Z}^m_{0+} \tag{54}$$

An approximate upper bound on the realized revenue of the stochastic network seat inventory control problem can then be obtained by taking the average of the found upper bounds of the runs of the Monte Carlo simulation.

## 5.3 Numerical results

In this section, we present numerical results that show the relative performance of the proposed approaches, the benchmark solution methods and the approximate upper bound on the realized revenue. Our goal is to evaluate the different approaches using the computation time, the realized revenue and robustness as criteria. The booking process simulation as described in Section 5.1 allows us to use the same sequence of product requests for each (benchmark) approach. In this way, the difference in simulation results can be solely attributed to using different approaches.

It is chosen to base the numerical experiments on real historical airline data. In this way, we try to obtain a realistic mix of local and connecting flow and a realistic mix of fare classes. Here, with local flow, we mean requests for single-leg ODF itineraries and with connecting flow, we mean requests for multi-leg ODF itineraries. First, in Section 5.3.1, we perform a case study on the entire large-scale network of a major airline in Europe. Applying a Monte Carlo simulation with a significant sample size on the entire network becomes problematic due to the size of the problem and computation limitations. The entire network can, however, be used to show whether the proposed approaches still work in a reasonable time for a real, large network of a major airline. Therefore, we also consider this network 3 months for departure, as then the remaining capacities and incoming demand forecasts are still high. Then, in Section 5.3.2, we generate subnetworks from the entire large-scale network of the airline. However, now, we consider the network 1 month before departure. These subnetworks do allow us to apply a Monte Carlo simulation with a significant sample size on all proposed approaches and benchmark solution methods in a manageable time. This allows us to derive several summary statistics about the stochastic problem. That is, the subnetworks are used to compare the realized revenues of the different approaches and to apply a sensitivity analysis on the influence of the sample size of the Monte Carlo simulation, the control mechanism and the network characteristics. In addition, it allows us to perform a robustness analysis on the influence of the demand estimates.

The framework is implemented in Java, version 18.0.2. Moreover, the mathematical programming models are solved using a CPLEX solver with version 22.1.0. The case study on the entire large-scale network is performed using an Intel Xeon Gold 6134 CPU 3.20 GHz device running Windows 10 with 192GB RAM. The machine is, however, also used by other highly demanding operations, therefore approximately half of the machine utilities are available. The case study on the subnetworks is performed using an Intel i7 3.0 GHz processor with 32 GB RAM.

The independent single-leg problems are solved in parallel to speed up computation. The number of threads is set equal to the machine's available logical processors. It was chosen to not compute the opportunity costs in parallel, although this would reduce computation time significantly. However, the opportunity costs are calculated using mathematical programming models, which use an external CPLEX library. Making multiple calls to this library simultaneously became problematic, though it should be possible. Nonetheless, due to time limitations, it was decided that the parallelization of the mathematical programming models is outside the scope of this research. Also, the CPLEX library itself can use multiple threads, so parallelization of the mathematical programming models becomes less urgent.

### 5.3.1 Case study entire large-scale network

The network consists of 661 flight legs and 267492 different products. To the best of the author's knowledge, this network is one of the largest networks considered in airline revenue management literature (Birbil et al., 2014). We consider the network three months before departure, therefore the booking horizon $T$ is set to 90 days. Of the 661 flight legs, in total 564 flight legs have non-fixed aircraft configurations. Moreover, let the demand factor of a flight leg be given by the sum of the total demand forecast and seats sold for the flight leg over the initial capacity of the flight leg. The demand factor varies between 0.06 to 5.62 on the flight legs, with an average of 1.01 and a standard deviation of 0.54. The 267492 different products correspond to 7505 different cabin-specific OD itineraries. A product uses between 1 and 3 flight legs, with an average of 1.497 flight legs and a standard deviation of 0.516 flight legs. This indicates that there is a good mix between local and connecting flow. Additional summary statistics are given in Appendix A.

Table 2 shows the results of the Monte Carlo simulations with a sample size of two for the different approaches. Both the OCD-DP approach and the OCC benchmark approach took too long and were, therefore, terminated early. However, using extrapolation, the average computation time for optimizing a single booking period could be computed for the two approaches. That is, the OCD-DP approach and the OCC benchmark approach were terminated halfway through the first run and using extrapolation, the average computation time per booking period could be determined.

As expected, the OCD-DP approach has the highest computation time, taking on average a small five hours per booking period and, thereby, is considered to be too slow. The OCC benchmark approach was also considered to be too slow, taking on average approximately over two hours per booking period. Both the ROCD-DP, DD-DP and MPC approaches had reasonable computation times, where the ROCD-DP approach took on average approximately one hour per booking period, being twice as slow as the DD-DP approach. The MPC approach had the smallest computation time, namely, on average a booking period was optimized in 2.9 seconds.

The large computation time of the OCD-DP approach and OCC benchmark approach is caused by the high number of cabin-specific OD itineraries. That is, the computation time of the opportunity costs of the 7505 different cabin-specific OD itineraries took on average approximately over two hours, making the approach impractical. 'Warm starting' the integer linear programming solver for the OCD-DP and OCC approaches did not give the large expected decrease in computation time. The decrease in computation time by using a 'warm start' was approximately 54.0%, which is substantial, however, not enough to make the computation time manageable. It must, however, also be noted that the opportunity costs can be calculated in parallel. Therefore, in theory, if there are no limitations on the used hardware, the opportunity costs can be solved in the same time as one opportunity cost, which is only a few seconds.

As only a sample size of two was used, one cannot make any statement regarding the significance of the differences in the average realized revenue, load factor and upgrade. The realized demand was 0.09% lower and 0.81% higher than the predicted demand for the first and second run, respectively. Quite surprisingly, the MPC approach succeeded to obtain a higher revenue than the ROCD-DP and DD-DP approaches for both runs. As the realized demand was both higher and smaller than the predicted demand, one cannot make any statements about whether this is a result of one method being more restrictive in accepting demand than another. In most cases, the final configuration corresponds to the zero configuration. That is, for the 564 flight legs with no-fixed aircraft configuration, on average in 84.6% of the cases the zero-configuration is chosen for the ROCD-DP, DD-DP and MPC benchmark approach.

Table 2: Results of the Monte Carlo simulations with a sample size of two on the entire large-scale network for the different approaches. The computation times of the optimization of a single booking period for the OCD-DP approach and OCC benchmark approach are determined using extrapolation.

| Approaches | OCD-DP | ROCD-DP | DD-DP | OCC | MPC |
|---|---|---|---|---|---|
| Ave. realized revenue (€) | - | 20,726,092 | 20,719,840 | - | 20,931,173 |
| Gap to approx. upper bound (%) | - | 2.15 | 2.18 | - | 1.18 |
| Ave. load factor | - | 0.797 | 0.797 | - | 0.807 |
| Ave. upgrades | - | 0.800 | 0.801 | - | 0.898 |
| Ave. comp. time booking period (s) | 16770.5 | 4167.36 | 2446.5 | 8187.8 | 2.9 |
| Ave. flight legs with final config. 0 | - | 478.0 | 478.0 | - | 475.5 |
| Ave. flight legs with final config. 1 | - | 23.5 | 23.0 | - | 27.5 |
| Ave. flight legs with final config. 2 | - | 16.0 | 16.0 | - | 14.5 |
| Ave. flight legs with final config. 3 | - | 33.0 | 33.5 | - | 33.5 |
| Ave. flight legs with final config. 4 | - | 12.5 | 12.5 | - | 12.0 |
| Ave. flight legs with final config. 5 | - | 1.0 | 1.0 | - | 1.0 |

### 5.3.2 Case study subnetworks

As stated previously, subnetworks are generated from the entire large-scale network. We do this by only considering those flight legs arriving within a specific arrival time interval or departing within a specific departure time interval from the hub of the considered airline. As connecting passengers need approximately one hour to transfer to different flight legs, the start of the departure time interval begins an hour after the start of the arrival time interval. The lengths of the arrival and departure time intervals are set such that we obtain a small enough network to be able to apply Monte Carlo simulations with sufficient sample size, but also still have a reasonable mix between local and connecting flow. Multiple networks can be generated by shifting the time intervals.

By experimenting, eight subnetworks were found, each having its own unique characteristics. Summary statistics of the eight subnetworks are given in Table 3. We consider five larger subnetworks (subnetworks 1 through 5) and three smaller subnetworks (subnetworks 6 through 8). Most interesting is the difference in demand factors, the number of different cabin-specific OD itineraries and the average number of flight legs used by a product. For example, subnetwork 2 differs from subnetwork 1 by having less connecting flow, resulting in fewer cabin-specific OD itineraries and a lower average number of flight legs used by products. Subnetworks 3, 4 and 5 are more mediocre networks compared to subnetworks 1 and 2 and differ mainly in demand factor. Subnetwork 6 does not contain any connecting flow and, therefore, can be seen as a network of independent single-flight legs. Subnetworks 7 and 8 do contain little connecting flow and mainly differ in demand factor.

Moreover, it must be noted that even the subnetwork with the most connecting flow does not come close to the amount of connecting flow seen in the entire network of the major airline. This is caused by the fact that the arrival and departure time intervals are limited to get sufficiently small networks, and, therefore, we miss a substantial part of the connecting flow. Moreover, we only consider connecting flow at the hub, and not at other airports. Lastly, note that the average demand factor of the subnetworks is considerably higher than the average demand factor of the entire network considered 3 months before departure, as considered in Section 5.3.1. This can be explained by the fact that the average demand factor of the entire network considered 1 month before departure has a significantly higher demand factor compared to 3 months before departure, namely 1.27 instead of 1.01.

Table 3: Summary statistics of the eight generated subnetworks.

| Network | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Total flight legs | 27 | 20 | 24 | 23 | 18 | 9 | 6 | 6 |
| With non-fixed aircraft config. | 24 | 15 | 20 | 21 | 14 | 9 | 4 | 5 |
| Open for sale economy cabin (%) | 33.3 | 28.2 | 45.7 | 32.9 | 42.4 | 51.4 | 19.8 | 50.6 |
| Open for sale business cabin (%) | 26.2 | 18.4 | 55.0 | 48.4 | 58.8 | 69.4 | 10.1 | 38.2 |
| Total Products | 6908 | 3807 | 4885 | 4501 | 3758 | 1307 | 1143 | 979 |
| Average demand factor | 1.32 | 1.36 | 1.11 | 1.40 | 1.13 | 1.33 | 1.68 | 0.97 |
| Number of different cabin-specific OD itineraries | 118 | 58 | 80 | 70 | 63 | 15 | 16 | 12 |
| Average number of flight legs used by product | 1.24 | 1.08 | 1.17 | 1.15 | 1.17 | 1.00 | 1.07 | 1.04 |

As the OCD-DP approach is proven to be impractical with the current implementation and available hardware during the case study on the entire large-scale network in Section 5.3.1, we will exclude this approach from the experiments on the subnetworks. The OCC benchmark approach is, however, included, as this method still provides a good benchmark and can be computed relatively fast due to the low number of different cabin-specific OD itineraries. The results of the Monte Carlo simulations with a sample size of 100 on the 5 larger subnetworks and the 3 smaller subnetworks for the different approaches are given in Tables 4 and 5, respectively.

On average, the ROCD-DP approach showed a revenue increase of, respectively, 0.83% and 0.74% compared to the MPC and OCC benchmark approaches, which corresponds on average to 0.15 standard deviations for both the MPC and OCC benchmark approaches. The DD-DP approach performed quite similarly, resulting in an average revenue increase of, respectively, 0.81% and 0.73% compared to the MPC and OCC benchmark approaches. Again, this increase corresponded on average to 0.15 standard deviations for both the MPC and OCC benchmark approaches. Only for subnetwork 1, the ROCD-DP and DD-DP approaches did not result in a positive revenue increase compared to the benchmark approaches. Although the increase in realized revenue seems relatively small, it can be considered fairly significant, considering that the average gap to the approximate upper bound equals, respectively, 3.11% and 3.02% for the MPC and OCC benchmark approaches. Therefore, it is reasonable to assume that the benchmark approaches are already quite close to the optimum for the network seat inventory control problem.

The ROCD-DP and DD-DP approaches have almost identical final load factors and are more restrictive in accepting requests compared to the MPC and OCC benchmark approaches. That is, for both the ROCD-DP and DD-DP approaches the average final load factors are, respectively, 0.42% and 0.09% smaller compared to the MPC and OCC benchmark approaches. This decrease on average corresponds to, respectively, 0.72 and 0.21 standard deviations for the MPC and OCC benchmark approaches. In addition, the ROCD-DP and DD-DP approaches on average upgrade less compared to the MPC benchmark approach. That is, the ROCD-DP and DD-DP approaches upgrade, respectively, 3.9% and 4.0% less compared to the MPC approach, which corresponds on average to, respectively, 0.26 and 0.27 standard deviations for the MPC approach. However, the OCC approach on average upgrades the least, respectively, 6.5% and 6.4% less compared to the ROCD-DP and DD-DP approaches. This decrease corresponds on average to, respectively, 0.30 standard deviations for both the ROCD-DP and DD-DP approaches.

The computation times of the ROCD-DP and DD-DP approaches do not differ remarkably, which can be explained by the small computation time of the relaxed DMILP model and the relatively few different cabin-specific OD itineraries. Moreover, a large part of the computation time is used for solving the dynamic programming models, which is the same for both approaches. The OCC benchmark approach does take a substantially longer time compared to the MPC benchmark approach. That is, on average, the OCC and MPC benchmark approaches take, respectively 2.04 and 0.07 seconds, which corresponds to a difference of 22.2 standard deviations for the MPC approach. This can be explained by the fact that the mixed integer linear programming model results in significantly higher computation times compared to the linear programming model.

Quite surprisingly, the ROCD-DP and DD-DP approaches ended up with the same final configuration on each of the flight legs for all 8 subnetworks. Nonetheless, not the same ODF itinerary requests were accepted, resulting in a small difference in realized revenue. This might be explained by the fact that the final step of the decomposition model is the same for the ROCD-DP and DD-DP approaches. That is, the independent single-flight legs are solved by the same dynamic programming model. The prorated revenues used within the independent single-leg problems do, however, differ by the difference in computation, however, this only causes small differences in the acceptance of requests and no differences in the final aircraft configurations.

| Approaches | ROCD-DP | DD-DP | OCC | MPC |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 529,846 (20,547) | 529,626 (20,566) | 531,282 (20,218) | 530,211 (19,594) |
| % Increase over MPC (st. dev.) | -0.07 (0.019) | -0.11 (0.030) | 0.20 (0.055) | - |
| Gap to approx. ub. (%) | 3.30 | 3.34 | 3.04 | 3.23 |
| Ave. (st. dev.) load factor | 0.9656 (0.0041) | 0.9655 (0.0041) | 0.9682 (0.0037) | 0.9716 (0.0036) |
| Ave. (st. dev.) upgrades | 0.758 (0.178) | 0.754 (0.178) | 0.693 (0.175) | 0.827 (0.164) |
| Ave. (st. dev.) comp. time (s) | 7.89 (6.67) | 7.45 (6.55) | 4.65 (2.25) | 0.17 (0.17) |

(a) Subnetwork 1

| | | | | |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 506,712 (15,759) | 506,614 (15,741) | 500,852 (16,261) | 500,927 (15,426) |
| % Increase over MPC (st. dev.) | 1.15 (0.375) | 1.14 (0.369) | -0.02 (0.005) | - |
| Gap to approx. ub. (%) | 2.67 | 2.69 | 3.79 | 3.78 |
| Ave. (st. dev.) load factor | 0.9497 (0.0038) | 0.9497 (0.0038) | 0.9500 (0.0038) | 0.9537 (0.0038) |
| Ave. (st. dev) upgrades | 0.507 (0.149) | 0.508 (0.149) | 0.487 (0.140) | 0.534 (0.150) |
| Ave. (st. dev.) comp. time (s) | 4.57 (2.23) | 4.44 (2.28) | 5.86 (3.52) | 0.14 (0.13) |

(b) Subnetwork 2

| | | | | |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 351,681 (19,331) | 351,634 (19,289) | 349,659 (19,328) | 350,035 (19196) |
| % Increase over MPC (st. dev.) | 0.47 (0.086) | 0.46 (0.083) | -0.11 (0.020) | - |
| Gap to approx. ub. (%) | 1.77 | 1.78 | 2.33 | 2.23 |
| Ave. (st. dev.) load factor | 0.8774 (0.0066) | 0.8774 (0.0066) | 0.8784 (0.0065) | 0.8805 (0.0065) |
| Ave. (st. dev) upgrades | 0.517 (0.116) | 0.517 (0.117) | 0.477 (0.111) | 0.562 (0.117) |
| Ave. (st. dev.) comp. time (s) | 8.74 (5.64) | 8.40 (5.60) | 2.26 (2.16) | 0.06 (0.06) |

(c) Subnetwork 3

| | | | | |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 282,072 (12,277) | 282,059 (12,278) | 279,669 (12,233) | 280553 (12,099) |
| % Increase over MPC (st. dev.) | 0.54 (0.126) | 0.54 (0.124) | -0.32 (0.073) | - |
| Gap to approx. ub. (%) | 1.88 | 1.89 | 2.72 | 2.41 |
| Ave. (st. dev.) load factor | 0.9098 (0.0056) | 0.9098 (0.0056) | 0.9103 (0.0056) | 0.9129 (0.0054) |
| Ave. (st. dev) upgrades | 0.996 (0.188) | 0.996 (0.188) | 0.870 (0.199) | 0.987 (0.186) |
| Ave. (st. dev.) comp. time (s) | 8.36 (6.33) | 8.18 (5.64) | 1.06 (0.51) | 0.03 (0.04) |

(d) Subnetwork 4

| | | | | |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 291,421 (18,776) | 291,410 (18,782) | 289,793 (18,651) | 289,697 (18,004) |
| % Increase over MPC (st. dev.) | 0.60 (0.096) | 0.59 (0.095) | 0.03 (0.005) | - |
| Gap to approx. ub. (%) | 1.70 | 1.70 | 2.25 | 2.28 |
| Ave. (st. dev.) load factor | 0.8673 (0.0078) | 0.8673 (0.0078) | 0.8675 (0.0078) | 0.8694 (0.0078) |
| Ave. (st. dev) upgrades | 0.629 (0.146) | 0.628 (0.145) | 0.584 (0.151) | 0.669 (0.143) |
| Ave. (st. dev.) comp. time (s) | 6.36 (3.95) | 6.16 (3.93) | 1.50 (1.41) | 0.06 (0.05) |

(e) Subnetwork 5

Table 4: Results of the Monte Carlo simulations with a sample size of 100 on the 5 larger subnetworks for the different approaches. This includes the average (and standard deviation in parenthesis) realized revenue, load factor, upgrades and computation time of a single booking period. Moreover, the realized revenue is expressed as a percentage difference of the realized revenue of the MPC benchmark approach (and number of standard deviations for the MPC benchmark approach corresponding to this difference in parenthesis). Lastly, the gap to the approximate upper bound is given.

| Approaches | ROCD-DP | DD-DP | OCC | MPC |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 77,853 (3264) | 77,853 (3264) | 77,421 (3185) | 77,458 (3097) |
| % Increase over MPC (st. dev.) | 0.51 (0.127) | 0.51 (0.127) | -0.05 (0.012) | - |
| Gap to approx. ub. (%) | 1.74 | 1.74 | 2.28 | 2.24 |
| Ave. (st. dev.) load factor | 0.9664 (0.0076) | 0.9664 (0.0076) | 0.9668 (0.0072) | 0.9708 (0.0065) |
| Ave. (st. dev) upgrades | 0.909 (0.282) | 0.909 (0.282) | 0.767 (0.262) | 0.941 (0.253) |
| Ave. (st. dev.) comp. time (s) | 3.08 (2.65) | 3.09 (2.68) | 0.64 (0.83) | 0.05 (0.09) |

(a) Subnetwork 6

| | | | | |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 90,555 (7252) | 90,548 (7251) | 89,098 (7713) | 88,923 (6696) |
| % Increase over MPC (st. dev.) | 1.84 (0.244) | 1.83 (0.243) | 0.20 (0.026) | - |
| Gap to approx. ub. (%) | 3.96 | 3.97 | 5.51 | 5.69 |
| Ave. (st. dev.) load factor | 0.9804 (0.0050) | 0.9804 (0.0050) | 0.9788 (0.0049) | 0.9837 (0.0042) |
| Ave. (st. dev) upgrades | 0.278 (0.198) | 0.278 (0.198) | 0.292 (0.212) | 0.235 (0.142) |
| Ave. (st. dev.) comp. time (s) | 0.93 (0.77) | 0.91 (0.75) | 0.27 (0.22) | 0.03 (0.03) |

(b) Subnetwork 7

| | | | | |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 86,195 (9659) | 86,192 (9663) | 85,491 (9130) | 84,864 (8725) |
| % Increase over MPC (st. dev.) | 1.57 (0.153) | 1.56 (0.152) | 0.74 (0.072) | - |
| Gap to approx. ub. (%) | 1.46 | 1.47 | 2.27 | 2.99 |
| Ave. (st. dev.) load factor | 0.8695 (0.0156) | 0.8695 (0.0156) | 0.8729 (0.0156) | 0.8745 (0.0157) |
| Ave. (st. dev) upgrades | 0.205 (0.140) | 0.203 (0.141) | 0.202 (0.145) | 0.255 (0.129) |
| Ave. (st. dev.) comp. time (s) | 2.06 (1.75) | 2.21 (1.86) | 0.07 (0.11) | 0.01 (0.04) |

(c) Subnetwork 8

Table 5: Results of the Monte Carlo simulations with a sample size of 100 on the 3 smaller subnetworks for the different approaches. This includes the average (and standard deviation in parenthesis) realized revenue, load factor, upgrades and computation time of a single booking period. Moreover, the realized revenue is expressed as a percentage difference of the realized revenue of the MPC benchmark approach (and number of standard deviations for the MPC benchmark approach corresponding to this difference in parenthesis). Lastly, the gap to the approximate upper bound is given.

Interesting to see is that subnetworks 3, 4 and 5, which were more mediocre networks compared to subnetworks 1 and 2 and mainly differ in demand factors, show largely identical performance. That is, on average, subnetworks 3, 4 and 5 show a realized revenue increase of approximately 0.54% and 0.53% for the ROCD-DP and DD-DP approaches, respectively, compared to the MPC approach.

For subnetwork 6, the ROCD-DP and DD-DP approaches performed identically. This can be explained by the fact that this network does not contain any connecting flow. Therefore, the network is basically a network of independent single-flight legs. Consequently, the proration step does not have any influence, and, therefore, both approaches are similar. Nonetheless, the ROCD-DP and DD-DP approaches still showed a revenue increase of, respectively, 0.51% and 0.56% compared to the MPC and OCC benchmark approaches. This increase can solely be attributed to the dynamic programming model used to solve the single-leg problems.

In the following sections, we apply a sensitivity analysis on the influence of the sample size of the Monte Carlo simulation, the type of control mechanism and the network characteristics. For the latter, we consider the influence of increasing the ratio of connecting to local flow, that is, increasing the number of products using multiple flight legs compared to products using single flight legs. Moreover, we look at the influence of the average demand factor on the performance of the different approaches. We try to incorporate the results obtained for the 8 subnetworks as starting point. There are also other factors that can influence the performance of the approaches, but, are for the scope of this research, not examined. For example, one can examine the influence of the demand process of the booking process simulation or examine the influence of the ratio of the number of flight legs with non-fixed aircraft configuration over the entire number of flight legs. Lastly, we perform a robustness analysis in the demand estimates for the different approaches.

#### 5.3.2.1 Sensitivity analysis on influence of sample size Monte Carlo simulation

As stated previously, only for subnetwork 1 the ROCD-DP and DD-DP approaches did not result in positive revenue increases compared to the benchmark approaches. The differences were, however, minimal, namely an average decrease of 0.07% and 0.11% for the ROCD-DP and DD-DP approaches, respectively, compared to the MPC benchmark approach. This corresponds on average only to, respectively, 0.019 and 0.030 standard deviations for the MPC approach. It can be interesting to know whether the obtained results differ drastically if the sample size of the Monte Carlo simulation is increased, and, therefore, whether the originally chosen sample size is justifiable. Therefore, for subnetwork 1, we increase the sample size from 100 to 400. The results are given in Table 6. Interesting to see is that the results are quite similar to the results of the Monte Carlo simulations with a sample size of 100. That is, the relative performance between the approaches is fairly identical. This gives a good indication that the chosen sample size of 100 for the Monte Carlo simulation is sufficient.

#### 5.3.2.2 Sensitivity analysis on influence of control mechanism

Considering the results for subnetwork 6, it is interesting to know whether the increase in realized revenue for the subnetworks is caused by the difference in solution approach or by the difference in control mechanism. That is, both benchmark approaches have an open/closed control philosophy, in the sense that we only have a single bid price as control mechanism per flight leg-cabin combination. This is in contrast with the proposed approaches which have a bid price matrix as control mechanism per flight leg. That is, the obtained bid prices of the proposed approaches are a function of both the remaining time $t$ and the remaining capacities in the cabins. Therefore, we introduce a third benchmark approach, which is similar to the MPC approach, however, this approach has

Table 6: Results of the Monte Carlo simulations with a sample size of 400 on subnetwork 1 for the different approaches.

| Approaches | ROCD-DP | DD-DP | OCC | MPC |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 529,924 (19,704) | 529,682 (19,707) | 531,048 (19,610) | 530,017 (19,262) |
| % Increase over MPC (st. dev.) | -0.02 (0.005) | -0.06 (0.017) | 0.19 (0.054) | - |
| Gap to approx. ub. (%) | 3.17 | 3.21 | 2.96 | 3.15 |
| Ave. (st. dev.) load factor | 0.9650 (0.0038) | 0.9650 (0.0037) | 0.9678 (0.0035) | 0.9711 (0.0034) |
| Ave. (st. dev.) upgrades | 0.744 (0.173) | 0.741 (0.172) | 0.684 (0.172) | 0.812 (0.178) |
| Ave. (st. dev.) comp. time (s) | 8.05 (6.57) | 7.45 (6.55) | 8.15 (3.75) | 0.07 (0.07) |

an additional step in which a heuristic is used to generate a bid price vector per flight leg-cabin combination. That is, we generate bid prices being a function of the remaining capacity in the flight leg-cabin combination.

The general procedure of the heuristic is the following. First, the length of the bid price vector is determined by the remaining capacities in the cabins. For this, we will use the aircraft configuration as determined by the DMILP model. Then, the revenues of the products are prorated using the proration scheme given by Equation (38). Now, for each flight leg-cabin combination, we order the associated products from highest to lowest prorated revenue. The products are aggregated in blocks with a predicted demand of 1. Here, the aggregation starts from the highest prorated revenue to the lowest prorated revenue. Then, the bid price of the last seat in the cabin equals the lowest prorated revenue of the products within the first aggregated block with a predicted demand of 1. That is, this first aggregated block contains the highest prorated revenues for the considered flight leg and we take the lowest prorated revenue within the block as bid price. The bid price of the second last seat in the cabin equals the lowest prorated revenue of the products within the second aggregated block with a predicted demand of 1. This procedure is continued until the bid price vector is filled. Note that by using this heuristic, the bid price of a capacity on a flight leg-cabin combination will always be equal to or higher than the bid price of the same capacity for the original MPC benchmark approach.

Surprisingly, this approach led to an average decrease in realized revenue of 7.4% compared to the MPC benchmark approach for the eight subnetworks, which corresponds on average to 1.53 standard deviations for the MPC benchmark approach. The adjusted MPC approach was more restrictive in accepting requests than expected. That is, the average final load factor decreased from 0.927 to 0.879, which corresponds on average to 7.2 standard deviations for the MPC benchmark approach. Evidently, the bid prices were set too high, leading to too much lost revenue.

Therefore, an alternative version is introduced. This version is fairly similar, however, now, for each flight leg-cabin combination, products are ordered from lowest to highest prorated revenue. All products with an associated prorated revenue lower than the bid price of the flight leg-cabin combination are now excluded. The remaining products are also aggregated in blocks with a predicted demand of 1, however, now, the aggregation starts from the lowest prorated revenue to the highest prorated revenue. Now, the bid price of the first open seat in the cabin equals the lowest prorated revenue of the products within the first aggregated block with a predicted demand of 1. That is, the first aggregated block contains the lowest prorated revenues of the products that were not excluded. The bid price of the second open seat in the cabin equals the lowest prorated revenue of the products within the second aggregated block with a predicted demand of 1. This procedure is continued until the bid price vector is filled. Also for this version of the heuristic, the bid price of a capacity on a flight leg-cabin combination will always be equal to or higher than the bid price of the same capacity for the original MPC benchmark approach.

This approach performed significantly better, leading to an average increase in realized revenue of 0.19% compared to the MPC benchmark approach for the eight subnetworks, which corresponds on average to 0.03 standard deviations for the MPC approach. The average final load factor decreased less excessively, from 0.927 to 0.925, which corresponds on average to 0.28 standard deviations for the MPC benchmark approach. The ROCD-DP and DD-DP approaches still showed a significant performance increase over the adjusted MPC benchmark approach, namely an average increase of, respectively, 0.64% and 0.63%, corresponding on average to 0.13 standard deviations for the adjusted MPC benchmark approach. This suggests that the increase in realized revenue is mainly caused by the difference in solution approach and not by the difference in control mechanism.

#### 5.3.2.3 Sensitivity analysis on influence of connecting flow

The increase in realized revenue of the ROCD-DP and DD-DP approaches was significantly higher for subnetwork 2 than for the other four larger subnetworks. Subnetwork 2 was characterized by its low connecting flow, and, therefore, is it interesting to know whether the relative performance of the ROCD-DP and DD-DP approaches decreases compared to the benchmark approaches when there is more connecting flow. Especially regarding the fact that the ROCD-DP and DD-DP approaches even performed slightly worse than the benchmark approaches for subnetwork 1, which was characterized by having the highest connecting flow.

We, therefore, introduce ten artificial multi-leg OD itineraries for subnetwork 2. These multi-leg OD itineraries consist of two adjacent flight legs. For both adjacent flight legs, we pick randomly an existing associated single-leg product and, then, combine the two products into one multi-leg product for the artificial multi-leg OD itinerary. The new multi-leg product has an associated revenue equal to 0.89 times the sum of the associated revenues of the randomly chosen single-leg products. Here, a factor of 0.89 is chosen as the multi-leg product is inferior to the corresponding individual single-leg products and, hence, customers are less willing to pay for this product. Moreover, the demand for the new multi-leg product is equal to the average demand for the two randomly chosen single-leg products. We generate 100 artificial products per multi-leg OD itinerary in this manner. In this way, it is very likely that, first, we observe demand for the artificial multi-leg OD itineraries and, second, that the associated requests for the artificial multi-leg OD itineraries are competitive.

By introducing these products for the artificial multi-leg OD itineraries, the demand factor of the network increased from 1.36 to 1.78. To make the comparison fair, all demand forecasts are adjusted by the same factor to decrease the demand factor to the original 1.36. A product now uses on average 1.27 flight legs instead of the original 1.08 flight legs, indicating that we have more connecting flow.

The results of the Monte Carlo simulations on subnetwork 2 with extra generated multi-leg demand are given in Table 7. Instead of a revenue increase, the ROCD-DP and DD-DP approaches show a decrease in realized revenue of, respectively, 0.57% and 1.00% compared to the MPC benchmark approach. Compared to the OCC benchmark approach, the decrease in realized revenue is even higher, namely 1.06% and 1.48% for the ROCD-DP and DD-DP approaches, respectively. The average load factor has decreased significantly compared to the original results for subnetwork 2, namely on average by 3.2% for the different approaches, which corresponds on average to 8.06 standard deviations.

Also, the gap to the approximate upper bound has increased significantly, on average from 3.2% to 5.5%. Looking at the original results of the large subnetworks, one can also observe that the average gap to the approximate upper bound is also highest for subnetwork 1, which has the largest connecting flow. The larger gap might be explained by the fact that non-realized predicted demand for multi-leg products has more impact than non-realized predicted demand for single-leg products. That is, for the multi-leg products we accounted for the demand on multiple flight legs and for the single-leg products we accounted for the demand only on the single-flight leg. Therefore, the gap between the perfect-hindsight solution might increase when we have more demand for multi-leg products. It seems that the ROCD-DP and DD-DP approaches are more sensitive to this behavior, and, therefore, hand in potential revenue. On the other hand, it can also be the case that the MPC and OCC benchmark approaches benefit from the extra generated products, and, therefore, extra demand points. That is, for the same demand factor, the number of products has increased from 3807 to 4807, corresponding to an increase of 26.3%. Future research should find out what the causes are of the deterioration in the relative performance of the ROCD-DP and DD-DP approaches compared to the benchmark approaches and whether this is structural for networks with more connecting flow or not.

Table 7: Results of the Monte Carlo simulations with a sample size of 100 on subnetwork 2 with added demand for artificial multi-leg OD itineraries for the different approaches.

| Approaches | ROCD-DP | DD-DP | OCC | MPC |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 431,550 (18,060) | 429,688 (18,298) | 436,153 (18,747) | 434,032 (16,061) |
| % Increase over MPC (st. dev.) | -0.57 (0.155) | -1.00 (0.270) | 0.49 (0.132) | - |
| Gap to approx. ub. (%) | 5.78 | 6.19 | 4.78 | 5.24 |
| Ave. (st. dev.) load factor | 0.9199 (0.0045) | 0.9197 (0.0047) | 0.9203 (0.0051) | 0.9206 (0.0054) |

#### 5.3.2.4 Sensitivity analysis on influence of demand factor

In this section, we examine the impact of different demand factors on the performance of the different approaches. This can be interesting as the decision to accept or reject specific requests becomes more critical when demand factors increase. Moreover, demand factors can differ notably per season, per year and between airlines. Therefore, information regarding the performance of different demand factors is crucial. We apply the sensitivity analysis on subnetwork 3, as it is a more mediocre subnetwork and we still have a large part of the seats open for sale. Moreover, it has a relatively low demand factor compared to subnetworks 1, 2 and 4. Therefore, it is also interesting to see how the approaches perform for subnetwork 3 when demand factors increase. A Monte Carlo simulation with a sample size of 50 is used for the different demand factors and different approaches to speed up the computation. For the original demand factor, the relative performance did not differ substantially when decreasing the sample size from 100 to 50, and, therefore, a sample size of 50 is justifiable.

Figure 2 shows the revenue impact of the different demand factors and the different approaches on subnetwork 3. Here, the realized revenue of the ROCD-DP and DD-DP approaches, the OCC benchmark approach and the approximate upper bound (UB) are expressed as a percentage difference of the realized revenue of the MPC benchmark approach. The relative performance of the different approaches for demand factor 1.4 is relatively odd. This can not be explained by the small sample size, as a sample size of 100 gave a relatively identical performance. The relative performance of OCC benchmark approach compared to the MPC benchmark approach seems to deteriorate with increasing demand factors. The relative performance of the ROCD-DP and DD-DP approaches compared to the MPC benchmark approach seems to decrease slightly when demand factors increase. Moreover, while for lower demand factors the ROCD-DP and DD-DP approach perform fairly similarly, for higher demand factors, the relative performance increase of the ROCD-DP over the DD-DP approach seems to increase.

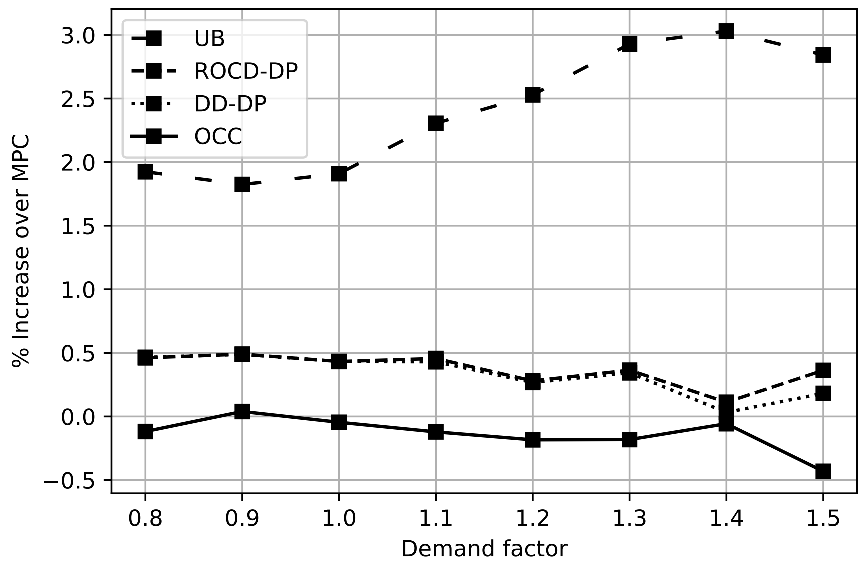


Figure 2: Comparison of the revenue impacts of the different demand factors and different approaches on subnetwork 3. The vertical order of the revenue lines and the legend is similar.

#### 5.3.2.5 Robustness analysis in the demand estimates

Until now, the decomposition model always used correct demand estimates. That is, both the demand model, as well as, the decomposition model used the same mean-demand forecasts. Although this was reasonable to assume in order to compare the different approaches, in practice, the demand estimates are rarely correct. Therefore, it is interesting to examine the robustness of the different approaches. For this, we introduce randomness within the demand estimates and measure the impact of this introduced randomness in terms of differences in obtained revenue. For this purpose, we will again use a Monte Carlo simulation to obtain reliable statistics.

The same simulation as in Section 5.1 is used, however, now for every booking period, we perturb the mean aggregate demand to come at time $t$, $\boldsymbol{\mu}(\mathrm{t})$, by drawing from a normal distribution with zero mean and variance of $\boldsymbol{\mu}(t)$. Here, the arrival of demand is still generated using the unperturbed mean aggregate demand to come at time $t$, $\boldsymbol{\mu}(\mathrm{t})$, however, the optimization uses the perturbed mean aggregate demand to come at time $t$, denoted by $\boldsymbol{\mu}'(t)$. That is, $\boldsymbol{\mu}'(t)$ is computed by Equation 55.

$$\boldsymbol{\mu}'(t) = \boldsymbol{\mu}(t) + N(0, \boldsymbol{\mu}(t)) = N(\boldsymbol{\mu}(t), \boldsymbol{\mu}(t)) \tag{55}$$

The robustness analysis is performed on subnetwork 5, which is like subnetwork 3 a more mediocre subnetwork with still a large part of the seats open for sale. On average, the demand estimates were perturbed by 0.56 standard deviations. The results of the Monte Carlo simulations with a sample size of 100 are given in Table 8. The realized revenue of the ROCD-DP and DD-DP approaches and OCC benchmark approach decreased by respectively 0.68%, 0.69% and 0.64%, which corresponds to respectively 0.11, 0.11 and 0.10 standard deviations for the different approaches. The realized revenue of the MPC benchmark approach remained almost the same. The results indicate that all approaches are quite robust, with the MPC benchmark approach being the most robust. The ROCD-DP and DD-DP approaches show a small average revenue decrease compared to the MPC benchmark approach of approximately 0.10%, corresponding on average to 0.014 standard deviations. The load factor decreased on average by 0.56% for the approaches, corresponding on average to 0.62 standard deviations. This can be explained by the fact that the optimization expected wrongfully predicted demand and, therefore, this demand was never realized.

The robustness of the different approaches might be explained by the fact that the booking limits are revised daily. That is, every day the booking limits are recalculated given the remaining capacity and the demand forecasts. The demand forecasts themselves might be off, however, the booking limits also change depending on the remaining capacity. That is, if many unpredicted products are sold, the booking limits will automatically become more restrictive due to having less remaining capacity.

Table 8: Results of the Monte Carlo simulations with a sample size of 100 on subnetwork 5 using perturbed demand estimates for the different approaches.

| Approaches | ROCD-DP | DD-DP | OCC | MPC |
|---|---|---|---|---|
| Ave. (st. dev.) revenue (€) | 289,441 (19,029) | 289,410 (19,026) | 287,944 (19,430) | 289,702 (18,873) |
| % Increase over MPC (st. dev.) | -0.09 (0.014) | -0.10 (0.015) | -0.61 (0.093) | - |
| Gap to approx. ub. (%) | 2.36 | 2.37 | 2.87 | 2.28 |
| Ave. (st. dev.) load factor | 0.8619 (0.0081) | 0.8619 (0.0081) | 0.8630 (0.0081) | 0.8653 (0.0081) |
| Ave. (st. dev.) upgrades | 0.502 (0.159) | 0.502 (0.158) | 0.468 (0.152) | 0.562 (0.158) |

#### 5.3.3 Overall summary

The case study on the entire large-scale network of the airline showed the impracticality of the OCD-DP approach and the OCC benchmark approach. The ROCD-DP and DD-DP approaches did have reasonable computation time and the Monte Carlo simulations on the eight subnetworks showed the potential of the approaches to generate additional revenue. The average increase in realized revenue goes up to approximately 0.8% compared to the two benchmark approaches. This additional revenue is achieved with lower final load factors, namely the ROCD-DP and DD-DP approaches showed an average decrease of, respectively, 0.42% and 0.09% in final load factor for the MPC and OCC benchmark approaches. Nonetheless, sensitivity analysis showed that the relative performance increase is subjective to the characteristics of the network. Namely, when demand factors increase, the relative performance increase seems to decrease slightly and in the case when the network contains more connecting flow, the relative performance increase even becomes negative. Sensitivity analyses also showed that the difference in control mechanism had less influence on the relative performance of the approaches. Also, sensitivity analysis on the influence of the sample size of the Monte Carlo simulation showed the sufficiency of a sample size of 100. Robustness analysis showed that all approaches are quite robust when perturbing the demand estimates heavily, which can be explained by the daily revision of the booking limits. Nonetheless, when we heavily perturb the demand estimates, the relative performance increase of the ROCD-DP and DD-DP approaches over the MPC benchmark approach seems to vanish.

The ROCD-DP and DD-DP approaches performed very similarly. The final configurations on the flight legs were identical and the final load factors were almost identical. Nonetheless, for all experiments, the ROCD-DP approach outperformed the DD-DP approach. This can be explained by the fact that the DD-DP approach uses the optimal dual variables as network bid prices, where the dual variables are an approximation of the opportunity cost of the corresponding flight leg-cabin capacity. The ROCD-DP approach calculates the opportunity cost of a product (and thereby of a set of flight leg-cabin capacities) directly and is, therefore, more accurate. The differences are, however, minimal and, therefore, the airline can make its choice depending on its interest. That is, depending on the available time and whether degeneracy in the DMILP model and the problem of non-additive bid prices are observed frequently, as explained in Section 4.2.1, one can choose between the ROCD-DP and DD-DP approach. There are, however, also small indications that when demand factors increase or when we consider networks with more connecting flow, the relative performance increase of the ROCD-DP over the DD-DP approach becomes larger.

# 6 Conclusion

In Section 6.1, we present the main findings. Lastly, in Section 6.2, we provide an extensive discussion on topics for further research.

## 6.1 Main findings

In this thesis, we introduced three different approaches, namely the OCD-DP, ROCD-DP and DD-DP approaches, to solve the network seat inventory control problem which incorporates aircraft configuration selection. Here, the goal was to find a control mechanism that is flight leg based instead of the traditional flight leg-cabin based, and, thereby, being a function of the remaining capacities in the cabins. It was found that a decomposition model, which uses a dynamic programming model to solve the independent single flight-legs, was very suitable for this purpose. We tested the approaches on an entire large-scale network of a major airline in Europe and on eight smaller subnetworks, which were generated using the large-scale network. Applying the approaches on the large-scale network showed the impracticality of the current implementation of the OCD-DP approach, namely the computation time of the optimization of a single booking period increased to 4.7 hours, which was approximately 4 and 7 times longer than the ROCD-DP and DD-DP approaches, respectively.

The subnetworks were sufficiently small in size to apply a Monte Carlo simulation for the booking process, which enabled us with interesting performance statistics about the stochastic network seat inventory control problem. The performance of the remaining two approaches was benchmarked against two well-known solution approaches and an approximate upper bound. The Monte Carlo simulation showed the potential of the ROCD-DP and DD-DP approaches to obtain additional revenue compared to the benchmark approaches. Namely, on average, the two approaches showed revenue increases of up to 0.8% compared to the benchmark approaches. The corresponding average gap to the approximate upper bound was equal to 2.3%, indicating that the found solutions are quite close to the optimum for the network seat inventory control problem. Nonetheless, sensitivity analysis showed that this relative performance increase is subjective to the characteristics of the network. That is, when demand factors increase, the relative performance increase seems to decrease slightly and when the network contains more connecting flow, the relative performance increase even becomes negative. Robustness analysis in the demand estimates showed that all approaches are fairly robust, indicating that the proposed approaches are of good practical value as in reality the demand estimates are seldom exact. Nonetheless, when we heavily perturb the demand estimates, the relative increase in performance of the ROCD-DP and DD-DP approaches over the MPC benchmark approach seems to vanish.

For all experiments the ROCD-DP approach outperformed the DD-DP approach, however, the differences were minimal. Nonetheless, there were small indications that the relative performance of the ROCD-DP approach over the DD-DP approach increases when demand factors increase or when the network contains more connecting flow. The ROCD-DP approach does, however, have a significantly longer computation time, namely, for the entire large-scale network the computation time of the optimization of a booking period increased from 40 minutes for the DD-DP approach to 70 minutes for the ROCD-DP approach. Therefore, depending on the interest of the airline, the choice between the ROCD-DP and DD-DP approaches must be made.

## 6.2 Further research

For further research, we propose to extend both the DMILP model and the stochastic dynamic programming model, given in Section 4.2.1 and Section 4.2.3, respectively, to incorporate overbooking, such that the proposed approaches are ready for practical use. Moreover, depending on the interest of the airline, one could incorporate demand variability within the dynamic programming model. Also, we recommend better finetuning when implementing the proposed approaches. For example, one can experiment with a finer discretization of time for the dynamic programming model. That is, to satisfy the assumption of at most one arrival per time-period, time was discretized properly. One could discretize time even further to obtain finer dynamic bid prices. This could be advantageous just before departure when remaining time and capacities are small and it could support implementations of dynamic pricing. Nevertheless, this comes at a cost of longer computation times. Contrariwise, when remaining time and capacities are still large, one could increase the length of a booking period to save computation time. That is, as the computed bid prices are now a function of remaining time, one can migrate from daily optimization to, for example, every other day when remaining time and capacities are large.

Using the same way of thinking, a fourth approach was introduced in Section 4.2.2, where the aircraft configuration is only relaxed when the remaining time is still large. Using this approach, the computational burden of the OCD-DP approach might decrease, and, therefore, it can also be interesting to test this approach.

It can also be beneficial to examine different prorating schemes. This is a very subtle task and was, therefore, considered outside the scope of this thesis. Moreover, the sensitivity analysis on the influence of the characteristics of the considered network showed that the performance of the approaches can differ drastically when more products using multiple flight legs are added. It is recommended to find out what the root cause is for the difference in performance and whether this is structural for the different approaches. Lastly, the sensitivity analysis on the influence of the characteristics of the network can also be extended. One can consider alternative models for the arrival of demand in the booking process simulation. That is, one could, as an example, use a rounded and truncated normal distribution instead of using a Poisson distribution. Also, one could change the ratio of the number of flight legs with non-fixed aircraft configuration over the entire number of flight legs and examine its influence.

# References

Adelman, D. (2007). Dynamic bid prices in revenue management. *Operations Research*, *55*(4), 647–661.

Alamdari, N. E., & Savard, G. (2021). Deep reinforcement learning in seat inventory control problem: An action generation approach. *Journal of Revenue and Pricing Management*, *20*(5), 566–579.

Alderighi, M., Nicolini, M., & Piga, C. A. (2019). Is low-cost carriers' revenue management a firm capability? *Journal of Air Transport Management*, *78*, 15–22.

An, J., Mikhaylov, A., & Jung, S.-U. (2021). A linear programming approach for robust network revenue management in the airline industry. *Journal of Air Transport Management*, *91*, 101979.

Belobaba, P. (1987). *Air travel demand and airline seat inventory management* (Doctoral dissertation). Massachusetts Institute of Technology.

Belobaba, P. (1989). Or practice—application of a probabilistic decision model to airline seat inventory control. *Operations Research*, *37*(2), 183–197.

Belobaba, P., & Hopperstad, C. (1999). Boeing.

Bertsimas, D., & De Boer, S. (2005). Simulation-based booking limits for airline revenue management. *Operations Research*, *53*(1), 90–106.

Bertsimas, D., & Popescu, I. (2003). Revenue management in a dynamic network environment. *Transportation science*, *37*(3), 257–277.

Bhatia, A., & Parekh, S. (1973). Optimal allocation of seats by fare. *Presentation by Trans World Airlines to AGIFORS Reservations Study Group.*

Birbil, Ş. İ., Frenk, J., Gromicho, J. A., & Zhang, S. (2014). A network airline revenue management framework based on decomposition by origins and destinations. *Transportation Science*, *48*(3), 313–333.

Bratu, S. (1998). *Network value concept in airline revenue management* (tech. rep.). [Cambridge, Mass.: Massachusetts Institute of Technology], Flight ...

Brumelle, S. L., & McGill, J. I. (1993). Airline seat allocation with multiple nested fare classes. *Operations research*, *41*(1), 127–137.

Calmon, A. P., Ciocan, F. D., & Romero, G. (2021). Revenue management with repeated customer interactions. *Management Science*, *67*(5), 2944–2963.

Chen, Y., Levi, R., & Shi, C. (2017). Revenue management of reusable resources with advanced reservations. *Production and Operations Management*, *26*(5), 836–859.

Curry, R. E. (1990). Optimal airline seat allocation with fare classes nested by origins and destinations. *Transportation science*, *24*(3), 193–204.

de Boer, S. V., Freling, R., & Piersma, N. (2002). Mathematical programming for network revenue management revisited. *European Journal of Operational Research*, *137*(1), 72–92.

Dror, M., Trudeau, P., & Ladany, S. P. (1988). Network models for seat allocation on flights. *Transportation Research Part B: Methodological*, *22*(4), 239–250.

D'Sylva, E. (1982). O and d seat assignment to maximize expected revenue. *Unpublished internal report, Boeing Commercial Airplane Company, Seattle, WA*.

Fuchs, S. (1987). Managing the seat auction. *Airline Bus*, *7*, 40–44.

Gerchak, Y., Parlar, M., & Yee, T. K. (1985). Optimal rationing policies and production quantities for products with several demand classes. *Canadian Journal of Administrative Sciences/Revue Canadienne des Sciences de l'Administration*, *2*(1), 161–176.

Glover, F., Glover, R., Lorenzo, J., & McMillan, C. (1982). The passenger-mix problem in the scheduled airlines. *Interfaces*, *12*(3), 73–80.

Lautenbacher, C. J., & Stidham Jr, S. (1999). The underlying markov decision process in the single-leg airline yield-management problem. *Transportation science*, *33*(2), 136–146.

Lee, A. O. (1990). *Airline reservations forecasting: Probabilistic and statistical models of the booking process* (tech. rep.). Cambridge, Mass.: Flight Transportation Laboratory, Dept. of Aeronautics and ...

Lee, T. C., & Hersh, M. (1993). A model for dynamic airline seat inventory control with multiple seat bookings. *Transportation science*, *27*(3), 252–265.

Liang, Y. (1999). Solution to the continuous time dynamic yield management model. *Transportation Science*, *33*(1), 117–123.

Lin, D., Lee, C. K. M., & Yang, J. (2017). Air cargo revenue management under buy-back policy. *Journal of Air Transport Management*, *61*, 53–63.

Littlewood, K. (1972). Forecasting and control of passenger bookings. *Airline Group International Federation of Operational Research Societies Proceedings, 1972*, *12*, 95–117.

Osadchiy, N., & Vulcano, G. (2010). Selling with binding reservations in the presence of strategic consumers. *Management Science*, *56*(12), 2173–2190.

Phillips, R. (1994). State-contingent airline yield management. *Presentation in Session TC33*, *4*.

Phillips, R. L. (1994). A marginal-value approach to airline origin & destination revenue management. *System modelling and optimization* (pp. 905–917). Springer.

Richter, H. (1982). The differential revenue method to determine optimal seat allotments by fare type. *AGIFORS PROCEEDINGS*.

Robinson, L. W. (1995). Optimal and approximate control policies for airline booking with sequential nonmonotonic fare classes. *Operations Research*, *43*(2), 252–263.

Simpson, R. (1989). Using network flow techniques to find shadow prices for market and seat inventory control. *Memorandum M89-1, Flight Transportation Laboratory, Massachusetts Institute of Technology, Cambridge, MA*.

Smith, B. C., Leimkuhler, J. F., & Darrow, R. M. (1992). Yield management at american airlines. *interfaces*, *22*(1), 8–31.

Stein, C., Truong, V.-A., & Wang, X. (2020). Advance service reservations with heterogeneous customers. *Management Science*, *66*(7), 2929–2950.

Subramanian, J., Stidham Jr, S., & Lautenbacher, C. J. (1999). Airline yield management with overbooking, cancellations, and no-shows. *Transportation science*, *33*(2), 147–167.

Talluri, K. (2008). On bounds for network revenue management. *Available at SSRN 1107171*.

Talluri, K., & Van Ryzin, G. (1998). An analysis of bid-price controls for network revenue management. *Management science*, *44*(11-part-1), 1577–1593.

Talluri, K., & Van Ryzin, G. (1999). A randomized linear programming method for computing network bid prices. *Transportation science*, *33*(2), 207–216.

Talluri, K., & Van Ryzin, G. (2004a). Revenue management under a general discrete choice model of consumer behavior. *Management Science*, *50*(1), 15–33.

Talluri, K., & Van Ryzin, G. (2004b). *The theory and practice of revenue management* (Vol. 1). Springer.

Tong, C., & Topaloglu, H. (2014). On the approximate linear programming approach for network revenue management problems. *INFORMS Journal on Computing*, *26*(1), 121–134.

Topaloglu, H. (2008). A stochastic approximation method to compute bid prices in network revenue management problems. *INFORMS Journal on Computing*, *20*(4), 596–610.

Topaloglu, H. (2009). Using lagrangian relaxation to compute capacity-dependent bid prices in network revenue management. *Operations Research*, *57*(3), 637–649.

Van Ryzin, G., & Vulcano, G. (2008). Simulation-based optimization of virtual nesting controls for network revenue management. *Operations research*, *56*(4), 865–880.

Vinod, B. (1989). A set partitioning algorithm for virtual nesting indexing using dynamic programming. *Internal Technical Report*.

Vulcano, G., Van Ryzin, G., & Chaar, W. (2010). Om practice—choice-based revenue management: An empirical study of estimation and optimization. *Manufacturing & Service Operations Management*, *12*(3), 371–392.

Williamson, E. L. (1992). *Airline network seat inventory control: Methodologies and revenue impacts* (Doctoral dissertation). Massachusetts Institute of Technology.

Wollmer, R. D. (1986). A hub-spoke seat management model. *Unpublished Internal Report, Mc Donnell Douglas Corporation, Long Beach, CA*.

Wollmer, R. D. (1992). An airline seat management model for a single leg route when lower fare classes book first. *Operations research*, *40*(1), 26–37.

Zhang, D. (2011). An improved dynamic programming decomposition approach for network revenue management. *Manufacturing & Service Operations Management*, *13*(1), 35–52.

# A Appendix

In Appendix A, we introduce additional statistics of the entire large-scale network.

## A Summary statistics of large-scale network.

The summary statistics of the 661 flight legs are given in Table 9. Note that again, by convention, the initial capacities are specified for the zero-configuration. The initial capacity in the economy cabin differs between 36 and 374 seats, with an average of 139.8 seats and a standard deviation of 71.9 seats. For the business cabin, the initial capacity differs between 4 and 38 seats, with an average of 10.5 seats and a standard deviation of 9.0 seats. As we consider the network 3 months before departure, part of the seats is already sold. In the economy cabin, between 0 and 234 seats are already sold, with an average of 24.7 seats and a standard deviation of 39.1 seats. In the business cabin, between 0 and 31 seats are sold, with an average of 2.3 seats and a standard deviation of 4.7 seats. Given the initial seats sold, the remaining capacities in both cabins can be determined, as can be seen in Table 9. Approximately 85.5% of the seats in the economy cabin are open for sale, with a standard deviation of 15.3%. For the business cabin, approximately 85.3% of the seats are open for sale, with a standard deviation of 24.7%. The forecast of demand in the economy cabin lies between 4.9 and 1376.2 seats, with an average of 126.2 seats and a standard deviation of 123.3 seats. Moreover, the forecast of demand in the business cabin lies between 0 and 113.8 seats, with an average of 6.0 seats and a standard deviation of 9.6 seats. The maximum number of upgrades lies between 0 and 36 seats, with an average of 5.7 seats and a standard deviation of 6.7 seats. Downgrading is not allowed for any of the flight legs.

Table 9: Summary statistics of the 661 flight legs of the large-scale network.

| | Min | Mean | St. dev. | Max |
|---|---|---|---|---|
| Initial capacity economy cabin | 36 | 139.818 | 71.947 | 374 |
| Initial capacity business cabin | 4 | 10.466 | 9.015 | 38 |
| Initial seats sold economy cabin | 0 | 24.744 | 39.115 | 234 |
| Initial seats sold business cabin | 0 | 2.286 | 4.723 | 31 |
| Remaining capacity economy cabin | 2 | 115.074 | 49.414 | 321 |
| Remaining capacity business cabin | -5 | 8.180 | 6.015 | 36 |
| Demand forecast economy cabin | 4.914 | 126.215 | 123.316 | 1376.216 |
| Demand forecast business cabin | 0 | 5.976 | 9.627 | 113.836 |
| Allowed upgrading | 0 | 5.731 | 6.737 | 36 |
| Allowed downgrading | 0 | 0 | 0 | 0 |

Previously mentioned remaining capacities and demand forecasts might suggest high load factors, however, this is not completely true as demand is not evenly spread across the flight legs. Figure 3 shows a histogram of the demand factors for the 661 flight legs. One can observe that the majority of the flight legs have a demand factor of less than 1, indicating lower final load factors. The average demand factor is equal to 1.0 with a standard deviation of 0.5.

For the 564 flight legs with non-fixed aircraft configuration, additional summary statistics are given in Table 10. The number of different configurations lies between 4 and 6 configurations, with an average of 4.3 configurations and a standard deviation of 0.5 configurations. Changing the configuration leads to a decrease in capacity of the economy cabin between 4 and 6 seats, with an average of 4.8 seats and a standard deviation of 1.0 seats. Moreover, this configuration change leads

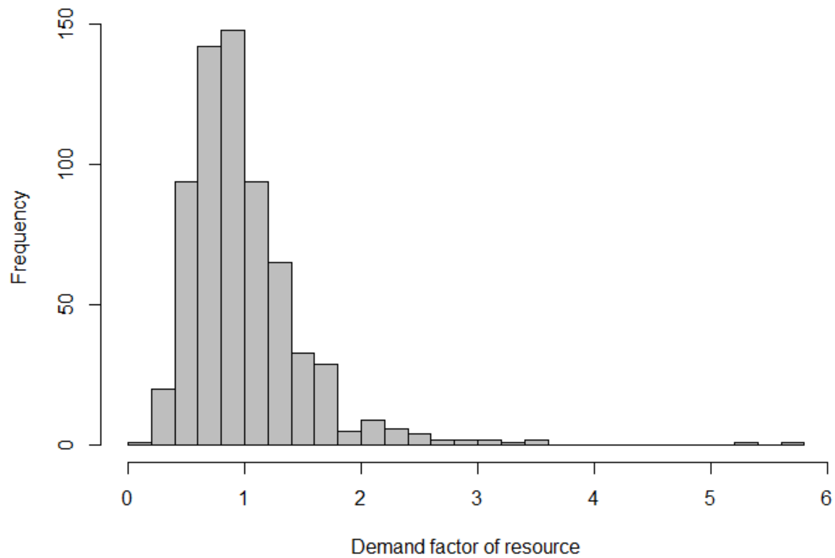


Figure 3: Histogram of the demand factors of the flight legs of the large-scale network.

to an increase in capacity of the business cabin of 4 seats, which is the same for all flight legs with non-fixed aircraft configuration.

Table 10: Summary statistics of the 564 flight legs with non-fixed aircraft configuration of the large-scale network.

| | Min | Mean | St. dev. | Max |
|---|---|---|---|---|
| Number of different configurations | 4 | 4.344 | 0.545 | 6 |
| Decrease in capacity economy cabin | 4 | 4.773 | 0.975 | 6 |
| Increase in capacity business cabin | 4 | 4 | 0 | 4 |

Lastly, the summary statistics of the 267492 different products are given in Table 11. In total 242482 products are for the economy cabin and the remaining 25010 products are for the business cabin. The 267492 different products correspond to 7505 different cabin-specific OD itineraries. The demand forecasts for the products lie between 0.0 and 154.2 seats, with an average of 0.3 seats and a standard deviation of 1.4 seats. The associated revenue of the products lies between 0.0 and 10809.35 euros, with an average of 451.3 euros and a standard deviation of 825.7 euros. Lastly, a product uses between 1 and 3 flight legs, with an average of 1.5 flight legs and a standard deviation of 0.5 flight legs.

Table 11: Summary statistics of the 267492 products of the large-scale network.

| | Min | Mean | St. dev. | Max |
|---|---|---|---|---|
| Demand forecast | 0.000 | 0.304 | 1.396 | 154.196 |
| Associated revenue (€) | 0.00 | 451.26 | 825.75 | 10809.35 |
| Number of flight legs used | 1 | 1.497 | 0.516 | 3 |