NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

OPTIMIZATION METHODS FOR REVENUE MANAGEMENT

A THESIS FULFILLED FOR THE DEGREE OF
MSC IN STATISTICS AND OPERATIONAL RESEARCH

ANASTASIA POLYCHRONOPOULOU
SUPERVISOR: ATHANASIA MANOU

DEPARTMENT OF MATHEMATICS

ATHENS, 2021

Abstract

Revenue Management is concerned with demand-management decisions, i.e., decisions about product availability and price that are taken with the objective to maximize revenue. This master thesis is on optimazation techniques that are employed in Revenue Management.

In the first part, we are dealing with the case of Single Resource Capacity Control. Our aim is to optimally allocate capacity of a single resource to various classes, namely different product categories. In order to achieve this we had developed the theory behind the cases of discrete and continuous demands, as well as some computational approaches by using "Matlab". And, also, we have analysed for different cases of demands and revenues the modification of protection levels.

Consequently, in the second part we are interested in maximizing the revenues in the case of a Network, i.e., when a company sells products that use multiple resources with limited capacity. Thus, since resources aren't independent we can't achieve this by maximizing the revenue from each resource. Again, we have developed the theory for different categories of network controls and some examples for the majority of them.

In the last part, our goal is to find out how we can increase capacity utilization, namely we are dealing with the case of Overbooking. Every airline books more seats than actual has because of the fact that there are cases of no-showing or cancellations which will cause revenue loss for the airlines. In order to analyse the case of Overbooking we have developed the legal issues concerning it and, also, the theory for different types of models and some computational approaches for finding the optimal overbooking limit.

Contents

Chapter 1

Introduction

Every seller of a product or service faces a number of fundamental decisions. You want to sell at a time when market conditions are most favourable, but who knows what the future might hold? You want the price to be right - not so high that you put off potential buyers and not so low that you lose out on potential profits. You would like to know how much buyers value your product, but more often than not you must just guess at this number. Anyone who has ever faced such decisions knows the uncertainty involved. Indeed, it is hard to find anyone who is entirely satisfied with pricing and selling decisions. Even if you succeed in making a sale, you often wonder whether you should have waited for a better offer or whether you accepted a price that was too low.

Businesses face even more complex selling decisions. For example, how can a firm segment buyers by providing different conditions and terms of trade that profitably exploit their different buying behaviour or willingness to pay? How can a firm design products to prevent cannibalization across segments and channels? Once it segments customers, what prices should it charge each segment? If the firm sells in different channels, should it use the same price in each channel? How should prices be adjusted over time based on seasonal factors and the observed demand to date for each product? If a product is in short supply, to which segments and channels should it allocate the products? How should a firm manage the pricing and allocation decisions for products that are complements (seats on two connecting airline flights) or substitutes (different car categories for rentals)?

Revenue Management is concerned with such demand-management decisions and the methodology and systems required to make them. It involves managing the firm's "interface with the market" as it were - with the objective of increasing revenues. Revenue Management can be thought of as the complement of supply-chain management, which addresses the supply decisions and processes of a firm, with the objective of lowering the cost of production and delivery.

Revenue Management has to deal with three basic demand-management categories. The first one is the structural decisions, such as, which selling format to use, which segmentation or differentiation mechanisms to use, which terms of trade to offer and so on. Secondly, businesses have to take price decisions considering product categories, as time goes by and discount over the product lifetime. Last but not least are the quantity decisions, which are whether to accept or reject an offer to buy, how to allocate output or capacity to different segments and so on.

In Chapter 2, we examine the problem of quantity-based Revenue Management for a single resource; specifically, optimally allocating capacity of a resource to different classes of demand. The units of capacity are assumed to be homogeneous and customers demand capacity only for the single resource. This is to be contrasted with the multiple-resource - or network - problems in Chapter 3, in which customers require a set of different resources (e.g. two connecting flights or a sequence of nights at the same hotel) to satisfy their demand. In reality many problems are in fact network problems, but in practice, they are still frequently solved as a collection of single-resource problems (treating the resources independently). For this reason, it is important to study single-resource Revenue Management models. Moreover, single-resource models are useful as building blocks in heuristics for the network case.

In the third Chapter, we examine quantity-based Revenue Management of multiple resources, namely network Revenue Management. This class of problems arises whenever customers buy bundles of resources in combination under various terms and conditions. Customers may require several resources at once to satisfy their needs. Thus, the lack of availability of one resource may cause a loss of demand for complementary resources. Consequently, there are interdependencies among the resources and for this reason to maximize total revenues, it becomes necessary to jointly manage the capacity controls on all resources.

Lastly, we look, a somewhat distinct subject but quiet relative to the above, overbooking. Overbooking is concerned with how much capacity to provide in the first place. Namely, its focus is increasing the total volume of sales in the presence of cancellations or no-shows. Here is important to note the difference between cancellation and no-show. A cancellation is defined as a reservation that is terminated by a customer strictly prior to the time of service, so the firm has some opportunity to compensate for a cancellation by accepting more reservations after the fact. A no-show, in contrast, occurs when a customer does not cancel his reservation but rather just fails to show up at the time of service and the firm has no opportunity to replace them.

1.1 Origins in the airline industry

Before 1970, the only service options offered by commercial airlines were first class and coach class service. Fares were identical and set by the Civil Aeronautics Board (or by the International Air Transport Association (IATA) on international flights) based on standard costs. The airline industry was deregulated in the U.S. in the mid 1970's. Shortly after the deregulation appears the problem of controlling the capacity for the discounted airfares. However, the writers Ingold and Huyton (1997) say that they had observed some kind of Revenue Management since 1920. The airlines were using statistical techniques for forecasts and mathematical methods for optimization, they had developed automatic systems which can control the tickets availability and allocate them right so the maximize their profits.

When the first innovative fares occurred, they offered travellers the option of buying a coach-class seat with a discount fare. These fares were attractive for many travellers but they came with restrictions, i.e., they required round trip travel, advance purchase and minimum stay. Even though, these fares were attractive to leisure travelers, the restrictions made them unattractive to business travellers, since they often could not meet one or more of them.

After deregulation "People Express" was one of the first airlines arise. It offers fares up to 70% below the major airlines. That was a serious threat for American Airlines. In 1975 American Airlines introduced "Super Saver" fares, which had a 7-day advance purchase requirement, minimum stay conditions and required round trip travel. After 10 years they introduced "Ultimate Super Saver" fares that required a 30-day advance purchase. People Express experienced 4 years of phenomenal growth. By August 1985, it was on the verge of bankruptcy. Obviously, People Express failed. Where did it go wrong? They had great people, tremendous value, terrific growth, but they didn't use Revenue Management correctly.

Two years after the "Ultimate Super Saver" fares, in 1987, Texas Air Corporation introduced a "Max Saver" fares. These fares had the extra restriction of being non-refundable, which became widly used practice really soon. Moreover, by the mid 1980s, instead of minimum stay, most airlines applied a "Saturday night stay" requirement. Through this restriction airlines achieved to prevent business travellers from using discount fares.

Presently, many U.S. airlines observed that even though, discount fares had significantly increased the demand, they also had reduced the revenues on some flights. Popular flights, which both leisure and business travellers want, were selling out early because of the discount tickets demand. Thus, there were no or little capacity for the full-fare travelers.

This was a widly known dilemma in the airlines industry. In order to face it, airlines introduced capacity-controlled fares-discounts that were sold only in limited quantities on each flight. Firstly, these capacity controls were set by purely informal methods, but it was obvious that more sophisticated methods were needed for setting the number of discount fares that they were going to offer. The Boeing Aircraft Company was the first one that developed a software package called the "Surplus Seat System", which predicted the demand for full-fare tickets. Moreover, internally, airlines also experimented with simple statistical methods to set their capacity controls. These efforts mark the origins of the practice of Revenue Management as we know it today.

Donald Burr was Former Chairman and CEO at People Express Airlines, which we saw, it failed. As he said, although they did a lot of things right they didn't get their hands around Revenue Management and it was extremely lethal. Additionally he said: "If starting an airline today,the number one priority on my list would be information technology. In my view, that's what drives airline revenue today more than any other factor. More than service. More than planes. More than routes.".

1.2 Types of Control

Revenue Management is applicable when the stock is fixed, the product is perishable, customers book capacity prior to "departure", the seller can offer a set of fare classes, each of which has a fixed price (at least in the short run) and also he can change the availability of fare classes over time.

Reservation systems may provide different mechanisms for controlling fare class availability. These mechanisms are typically deeply embedded in the software logic of the reservation system itself and can be quite expensive (if not impossible) to change as a result. Therefore, the control mechanism itself is frequently an important practical constraint faced when implementing a Revenue Management strategy. The most common types of controls are booking limits, protection levels and bid prices.

Definitions 1. *Booking limits are controls that limit the amount of capacity that can be sold to any particular demand class at a given point in time.*
Protection levels are in many ways equivalent to booking limits. A protection level specifies an amount of capacity to reserve (protect) for a particular demand class or set of classes.
A bid-price control sets a threshold price (which may depend on variables such as the remaining capacity or time), such that a request is accepted if its revenue exceeds the threshold price and rejected if its revenue is less than the threshold price.

Booking limits and protection levels are either partitioned or nested. A partitioned protection level is trivially equivalent to a partitioned booking limit. A partitioned booking limit logically divides the available capacity into separate blocks - one for each demand class - which can be sold only to the designated class. With a nested booking limit, the capacity available to different classes overlaps in a hierarchical manner - with higher revenue classes having availability to all the capacity reserved for lower-revenue classes (and perhaps then some). The nested booking limit for Class j is denoted by b_j. Also, in the nested case, protection levels are defined for sets of demand classes - again ordered in a hierarchical manner according to revenue. Suppose Class 1 is the highest revenue class, Class 2 the second highest, etc. Then the protection level j, denoted by y_j, is defined as the amount of capacity to save for Classes $j, j-1, .., 1$; that is, for Classes j and higher (in terms of revenue).

Nested booking limits avoid the problem of capacity being simultaneously unavailable for a high-revenue class yet available for lower-revenue classes. Most reservations systems that use booking limit controls, quite sensibly, use nested rather than partitioned booking limits for this reason.

Example 1. ***(Partitioned)***

We suppose that we have 100 *seats availiable to sell. A partitioned booking limit may set a booking limit of*

- 50 *seats for Class* 1*,*
- 20 *seats for Class* 2 *and*
- 30 *seats for Class* 3*.*

If the 50 *seats of Class* 1 *are sold, Class* 1 *would be closed regardless of how much capacity is available in the remaining Classes. This could obviously be undesirable if Class* 1 *has higher revenues than Classes* 2 *and* 3*.*

Example 2. ***(Nested)***

Again we suppose that we have 100 *seats availiable to sell, the nested booking limit could be*

- $b_1 = 100$ *on Class* 1 *(all the available capacity),*
- $b_2 = 50$ *on Class* 2 *and*
- $b_3 = 20$ *on Class* 3*.*

We would accept at most 20 *customers from Class* 3*, at most* 50 *Class* 2 *customers, but as many Class* 1 *customers as possible.*

Equivalently, suppose we set a protection level

- $y_1 = 50$ *for Class* 1*, i.e.,* 50 *units of capacity would be protected for sale only to Class* 1*,*
- $y_2 = 80$ *for Classes* 1 *and* 2 *combined and*
- $y_3 = 100$ *for Classes* 1*,* 2 *and* 3 *combined.*

Assuming that $p_1 = \$150$*,* $p_2 = \$120$*,* $p_3 = \$80$*, when there are 50 or fewer units remaining, the bid price is over* \$120 *but less than* \$150*, so only Class* 1 *demand is accepted. With* 51 *to* 80 *units remaining, the bid price is over* \$80 *but less than* \$120 *so either Class* 1 *either* 2 *are accepted. With more than* 80 *units of capacity available, the bid price drops below* \$80 *so all three classes are accepted.*

The booking limit for Class j, b_j, is simply the capacity minus the protection level for Classes $j-1$ and higher. That is, $b_j = C - y_{j-1}$, $j = 2, .., n$ where C is the capacity. For convenience, we define $b_1 = C$ (e.g. the highest revenue class has a booking limit equal to the capacity) and $y_n = C$ (all classes have a protection level equal to capacity). Moreover, bid prices can usually be used to implement the same nested allocation policy as booking limits and protection levels. Also, the bid price $\pi(x)$ is plotted as a function of the remaining capacity x.

To be effective bid prices must be updated after each sale - and possibly also with time as well - and typically this requires storing a table of bid-price values that can be indexed based on the current available remaining capacity, current time or both. If the bid price is a function of the current remaining capacity, then it performs exactly like a booking limit or protection level,closing off capacity to successively higher revenue classes as capacity is consumed, otherwise the system will sell an unlimited amount of capacity to any class whose revenues exceed the bid-price threshold.

One potential advantage of bid-price controls is that if actual revenue information is available for a request and there are multiple revenue values in a fare class, then a bid-price control can selectively accept only the higher revenue requests in a class, whereas a control based on class designation alone can at best simply accept or reject all requests in a class.

In order to propose a way to determine optimal controls, we need to introduce the notion of displacement or opportunity cost.

Definition 1. *Displacement cost - or opportunity cost - is the expected losses in future revenue from using the capacity now rather than using it in the future.*

Example 3. *Suppose we have* 100 *seats availiable and only two classes*

- *a full-fare class (Class* 1*) which costs* \$180 *and*
- *a discount-fare class (Class* 2*) which costs* \$70.

Also, we know for certain that demand for Class 1 *will be 30 and the demand for Class* 2 *will be 85.*
Ideally, since Class 1 *generates bigger revenues, we would like to sell* 30 *seats to Class* 1 *and sell the remaining seats to Class* 2*, i.e., we would set a booking limit of* 70 *for Class* 2 *or a protection level of* 30 *for Class* 1*.*
Another way to decide how to allocate capacity is by evaluating the displacement cost of each seat. Since, Class 2 *customers generate less revenue, they are the best displacement option. Namely, the displacement cost of the* $100^{th}, 99^{th}, .., 21^{st}$ *seat is the revenue of a Class* 2 *customer.*
If we have 15 *units of capacity remaining the displacement cost is* \$180*, so if we receive a request for the discounted fare we would reject it, because its revenue is less than the displacement cost of capacity. On the other hand, with* 50 *seats remaining the displacement cost is only* \$70*, so a request for a discount-fare would be accepted.*

At each point in time, we can evaluate the decision of whether to accept a request by simply comparing the revenue of the request to the displacement cost. The displacement cost is easy to compute if demand is known for certain,but much more complex to compute when demand is uncertain. To calculate displacement cost we use value function

Definition 2. *A value function,* $V(x)$*, measures the optimal expected revenue as a function of the remaining capacity* x*.*

The displacement cost is then the difference between the value function at x and the value function at $x-1$, or $V(x) - V(x-1)$.

Chapter 2

Single Resource Capacity Control

2.1 Introduction

In this section we will discuss the problem of allocating the capacity of a single resource to various demand classes. For example, controlling the availability of discounted seats on a single flight leg or different rate classes for hotel rooms on a single night. The units of capacity are assumed to be homogeneous and customers demand capacity only for the single resource. We will assume the sales conditions of the single resource have been differentiated to form n distinct products - or classes - as they are traditionally called in the airline context. In most cases, we will assume that these products appeal to different segments of the market and effectively segment the market into n classes, one for each product. The central problem of the chapter is how to optimally allocate the capacity of the single-resource to these various demand classes. This allocation must be done, typically, in a dynamic fashion as demand arrives and with considerable uncertainty about the demand that will be arriving in the future.

2.2 Static Models

In this section, we examine one of the first models for quantity-based Revenue Management, the static single-resource model. The static model makes several assumptions that are worth examining in some detail.

The first is that demand for the different classes arrives in non-overlapping intervals in the order of increasing prices of the classes. For example, in the airline case advance-purchase discount demand typically arrives before full-fare coach demand, so this assumption is a reasonable approximation. Moreover, the strict low-before-high assumption represents the worst case, for instance, if high-revenue demand arrives before low-revenue demand, the problem is trivial because we simply accept demand first come.

The second main assumption is that the demands for different classes are independent random variables. Largely, this assumption is made for analytical convenience. To make some justification of the assumption note that to the extent that there are systematic factors affecting all demand classes, these are often reflected in the forecast and become part of the explained variation in demand in the forecasting model. A potential weakness of the independence assumption is the residual, unexplained variation in demand.

A third assumption is that demand for a given class does not depend on the availability of other classes. Its only justification is if the multiple restrictions associated with each class are so well designed that customers in a high revenue class will not buy down to a lower class and if the prices are so well separated that customers in a lower class will not buy up to a higher class if the lower class is closed. The assumption that demand does not depend on the capacity controls is a weakness, because the price differences between the classes are rarely that dispersed.

Fourth, the static model assumes an aggregate quantity of demand arrives in a single stage and the decision is simply how much of this demand to accept. Yet in a real reservation system, we typically observe demand sequentially over time, or it may come in batch downloads. However, fortunately, the form of the optimal control is not sensitive to this assumption and can be applied quite independently of how the demand is realized within a period.

A fifth assumption of the model is that either there are no groups, or if there are group bookings, they can be partially accepted.

Finally, the static models assume risk-neutrality. This is a reasonable assumption in practice, since a firm implementing Revenue Management typically makes such decisions for a large number of products sold repeatedly. Maximizing the average revenue, therefore, is what matters in the end.

2.3 The two-class problem

In this subsection we consider the case of 2 classes, First, we examine the discrete demand case. Then we consider the case of continuous demand and finally the special case of normally distributed demand.

2.3.1 Discrete Demand

Suppose we have a flight with a limited number of C seats. There are two classes of customers: the discount customers who book early and the full-fare customers who book later. Note that all discount booking requests occur before any full-fare passengers seek to book. Discount customers pay $p_d > 0$ and full-fare customers pay $p_f > p_d$. Discount demand is a random variable D_d with c.d.f. (cumulative distribution function) $F_d(x) = Pr[D_d \leq x]$ and full-fare demand is a random variable D_f with c.d.f. $F_f(x) = Pr[D_f \leq x]$. The central problem is to decide how much of discount customers demand to accept prior to the realization of full-fare customers demand, i.e., to find the optimal booking limit b_2^*, or, equivalently, to decide how many units of capacity to keep for full-fare customers, i.e., to find the protection level y_1^* ($y_1^* = C - b_2^*$).

The challenge of capacity allocation is to balance the risks of spoilage and dilution to maximize expected revenue. Setting a booking limit might cause one of the following problems. On the one hand, if the booking limit setted too low, that will turn away discount customers but not see enough full-fare demand to fill the plane (spoilage), on the other hand if the booking limit setted too high, that runs the risk of turning away full-fare customers (dilution).

To find the optimal decision we can think as follows: Suppose we have x units of capacity remaining and receive a request from discount customers. If we accept the request, we collect revenues of p_d, but if we do not accept it, we will sell seat x at p_f iff demand for full fare is x or higher ($D_f \geq x$). Thus, the expected gain from reserving the x^{th} seat is $p_f Pr[D_f \geq x]$. Therefore, it makes sense to reject a discount request as long as the current revenue does not exceed this marginal value, i.e., $p_d \leq p_f Pr[D_f \geq x]$.

Note the right-hand side of $p_d \leq p_f Pr[D_f \geq x]$ is decreasing in x, namely as the remaining capacity increases, the probability that full-fare demand is greater than or equal to x decreases. Thus, the optimal protection level is

$$y_1^* = \max\{y \in \{0, 1, ..., C\} : p_d \leq p_f Pr[D_f \geq y]\}.$$

Since, $b_2^* = C - y_1^*$, the optimal booking limit is

$$b_2^* = \min\{b \in \{0, 1, ..., C\} : p_d \leq p_f Pr[D_f \geq C - b]\}.$$

The results are summarized in the following theorem.

Theorem 1. *In the capacity allocation problem with 2 classes, the optimal booking limit is*

$$b_2^* = \min\{b \in \{0, 1, ..., C\} : p_d \leq p_f Pr[D_f \geq C - b]\},$$

and the optimal protection level is

$$y_1^* = C - b_2^* = \max\{y \in \{0, 1, ..., C\} : p_d \leq p_f Pr[D_f \geq y]\}.$$

2.3.2 Continuous Demand

Next corollary gives the optimal protection level and the optimal booking limit when demand is continuous.

Corollary 1. *When D_f is continuous random variable, the optimal booking limit is given by $b_2^* = \max\{0, \lceil b^* \rceil\}$, where b^* is the solution of*

$$p_d = p_f Pr[D_f \geq C - b],$$

and the optimal protection level is given by $y_1^* = \min\{C, \lfloor y^* \rfloor\}$, where y^* is the solution of

$$p_d = p_f Pr[D_f \geq y]. \tag{2.1}$$

The last equation is known as Littlewood's rule.

A moment of reflection shows that the optimal protection level:

1. Is independent of the distribution of discount demand, but it depends on the distribution of full-fare demand.

2. Does not depend on the capacity.

3. Depends on the two fares through their ratio $\frac{p_d}{p_f}$. Specifically, the optimal protection level decreases as the ratio increases.

Also, the optimal booking limit:

1. Is independent of the distribution of discount demand, but it depends on the distribution of full-fare demand.

2. Depends on the capacity.

3. Depends on the two fares through their ratio $\frac{p_d}{p_f}$. Specifically, the optimal booking limit increases as the ratio increases. This makes sense because while the ratio increase this means that the discount fare is close to full fare, so there is no need for more seats at full fare since it cost approximately the same money as the discount fare.

2.3.3 Independent Normal Demands

To gain additional insight into Littlewood's rule (2.1), it is useful to examine the case of normally distributed demand.

If the demand for full fare is normally distributed with mean μ_f and standard deviation σ_f, Littlewood's rule (2.1) gives:

$$\begin{aligned}
p_d &= p_f Pr[D_f \geq C - b] \\
\frac{p_d}{p_f} &= 1 - Pr[D_f \leq C - b] \\
Pr[D_f \leq C - b] &= 1 - \frac{p_d}{p_f} \\
Pr\left[\frac{Df - \mu_f}{\sigma_f} \leq \frac{C - b - \mu_f}{\sigma_f}\right] &= 1 - \frac{p_d}{p_f} \\
Pr\left[Z \leq \frac{C - b - \mu_f}{\sigma_f}\right] &= 1 - \frac{p_d}{p_f}
\end{aligned}$$

$$
\begin{aligned}
\Phi\left(\frac{C-b-\mu_f}{\sigma_f}\right) &= 1-\frac{p_d}{p_f} \\
\frac{C-b-\mu_f}{\sigma_f} &= \Phi^{-1}\left(1-\frac{p_d}{p_f}\right) \\
b &= C-\sigma_f\Phi^{-1}\left(1-\frac{p_d}{p_f}\right)-\mu_f,
\end{aligned}
$$

where Z is normally distributed with mean 0 and standard deviation 1 (Standard Normal distribution). Hence, $b^* = C-\mu_f-\sigma_f\Phi^{-1}(1-\frac{p_d}{p_f})$ and $b_2^* = \max\{0, \lceil b^* \rceil\}$. Also, $y^* = \mu_f + \sigma_f\ \Phi\ ^{-1}(1-\frac{p_d}{p_f})$ and $y_1^* = \min\{\lfloor y^* \rfloor, C\}$. Thus, we reserve enough capacity to meet the mean demand for full-fare customers (μ_f) plus or minus a factor that depends both on the fare ratio and the demand variation σ_f.

At first, we examine the effect of $\frac{p_d}{p_f}$ on the protection level.

• If $\frac{p_d}{p_f} = 0.5$, then the optimal is to set the protection level equal to the expected full-fare demand ($y = \mu_f$).

• If $\frac{p_d}{p_f} > 0.5$, the optimal is to protect fewer seats than the expected full-fare demand.

• If $\frac{p_d}{p_f} < 0.5$, the optimal is to protect more seats than the expected full-fare demand.

The inequality $\frac{p_d}{p_f} > 0.5$ means that the difference between p_d and p_f is relatively small, i.e., the discount fare is quite big, so the airline doesn't lose a big amount of money by selling at the discount customers. In general, the lower the ratio $\frac{p_d}{p_f}$, the more capacity we reserve for full-fare. This makes intuitive sense, because we should be willing to take very low prices only when the chances of selling at a high price are remote.

Next, we examine the effect of σ_f on the protection level.

• If $\sigma_f = 0$ the optimal is to set the protection level equal to the expected full-fare demand.

• If $\sigma_f > 0$, we have the following cases:

  • If $\frac{p_d}{p_f} = 0.5$, then y_s is independent of σ_f ($\Phi\ ^{-1}(0.5) = 0$). As the uncertainty for the full-fare demand increases the protection level remains equal to the expected full-fare demand.

  • If $\frac{p_d}{p_f} > 0.5$, then y_s is decreasing in σ_f. As the uncertainty for the full-fare demand increases the protection level decreases.

- If $\frac{p_d}{p_f} < 0.5$, then y_s is increasing in σ_f. As the uncertainty for the full-fare demand increases the protection level increases.

```
function [y] = littlewoodnorm(mf,sf,pd,pf,C)
  %Littlewood function for normal demand
  %y optimal protection level
  %mf mean and sf variation for full-fare
  %discount customers pay pd
  %full-fare customers pay pf
  %C capacity
  y=mf+sf*norminv(1-pd/pf);
endfunction
y=floor(min(y,C))
```

Figure 2.1: Littlewood function for Normal demand

Below we see Figure 2.2 that shows how y^* changes as σ_f increases for fixed μ_f. The blue line is for $\dfrac{p_d}{p_f} < 0.5$, the red one for $\dfrac{p_d}{p_f} = 0.5$ and the orange for $\dfrac{p_d}{p_f} > 0.5$.

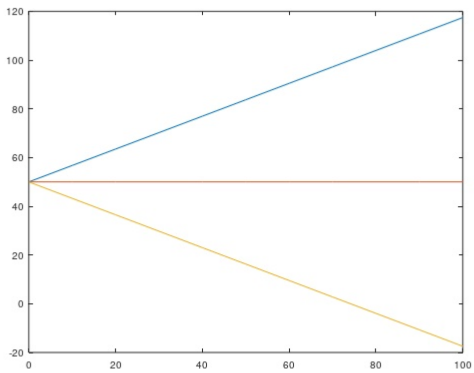


Figure 2.2: The relation of y^* and σ_f for fixed μ_f and different $\dfrac{p_d}{p_f}$

2.4 n-Class Models

Now we will expand our research in the case that we have n classes, $n \geq 2$. We assume that demand for the n classes arrives in stages in increasing order of their revenue values. We express the fee of class i as p_i, so that $p_1 > p_2 > ... > p_n$. Hence, class n (the lowest price) demand arrives in the first stage (stage n), followed by class $n-1$ demand in stage $n-1$ and so on, with the highest price class (class 1) arriving in the last stage (stage 1). Moreover, as D_j we denote the demand at stage j. Demand and capacity are most often assumed to be discrete, but occasionally we model them as continuous variables when it helps simplify the analysis and optimality conditions.

2.4.1 Discrete Case

In the case that we have discrete demand and capacity we can formulate this problem by using a dynamic program in the stages (equivalently, classes). The remaining capacity x is the state variable. At the start of each stage j we do not know the demand $D_j, D_{j-1}, ..., D_1$. Thus, at stage j, we know the remaining capacity x, the demand D_j and only **the distribution** of the demand for the remaining stages $D_{j-1}, ..., D_1$. By using these information we have to decide how much of D_j demand we will accept.

The following theorem gives Bellman equation for the problem with n classes.

Theorem 2. ***(Bellman equation)*** *Let $V_j(x)$ denote the value function at the start of stage j. That is, the maximum expected revenue that can be obtained starting in stage j with x units of capacity remaining. The Bellman equation is:*

$$V_j(x) = E\left[\max_{0 \le u \le \min\{D_j, x\}} \{p_j u + V_{j-1}(x-u)\}\right] \tag{2.2}$$

with boundary conditions $V_0(x) = 0$, $x = 0, 1, ..., C$.

Proof. Within stage j, the realization of the demand D_j occurs and we observe its value. We decide on a quantity u of this demand to accept. The amount accepted must be less than the capacity remaining, so $u \le x$. The optimal control u^* is therefore a function of the stage j, the capacity x, and the demand D_j. Thus, $u^* = u^*(j, x, D_j)$, though we often suppress this explicit dependence in what follows. The revenue $p_j u$ is collected, and we proceed to the start of the stage $j-1$ with a remaining capacity of $x - u$.
Once the value D_j is observed, the value of u is chosen to maximize the current stage j revenue plus the revenue to go, or $p_j u + V_{j-1}(x-u)$ subject to the constraint $0 \le u \le \min\{D_j, x\}$. The value function entering stage j, $V_j(x)$, is then the expected value of this optimization with respect to the demand D_j.

□

We first consider the case where demand and capacity are discrete. To analyse the form of the optimal control in this case, define

$$\Delta V_j(x) = V_j(x) - V_j(x-1),$$

which is the marginal value of capacity at j stage. A key result concerns how these marginal values change with capacity x and the stage j.

The following lemma helps us find the monotonicity of $\Delta V_j(x)$ with respect to x.

Lemma 1. *Suppose $g : Z_+ \to R$ is concave. Let $f : Z_+ \to R$ be defined by*

$$f(x) = \max_{a=0,1,\ldots,m} \{ar + g(x-a)\}$$

for any given $r \geq 0$ and non-negative integer $m \leq x$. Then $f(x)$ is concave in $x \geq 0$ as well.

Proof. We define $y = x - a$, then we have $f(x) = \widehat{f}(x) + rx$, where

$$\widehat{f}(x) = \max_{x-m \leq y \leq x} \{-yr + g(y)\}.$$

We know that if $\widehat{f}(x)$ is concave, then so is $f(x)$. To analyse $\widehat{f}(x)$, let y^* denote the value that attains the *max* in $\max_{y \geq 0}\{-yr + g(y)\}$. Since $g(x)$ is concave, $-yr+g(y)$ is also concave and moreover it is non-decreasing for values of $y \leq y^*$ and non-increasing for values of $y > y^*$. Therefore,

$$\widehat{f}(x) = \begin{cases} -xr + g(x) & x \leq y^* \\ y^* r + g(y^*) & y^* \leq x \leq y^* + m \\ -(x-m)r + g(x-m) & x \geq y^* + m. \end{cases}$$

Therefore, in the range $x \leq y^*$ and using the fact that $g(x)$ is concave

$$\begin{aligned} \widehat{f}(x+1) - \widehat{f}(x) &= -r + g(x+1) - g(x) \\ &\leq -r + g(x) - g(x-1) \\ &= \widehat{f}(x) - \widehat{f}(x-1). \end{aligned}$$

For $y^* \leq x \leq y^* + m$, it follows that $\widehat{f}(x+1) - \widehat{f}(x) = 0$ so $\widehat{f}(x)$ is trivially concave in this range.

Finally, for $x \geq y^* + m$, from the concavity of $g(x)$

$$\widehat{f}(x+1) - \widehat{f}(x) = -r + g(x+1-m) - g(x-m)$$

$$\begin{aligned} &\leq -r + g(x-m) - g(x-1-m) \\ &= \widehat{f}(x) - \widehat{f}(x-1). \end{aligned}$$

Thus, $\widehat{f}(x)$ is concave in $x \geq 0$ and since $f(x) = \widehat{f}(x) + rx$, $f(x)$ is concave in $x \geq 0$ as well.

□

Proposition 1. *The marginal values $\Delta V_j(x)$ of the value function $V_j(x)$ $\forall x$, j satisfy:*

$$\Delta V_j(x+1) \leq \Delta V_j(x),$$

that is at a given stage j the marginal value is decreasing in the remaining capacity.

Proof. We should show that $V_j(x)$ is concave in x, $\forall j$. We know that for the terminal stage (stage 0), $V_0(x) = 0$ $\forall x$, which is trivially concave. Now assume that $V_{j-1}(x)$ is concave in x and consider

$$V_j(x) = E\left[\max_{0 \leq u \leq \min\{D_j, x\}} \{p_j u + V_{j-1}(x-u)\}\right].$$

The inner maximization for any realization of D_j is concave function in x (Lemma 1: for $r = p_j \geq 0$ and $m = \min\{D_j, x\} \leq x$). Since $V_j(x)$ is a weighted average of concave functions, it follows that $V_j(x)$ is concave as well.

□

Corollary 2. *We can write*

$$V_{j+1}(x) = V_j(x) + E\left[\max_{0 \leq u \leq \min\{D_{j+1}, x\}} \sum_{z=1}^{u} (p_{j+1} - \Delta V_j(x+1-z))\right], \; j = 1, .., n-1,$$

where we take the sum above to be empty if $u = 0$.

Proof. From Bellman equation (2.2) for the $j+1$ stage we have:

$$\begin{aligned} V_{j+1}(x) &= E\left[\max_{0 \leq u \leq \min\{D_{j+1}, x\}} \{p_{j+1} u + V_j(x-u)\}\right] \\ &= E\left[\max_{0 \leq u \leq \min\{D_{j+1}, x\}} \{p_{j+1} u + V_j(x-u) - V_j(x) + V_j(x)\}\right]. \end{aligned}$$

$V_j(x)$ is independent of u, so we can move it out of the average. Hence, we have:

$$V_{j+1}(x) = V_j(x) + E\left[\max_{0 \leq u \leq \min\{D_{j+1}, x\}} \{p_{j+1} u + V_j(x-u) - V_j(x)\}\right].$$

We know that $\Delta V_j(x) = V_j(x) - V_j(x-1)$.
Moreover, we have

$$\begin{aligned}
V_{j+1}(x) &= V_j(x) + E\left[\max_{0\le u\le \min\{D_{j+1},x\}} \sum_{z=1}^{u}(p_{j+1} - \Delta V_j(x+1-z))\right] \\
&= V_j(x) + E\left[\max_{0\le u\le \min\{D_{j+1},x\}} \{p_{j+1}u - \sum_{z=1}^{u}\Delta V_j(x+1-z)\}\right] \\
&= V_j(x) + E\left[\max_{0\le u\le \min\{D_{j+1},x\}} \{p_{j+1}u - \Delta V_j(x) - \Delta V_j(x-1) - ... - \Delta V_j(x+1-u)\}\right] \\
&= V_j(x) + E\Bigg[\max_{0\le u\le \min\{D_{j+1},x\}} \{p_{j+1}u - V_j(x) + V_j(x-1) - V_j(x-1) + V_j(x-2) \\
&- V_j(x-2) + V_j(x-3) - ... - V_j(x+1-u) + V_j(x-u)\}\Bigg] \\
&= V_j(x) + E\left[\max_{0\le u\le \min\{D_{j+1},x\}} \{p_{j+1}u - V_j(x) + V_j(x-u)\}\right].
\end{aligned}$$

□

Theorem 3. *For the static model with* n *classes, the optimal protection levels are given by*

$$y_j^* = \max\{u : p_{j+1} < \Delta V_j(u)\},\ j = 1,..,n-1 \textit{ and } y_n \equiv C.$$

Also, the nested booking limits are given by

$$b_j^* = C - y_{j-1}^*,\ j = 2,...,n \textit{ and } b_1 \equiv C.$$

Proof. Proposition 1 gives that $\Delta V_j(x)$ is decreasing in x. So, $\Delta V_j(x+1-z)$ is increasing in z. Thus, $p_{j+1} - \Delta V_j(x+1-z)$ is decreasing in z, that is while z increases it decreases until it becomes negative.
We would like to maximize $\sum_{z=1}^{u}(p_{j+1} - \Delta V_j(x+1-z))$,thus we would like to add only positive values. Therefore, the optimal policy is

$$u^* = \max\{u \le \min\{D_{j+1}, x\} : p_{j+1} - \Delta V_j(x+1-u) \ge 0\}$$

or, equivalently,

$$u^* = \min\{u \le \min\{D_{j+1}, x\} : p_{j+1} - \Delta V_j(x-u) < 0\}.$$

Hence, the optimal protection level is

$$y_j^* = \max\{u : p_{j+1} < \Delta V_j(u)\}.$$

The equation for optimal booking limits is apparent from their definition.

$\square$

In other words, the optimal control in stage $j+1$, when the remaining capacity is x seats, is to keep accepting demand (give the u-th seat) as long as $p_{j+1} \geq \Delta V_j(x+1-u)$ and stop accepting once this condition is violated or the demand D_{j+1} is exhausted, whichever comes first.

We can see that, similarly to the $n = 2$ case, that the optimal protection level y_j^*:

1. Depends on the distribution of demand only at stages $j, j-1, j-2, ..., 1$.

2. Depends on the fares $p_{j+1}, p_j, p_{j-1}, ..., p_1$.

3. Is independent of the capacity.

Comment 1. *We derive the optimal control u^* "as if" the decision on the amount to accept is made after knowing the value of demand. In reality, of course, demand arrives sequentially over time, and the control decision has to be made before observing all the demand. However, it turns out that optimal decisions do not use the prior knowledge as we show below. Hence, the assumption that is known is not restrictive.*

Comment 2. *Since ΔV_j is decreasing in x, if $p_1 > p_2 > .. > p_n$, then the protection levels are ordered $y_1^* \leq y_2^* \leq ... \leq y_n^*$.*

2.4.2 Continuous Case

Next, consider the case where capacity is continuous and demand at each stage has a continuous distribution. In this case, the dynamic program is still given by

$$V_j(x) = E\left[\max_{0 \leq u \leq \min\{D_j, x\}} \{p_j u + V_{j-1}(x-u)\}\right],$$

however D_j, x and u are now continuous quantities. The analysis of the dynamic program is slightly more complex than it is in the discrete-demand case, but many of the details are quite similar. Hence, we only briefly describe the key differences.

The main change is that the marginal value ΔV_j is now replaced by $\frac{\partial}{\partial x} V_j(x)$.

This derivative is still interpreted as the marginal expected value of capacity and is decreasing in x, equivalently $V_j(x)$ is concave in x. Similarly to the discrete case, the optimal control in stage $j+1$ is to keep increasing u as long as $p_{j+1} \geq \frac{\partial}{\partial x} V_j(x-u)$ and to stop accepting once this condition is violated or the demand D_{j+1} is exhausted, whichever comes first.

One of the chief virtues of the continuous model is that it leads to simplified expressions for the optimal vector of protection levels.

First, for an arbitrary vector of protection levels y, define the following $n-1$ fill events

$$A_j(y, D) = \{D_1 > y_1, D_1 + D_2 > y_2, ..., D_1 + ... + D_j > y_j\},\ j = 1, ..., n-1.$$

Then $A_j(y, D)$ is the event that demand to come in stages $1, 2, ..., j$ exceeds the associated protection levels. A necessary and sufficient condition for y^* to be an optimal vector of protection levels is that it satisfies the $n-1$ equations

$$P(A_j(y^*, D)) = \frac{p_{j+1}}{p_1},\ j = 1, 2, ..., n-1.$$

That is, fill event A_j should occur with probability equal to the ratio of Class $j+1$ revenue to Class 1 revenue. As it should, this reduces to Littlewood's rule in the $n = 2$ case, since

$$P(A_1(y^*, D)) = P(D_1 > y_1^*) = \frac{p_2}{p_1}.$$

Note that $A_j(y, D) = A_{j-1}(y, D) \cap \{D_1 + ... + D_j > y_j\}$, so $A_j(y, D)$ can occur only if $A_{j-1}(y, D)$ occurs. Also, if $y_j = y_{j-1}$ then $A_j(y, D) = A_{j-1}(y, D)$. Thus, if $p_j < p_{j-1}$, we must have $y_j^* > y_{j-1}^*$ in order to satisfy $P(A_j(y^*, D)) = \frac{p_{j+1}}{p_1}$, $j = 1, 2, ..., n-1$. Thus, the optimal protection levels are strictly increasing in j if the revenues are strictly decreasing in j.

2.5 Computational Approaches

At first glance it may appear that the optimal nested allocations are difficult to compute. However, computing these values is in fact quite easy and efficient algorithmically. There are two basic approaches: *dynamic programming and Monte Carlo integration.*

2.5.1 Dynamic Programming

The first approach is based on using the dynamic programming recursion

$$V_j(x) = E\left[\max_{0\le u\le \min\{D_j,x\}}\{p_j u + V_{j-1}(x-u)\}\right]$$

directly and requires that demand and capacity are discrete, or in the continuous case that these quantities can be suitably discretized. The inner optimization is simplified by expressing it with respect to y^*_{j-1} from the previous stage.

We know that $u^*(j,x,D_j) = \min\{(x-y^*_{j-1})^+, D_j\}$. Thus, substituting $u^*(j,x,D_j)$ into Bellman equation we obtain the recursion:

$$V_j(x) = E[p_j \min\{D_j,(x-y^*_{j-1})^+\} + V_{j-1}(x-\min\{D_j,(x-y^*_{j-1})^+\})]$$

where $y^*_j = \max\{x : p_{j+1} < \Delta\ V_j(x)\}$, $j = 1,2,..,n-1$ and we define $y^*_0 = 0$. This procedure is repeated starting from $j = 1$ and working backward to $j = n$.

For discrete demand distributions, computing the expectation above for each state x requires evaluating at most $O(C)$ terms since $\min\{D_j,(x-y^*_{j-1})^+\} \le C$. Since there are C states (capacity levels), the complexity at each stage is $O(C^2)$. The critical values y^*_j can then be identified in $log(C)$ time by binary search since $\Delta\ V_j(x)$ is non-increasing. Indeed, since we know $y^*_j \ge y^*_{j-1}$, the binary search can be further constrained to values in the interval $[y^*_{j-1}, C]$. Therefore, computing y^*_j does not add to the complexity at stage j. Since these steps must be repeated for each of the $n-1$ stages (stage n need not be computed as mentioned above), the total complexity of the recursion is $O(nC^2)$.

Figure 2.3 shows a matlab function that applies dynamic programming approach. Now we adduce some examples for the dynamic program.

Example 4. *We consider the following demand matrices:*

$$D_1 = \begin{bmatrix} 0&0&0&0&0.2&0.2&0.2&0.2&0.2&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0.2&0.2&0.2&0.2&0.2&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0.2&0.2&0.2&0.2&0.2&0&0&0&0&0&0&0&0 \end{bmatrix}$$

$$D_2 = \begin{bmatrix} 0&0&0&0&0&0&0&0&0.2&0.2&0.2&0.2&0.2&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0.2&0.2&0.2&0.2&0.2&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0.2&0.2&0.2&0.2&0.2&0&0&0&0&0&0&0&0 \end{bmatrix}$$

$$D_3 = \begin{bmatrix} 0&0&0&0&0.2&0.6&0.2&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0.3&0.5&0.2&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0.1&0.7&0.2&0&0&0&0&0&0&0&0&0 \end{bmatrix}$$

$$D_4 = \begin{bmatrix} 0&0&0.1&0.1&0.15&0.3&0.15&0.1&0.1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0.1&0.1&0.1&0.1&0.2&0.15&0.1&0.1&0.05&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0.05&0.05&0.2&0.5&0.1&0.05&0.05&0&0&0&0&0&0&0 \end{bmatrix}$$

```
function [y,V] = dynamic(D,C,p,n)
  %function[y,V]=dynamic(D,C,p,n) computes a vector of optimal
  %protection levels y and a matrix with values V_{j}(x)
  %inputs
  %D: nx(C+1) matrix. The jth row gives the pmf for the demand of jth class
  %C: total capacity
  %p: nx1 matrix that gives the fees for each class
  %n: number of classes
  %outputs
  %y: nx1 matrix with the optimal protection levels
  %V: (n+1)x(C+1) matrix V(j+1,x+1)=V_{j}(x)
  y(1,1)=0; %y_{0}^{*}=0
  V=zeros(n+1,C+1); %V_{0}(x)=0
  for j=1:(n-1)
    for x=0:C
      %remaining units of capacity
      for k=0:C
        V(j+1,x+1)=V(j+1,x+1)+D(j,k+1)*(p(j)*min(k,max(x-y(j,1),0))+V(j,x+1-min(k,max(x-y(j,1),0))));
      endfor
    endfor
    x=1;
    while p(j+1)<V(j+1,x+1)-V(j+1,x)
      x=x+1;
    endwhile
    y(j+1,1)=x-1;
  endfor
endfunction
```

Figure 2.3: The function of the Dynamic Programming

Applying the algorithm for different sets of parameters we obtain the following results:

n	C	D	p	y^*
3	30	D_1	[110 100 70]′	[4 19 30]′
3	30	D_1	[150 100 70]′	[5 20 30]′
3	30	D_1	[250 100 70]′	[6 20 30]′
3	30	D_1	[350 100 70]′	[7 21 30]′
3	30	D_1	[350 335 70]′	[4 22 30]′
3	30	D_1	[350 80 70]′	[7 20 30]′
3	30	D_2	[110 100 70]′	[8 23 30]′
3	30	D_3	[110 100 70]′	[4 19 30]′
3	30	D_4	[110 100 70]′	[2 18 30]′

As we can see in every case we have kept the same capacity and the same number of classes. In the first six cases, we have, also, kept the same demand matrix.

1. *In the second case, comparing to the first one, we have increased p_1, which means that we have a more expensive first-class fee. As we see we get bigger protection levels for class 1 and for class 1 and 2, which is logical since if a first-class fee is much more expensive than the other ones we want to sell them in order to have bigger profit. Moreover, as we can see in the third and the fourth line of the matrix if we increase the value of p_1 even more we get even bigger protection levels.*

2. *In the fifth case, comparing to the first one, we have increased both p_1, p_2, but we have keep their values close enough. We can see that the protection level of classes 1,2 has been increased, but the protection level for class 1 remains the same, which is logical since their values are close enough we will have no big difference in the profit if we sell fees of class 1 or 2. Then in the sixth case, we see that if we decrease p_2 and keep it close to p_3 the protection level y_1 increases, but the protection level y_3 decreases, since now we want to sell more fees from class 1.*

Now in the last three cases we have kept the same capacity and the number of classes, but we have also kept the same fees for each class.

3. *In case 7, we have the D_2 matrix for demand, in which we have increased the demand for the first class. As we can see, the probability for demanding* 12 *seats in class* 1 *is equal to the probability for demanding* 12 *seats in class* 2. *If we compare the protection levels from the first case, the ones we got here are higher. More specific, y_2 has been doubled, which is logical since the demand for class* 1 *has been increased, so obviously we want to sell more seats in class* 1.

4. *In case 8, we have reduce the spread of the demand and we have also decreased the demand for every class, so we have the same protection levels with the ones we had in the first case.*

5. *In the last case, we have spread even more the probabilities of demand, i.e. we have smaller probabilities in each row but for more different seat demands. As we can see the protection level for class* 1 *is really small, which is expected since the probability of demanding many seats in class* 1 *is small, only* 10%.

2.5.2 Monte Carlo integration

The second approach to computing optimal protection levels is based on using

$$Pr[A_j(y^*, D)] = \frac{p_{j+1}}{p_1},\ j = 1, 2, ..., n-1,$$

together with Monte Carlo integration. It is most naturally suited to the case of continuous demand and capacity, though the discrete case can be computed (at least heuristically) with this method as well.

The idea is to simulate a large number K of demand vectors $D = (D_1, ..., D_n)$.

One then progressively sorts through these values to find thresholds $y_1, y_2, ..., y_{n-1}$ that approximately satisfy the $Pr[A_j(y^*, D)]$.

In what follows, it is convenient to note that

$$Pr[A_j(y, D)] = Pr[D_1 + .. + D_j > y_j | A_{j-1}(y, D)] Pr[A_{j-1}(y, D)]$$

Thus, the optimal y^* must satisfy

$$Pr[D_1 + .. + D_j > y_j^* | A_{j-1}(y^*, D)] = \frac{p_{j+1}}{p_j},\ j = 1, 2, .., n-1.$$

Indeed, we have

$$Pr[D_1 + .. + D_j > y_j^* | A_{j-1}(y^*, D)] = \frac{1}{Pr[A_{j-1}(y^*, D)]} \frac{p_{j+1}}{p_1} = \frac{1}{\frac{p_j}{p_1}} \frac{p_{j+1}}{p_1} = \frac{p_{j+1}}{p_j}, j = 1, ..., n-1.$$

The algorithm in Figure 2.4 computes the optimal y^* approximately using the empirical conditional probabilities derived from a sample of simulated demand data.

The complexity of this method is $O(nKlogK)$, which is nearly linear in the number of simulated demand vectors K. Thus, it is relatively efficient even with large samples. It is also quite simple to program and can be used with any general distribution.

2.6 Heuristic Methods

As we have seen, computing optimal controls for the static single resource model is not mathematically difficult, but they can be computationally intensive when need to be applied to thousands of flights in a limited period of time. For this reason, exact optimization models are not widely used in practice. Indeed, most single resource airline revenue management systems use one of several heuristics to compute booking limits and protection levels.

There are two main reasons for this state of affairs. The first is simply a case of practice being one step ahead of the underlying theory. Heuristics are also widely used because they are simpler to code, quicker to run, and generate revenues that in many cases are close to optimal.

The most popular of these heuristics are the expected marginal seat revenue (EMSR) approaches, especially EMSR-a and EMSR-b. Both heuristics are based on the n-class, static, single-resource model defined above. Static model assumptions apply: classes are indexed so that $p_1 > ... > p_n$, $F_j(x)$ denotes the c.d.f. of class j

```
function [y] = montecarlo(D,r,K)
%The demand for each class is Normally distributed
%inputs
%D: is a nx2 matrix
%in the first column it has the mean for each class
%and in the second one it has the standard deviation for each class
%r: is a nx1 vector with the revenues for each class
%K: the number of the random demand vectors
%outputs
%mc: 1x(n-1) vector with the protection levels
[n]=size(D,1);
for i=1:n
  for j=1:K
    Dr(i,j)=D(i,1)+D(i,2)*randn;
    %random demand vector with m=D(i,1) and std=D(i,2)
  endfor
endfor
S=zeros(n,K);
for k=1:K
  S(1,k)=Dr(1,k);
  for j=2:(n-1)
    S(j,k)=S(j-1,k)+Dr(j,k);
  endfor
endfor
T=[1:K];
for j=1:(n-1)
  Sp=S(j,:);
  for k=T
    temp=Sp(1,k);
    m=find(Sp==min(Sp(1,k:max(T))));
    Sp(1,k)=min(Sp(1,k:max(T)));
    Sp(1,m)=temp;
  endfor
  l=length(T)-max(floor((r(j+1)/r(j))*length(T)),1);
  y(j)=0.5*(Sp(1,l)+Sp(1,l+1));
  i=1;
  for k=T
    if S(j,k)>y(j)
      Stemp(:,i)=S(:,k);
      i=i+1;
    endif
  endfor
  S=Stemp;
  Stemp=[];
  T=[1:size(S,2)];
endfor
y
endfunction
```

Figure 2.4: Monte-Carlo Algorithm

demand, and low revenue demand arrives before high-revenue demand in stages that are indexed by j as well. Moreover, for ease of exposition we assume that capacity and demand are continuous and that the distribution functions $F_j(x)$, $j = 1, .., n$, are continuous as well, though these assumptions are easily relaxed.

2.6.1 EMSR-a

EMSR-a is based on the idea of adding the protection levels produced by applying Littlewood's rule to successive pairs of classes. Consider stage $j+1$ in which demand of class $j+1$ arrives with price p_{j+1}. We are interested in computing how much capacity to reserve for the remaining classes $j, j-1, .., 1$ that is the protection level y_j, for classes j and higher. To do so, suppose there was only one class remaining, call it k. Considering only classes k, for each $k = j, j-1, ..., 1$, and $j+1$, solving $Pr[D_k \geq y_k^{j+1}] \leq \dfrac{p_{j+1}}{p_k}$ we reserve capacity y_k^{j+1} for class k. Now we have computed protection level for each class k in isolation. To approximate the total protection level y_j for classes j and higher we have $y_j = \sum_{k=1}^{j} y_k^{j+1}$. Then we repeat the same calculation for each stage j.

EMSR-a can be excessively conservative and produce protection levels y_k^{j+1} that are larger than optimal in certain cases. This is because adding the individual protection levels ignores the statistical averaging effect (pooling effect) produced by aggregating demand across classes.

```
function [Y] = emsra(m,s,p,C)
  %Computes a vector of optimal protection levels Y
  %the j+1 component of Y is protection level yj
  %inputs
  %m: nx1 matrix that gives the mean demand for each class
  %s: nx1 matrix that gives the variance of the demand for each class
  %p: nx1 matrix that gives the fees for each class
  %C: total capacity
  n=length(p);
  Y=zeros(1,n);
  y=zeros(n,n);
  for j=1:n
    for i=1:(j-1)
      y(j,i)=littlewoodnorm(m(i),s(i),p(j),p(i),C);
      Y(j)=Y(j)+y(j,i);
    endfor
   endfor
   Y
endfunction
```

Figure 2.5: EMSR-a Algorithm

2.6.2 EMSR-b

EMSR-b again reduces the problem at each stage to two classes, but now the approximation is based on aggregating demand, no protection levels like in EMSR-a. Specifically, the demand from future classes is aggregated and treated as one class with a revenue equal to the weighted average revenue.

This heuristic assumes that the demand in all classes follows normal distribution. Specifically, the demand for class j is $D_j \sim Normal(\mu_j, \delta_j^2)$. It is based on the idea of finding the protection level y_j by creating an "artificial class" with demand equal to the sum of the demands for the future periods, i.e., $D^{(j)} \sim Normal(\mu^{(j)}, (\delta^{(j)})^2)$ and a fare equal to the average expected fare from the future bookings, i.e., $p^{(j)} = \sum_{i=1}^{j} \frac{p_i \mu_i}{\mu^{(j)}}$, where $\mu^{(j)} = \sum_{i=1}^{j} \mu_i$ and $\delta^{(j)} = \sqrt{\sum_{i=1}^{j} \delta_i^2}$. Then, it uses Littlewood's rule (2.1) to find the protection level for this artificial class. So, y_j is given as $y_j = \mu^{(j)} + \delta^{(j)} \Phi(1 - \frac{p_{j+1}}{p^{(j)}})$.

EMSR-b avoids the lack-of-pooling defect in EMSR-a mentioned above. However, using the weighted average revenue is a somewhat crude approximation that can distort the protection levels produced.

```
function [Y] = emsrb(m,s,p,C)
  %Computes a vector of optimal protection levels
  %the j+1 component of Y is protection level yj
  %inputs
  %m: nx1 matrix of means
  %s: nx1 matrix of variations
  %p: nx1 matrix that gives the fees for each class
  %C: total capacity
  n=length(p);
  mu=zeros(1,n);
  su=zeros(1,n);
  pu=zeros(1,n);
  y=zeros(1,n);
  for j=2:n
    for i=1:(j-1)
     mu(j)=mu(j)+m(i);
     su(j)=su(j)+s(i)^2;
     pu(j)=pu(j)+p(i)*m(i);
    endfor
    pu(j)=pu(j)/mu(j);
    su(j)=su(j)^(1/2);
    y(j)=littlewoodnorm(mu(j),su(j),p(j),pu(j),C);
  endfor
  y
endfunction
```

Figure 2.6: EMSR-b Algorithm

2.6.3 Comparison EMSR and Monte Carlo

EMSR-a and EMSR-b generally capture a high percentage of optimal control. There are cases where EMSR-a approximates the optimal protection level better than EMSR-b. Also, there are cases where EMSR-b approaches the optimal protection level better. Below we see one example of its case.

Example 5. *We assume that there are four classes with fares* \$1150, \$965, \$750 *and* \$530 *and the capacity is* 120. *The demand for every class is normally distributed. Class* 1 *has mean* 15 *and standard deviation* 6, *class* 2 *has mean* 45 *and standard deviation* 12, *class* 3 *has mean* 37 *and standard deviation* 9 *and class* 4 *has mean* 29 *and standard deviation* 15.

In Monte Carlo algorithm for 1000 simulated data points we have $y = \begin{bmatrix} 9.1530 \\ 52.8609 \\ 96.9874 \\ 120 \end{bmatrix}$,

while in EMSR-a and EMSR-b we get $y = \begin{bmatrix} 9.05466 \\ 48.49949 \\ 91.21203 \\ 120 \end{bmatrix}$ *and* $y = \begin{bmatrix} 9.05466 \\ 51.29999 \\ 93.68057 \\ 120 \end{bmatrix}$, *respectively.*

Example 6. *As in the above example we assume four classes with fares* \$1150, \$465, \$450, *and* \$430. *We keep the same capacity and demand distribution for every class. Again we simulate* 1000 *data points in Monte Carlo algorithm and we get* $y = \begin{bmatrix} 16.722 \\ 41.874 \\ 79.787 \\ 120 \end{bmatrix}$, *while in EMSR-a and EMSR-b we get* $y = \begin{bmatrix} 16.45265 \\ 39.47237 \\ 66.36583 \\ 120 \end{bmatrix}$ *and* $y = \begin{bmatrix} 16.45265 \\ 52.68236 \\ 85.54854 \\ 120 \end{bmatrix}$, *respectively.*

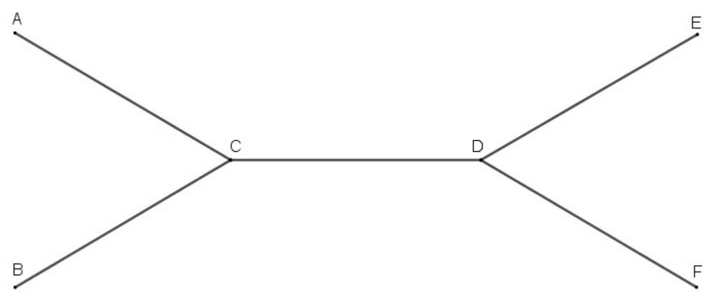
Chapter 3

Network Capacity Control

3.1 Introduction

Network Revenue Management is applied when a company sells products that use multiple resources and each resource has limited capacity. We first give two examples where an airline company and a hotel need to apply Network Revenue Management.

Example 7. *Suppose we have to manage the capacities of a set of flights in a hub-and-spoke airline network with connecting and local traffic. At the following figure we can see the connection among the flights:*



We assume that we can move only from the left to the right. Man can see that

- *from town A and B we can go to town C, D, E, and F,*
- *from town C we can go to town D, E, and F, and*
- *from town D we can go to town E and F.*

Thus, we have 13 *flights totally. Moreover, we let every flight have only full-fare and discount-fare. The combination of our flights and fees is our* ***products****, so here we have* 26 *products. Every flight, without a stop, is a* ***resource****, i.e., here we have* 5 *resources* $A \to C$, $B \to C$, $C \to D$, $D \to E$ *and* $D \to F$. *We should allocate the capacity of each resource to the products that use this resource. For example, the capacity of resource* $B \to C$ *should be allocated to the following* 8 *products:* $B \to C$, $B \to D$, $B \to E$ *and* $B \to F$ *full and discount-fare. Then in order to accept a demand for a particular product, we should check the booking limits of every resource that this product uses.*

Example 8. *Assume now we have to manage hotel capacity on consecutive days where customers have varying lengths of stay. Here, resources are each day and products are the combination of the days, i.e., the arrival date, and the length of stay. For example, a hotel accepting reservations for customers arriving for the next 365 days with stays from 1 to 10 days in length and only one room type offers*

$$1 \times 10 \times 365 = 3,650 \; products.$$

We have to deal with the following problem: if a demand from Monday till Friday comes should we accept it or not? A reservation from Thursday till Monday will be much more expensive than the aforementioned, due to it includes weekend. So, if we accept the first one maybe we will lose money.

Again, to solve this problem we should allocate the capacity of each resource to the products that use this resource.

From the above examples it is obvious that our challenge is to allocate capacity from every resource to every product. Then when demand of a product arrives we should check the remaining capacity of each resourse it uses. In the example above we see a very simple airline network, however, someone can notice that for even less resources it is very difficult to decide which resource to sell to each product. Imagine having even more fees, not only the full and the discount ones, as well as even more towns, which is actually what happens in practice.

Due to the above, the need to develop Network Capacity Control arises. Here the demand of a resource isn't independent of the demand of the other resources, because of the fact that customers may demand more than one resource concurrently to accommodate their needs, so limiting availability of one resource may cause a loss of demand for complementary resources. Thus, we can no longer maximize total revenue by maximizing revenue from each resource independently, as we did in single resource problems. Rather, we need to consider the interactions among the various products we sell and their effect on our ability to sell other products.

Making control decisions at network level can provide significant revenue benefits. Simulation studies have shown that improvements start from 0.5% and they can be as high as 2% or more. The potential benefit may be high, however, network Revenue Management has some drawbacks as well. Firstly, on the implementation side, network Revenue Management increases the complexity and variety of data that one must collect, store, and manage. Secondly, there are organizational challenges, since revenue decisions and their effects now span an entire network and revenue losses at one point in the network may be offset by gains elsewhere in the network. Next, one will face methodological challenges as well. The forecasting system now must produce forecasts for each individual itinerary and price-class combination, i.e. for each product, at each point in the booking process. Lastly, optimization is more complex too. As long as, exact optimization is, for all practical purposes, impossible optimization methods necessarily require approximations of various types.

3.2 Types of controls

In network allocation problems there are a variety of ways one can control the availability of capacity. The main purpose is to find an effective and easy to apply method to allocate capacity. We next look at the major categories of network controls.

3.2.1 Partitioned Booking Limits

In the network case, partitioned booking limits allocate a fixed amount of capacity on each resource for every product that is offered. Demand for a product has exclusive access to its allocated capacity, and no other product may use this capacity.

Partitioned booking-limit control for network is an extension of the partitioned control for single-resource, thus they have all the defects mentioned in single resource section and some more. First of all, the number of products in even a small-size network, like in our example, can be very large. Therefore, allocating fixed amounts of capacity to each product results in dividing the capacity of each resource into a very large number of small allocations. As a result maybe there is a severe loss, since many products may go unsold because they use up their meagre allocations if demand is even slightly higher than the allocation, while many others may have excess capacity if demand is slightly less than their allocation. Apart from inefficiencies, the tremendous number of product combinations makes the storing and checking allocations for each combination impractical.

Example 9. *For the airline example above, suppose that at each aeroplane we have* 100 *seats. As we said above we have* 14 *products. Let* $A \to C$ *be the aeroplane goes from town* A *to* C*,* $B \to C$ *the one goes from* B *to* C*, etc., namely the resources. Resource* $A \to C$ *can be used at* 8 *products, resource* $B \to C$ *at* 8 *as well, resource* $C \to D$ *at* 10*, resource* $D \to E$ *and* $D \to F$ *at* 8*. Thus, in partitioned booking limits many of these products will have only a few seats.*

For all the reasons above, partitioned booking limits are seldom used in practice. Nevertheless, partitioned allocation do have an important role to play both theoretically and computationally. Theoretically, they are used to provide bounds on the optimal network revenue. Computationally, they are used in many approximate models.

3.2.2 Greedy Heuristic Method

Here we will introduce a heuristic method called Greedy Heuristic that can be used when only one resource has capacity that is less than the total expected demand. A Greedy Methodology is a procedure were we take under consideration only the resource that its demand exceeds its capacity and solve this problem by using EMSR algorithms from the previous chapter.

In general, many mathematical methods of operation research may fail to solve a problem, thus we use greedy methods which can provide high quality solution very fast and are applicable to huge problems. But the disadvantage of these algorithms is that they don't always produce the optimal solution just a very good one.

Example 10. *Suppose we have an airline that offers two flights. Flight* 1 *goes from town* A *to* B *and flight* 2 *from town* B *to* C*. The airline has assigned a* 100*-seat aircraft on flight* 1 *and a* 300*-seat aircraft on flight* 2*. Moreover, suppose we have* *only full and discount fares. The demand of each product is normally distributed. The fare of each product, the mean and standard deviation of its demand are given in the following table:*

Product	Fare	Mean	SD
1. A-B full-fare	\$ 350	40	5
2. A-B discount	\$ 280	55	15
3. B-C full-fare	\$ 400	25	10
4. B-C discount	\$ 300	60	13
5. A-C full-fare	\$ 700	50	8
6. A-C discount	\$ 550	70	12

We notice that here, expected demand exceeds capacity only on flight 1. *All the demand of products that use only the other resource (i.e., products 3 and 4) will be accepted.*

One approach to maximize our revenues is to limit the number of discount fares booking any product that includes flight 1 in order to protect availability for full fare customers. This approach has an obvious drawback. Full fare customers that buy a ticket for flight $A \to B$ pay \$350, *but discount customers for flight $A \to C$ pay* \$550. *Namely, we have the risk of losing* \$200.

A more sophisticated approach, which considers both flight rate and combinations of flights, is called Greedy heuristic. Here we order all products that use flight 1 *by total rate paid and solve the multi-class problem using EMSR heuristic.*

We will order the products that use the constrained recourse by their fare and use EMSR heuristics to find the optimal allocation.

Fare class	Product	Fare	Mean	SD
1	5	\$ 700	50	8
2	6	\$ 550	70	12
3	1	\$ 350	40	5
4	2	\$ 280	55	15

By applying the above to EMSR algorithms from Chapter 2, we have:

Protection level	EMSR-a	EMSR-b
y_1	43.66689	43.66689
y_2	115.81493	117.40382
y_3	157.54520	159.54079

Thus, according to EMSR-A algorithm we should protect 43.66 *seats for flight* $A \to C$ *full-fare,* 115.81 *seats for flights* $A \to C$ *full-fare and discount and* 157.54 *seats for flights* $A \to C$ *full-fare and discount and for flight* $A \to B$ *full-fare. The remaining seats will go to the flight* $A \to B$ *discount. Accordingly for the EMSR-B algorithm. Since, our capacity is* 100, *it is obvious that* 43.66 *seats will be protected for flight* $A \to C$ *full-fare, and 100 for flight* $A \to C$ *full-fare and all the remaining.*

Here, we should note that Greedy heuristic is optimal if there is only a single resource that is expected to be constrained and breaks down quickly for more than one constrained resources.

3.2.3 Linear Programming

Linear programming provides an exact solution to the network management problem, if we know the exact future demands and they are independent. On the one hand, we know that uncertainty is a key element of revenue management decision making, so this solution is not entirely sufficient. On the other hand, it provides a good starting point for a fully optimal solution.

Suppose we have m resources and n products, which are using combinations of these resources, with $n \geq m$. We represent with i the resources for $i = 1, 2, .., m$ and with j the products for $j = 1, 2, .., n$. Moreover, the resource i has capacity c_i and the product j has a known demand d_j and price p_j. We define:

$$a_{ij} = \begin{cases} 1, \, if \; resource \; i \; is \; used \; by \; product \; j \\ 0, \, otherwise. \end{cases}$$

Lastly, we have the decision variable x_j, which is the number of product j units that we will sell. So, now we have to solve the following linear program:

$$\max \sum_{j=1}^{n} p_j x_j$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \leq c_i, \; i = 1, 2, .., m,$$

$$x_j \leq d_j, \; j = 1, 2, .., n,$$

$$x_j \geq 0, \; j = 1, 2, .., n.$$

3.2.4 Virtual nesting controls

In network case the ordering of classes is no straightforward and because of the different capacities of the resources it is difficult to specify protection levels or booking limits for products that are consistent across the resources in the network. Thus, it's difficult to adjust booking limits and protection levels from the single-resource case.

Contrary to partitioned controls, nested controls have the advantage to dynamically share the capacity of a resource, namely it allows products to be nested. Moreover, companies want a control, which allows them to adapt the pre-existing, leg-based control structures in its reservation system to the network management problem with minimal changes. Hence, it is desirable to have a control with these features, so American Airlines developed virtual nesting controls.

Virtual nesting uses single-resource nested-allocation controls for each resource in the network. But here, we use **virtual classes** in nested allocation, which group together sets of products that use a given resource. Products are assigned to a virtual class through a process known as **indexing**, which fundamentally provides a table that maps every product to a virtual class on each resource. We don't use product's total fares for the indexing process, because a product with a high fare but a high opportunity cost on its other resources should potentially be placed below a product with lower fare but no opportunity cost on its other resources. For this reason, the indexing process associates a network revenue benefit called **net leg fare** to each product defined as:

$$net\,leg\,fare\,for\,product\,j\,on\,resource\,i = \begin{array}{c} fare\,for\,product\,j \\ - \\ sum\,of\,opportunity\,costs\,on\,all\,resources\,other\,than\,i \end{array}.$$

The 3 parts of virtual nesting are initialization, operation and re-optimization. We describe each one below.

I) Initialization: The initialization of virtual nesting consists of 3 steps.

Step 1

We define a set of buckets on each resource and we map each product into a bucket on each of its resources based on an estimation of the product's value, i.e., the net leg fare, to the company. The number of buckets is based on the number of products that use the resource, but there is no strict rule on it. Each bucket should contain products of similar value.

Step 2

For each resource bucket, we estimate the mean and the standard deviation of the total demand and a corresponding fare.

Step 3
We use EMSR to determine booking limits and protection levels for each bucket on each resource.

Comment 3. *We have the following characteristics for buckets:*

1. *Each bucket contains products of similar value.*

2. *The bucket* 1 *on each resource contains the most expensive products and the last bucket the cheapest products.*

3. *The buckets are nested, namely bucket* 1 *has access to the entire capacity, bucket 2 has access to all the capacity except that protected for bucket 1, and the lowest bucket has access only to its own allocation.*

II) Operation: Once initialization is done the booking process starts. When a request comes for a particular product combination, its virtual class is identified and the system checks for availability of this virtual class on each resource required by the product.

- If all the virtual classes are available, the request is accepted and we reduce bucket availabilities on all resources in the booked product, else, the request is rejected.

- If a cancellation occurs we increase bucket availabilities for each resource in the cancelled product.

III) Re-optimization: We periodically rerun EMSR based on the new forecasts and capacity remaining on each resource.

Virtual nesting has proven to be quite effective and popular in practice, on account of its ability to preserve the single-resource, nested allocation structure of control. However, it has a few remarkable disadvantages. First of all, if data is collected at the virtual class level, then re-indexing can alter which products are mapped into each virtual class. So, due to indexing the virtual class demand statistics may shift dramatically even when the underlying product-level demand is unchanged. Virtual classes can also cause confusion for analysts, who may not be able to easily interpret virtual class demand.

Example 11. *As in the above example in Greedy Heuristic section, suppose we have an airline that offers two flights. Flight* 1 *goes from town* A *to* B *and flight* 2 *from town* B *to* C. *The airline has assigned a* 100*-seat aircraft on flight* 1 *and a* 250*-seat aircraft on flight* 2. *Moreover, suppose we have only full and discount fares. The demand of each product is normally distributed. The fare of each product, the mean and standard deviation of its demand are given in the following table:*

Product	Fare	Mean	SD
1. A-B full-fare	\$ 350	40	5
2. A-B discount	\$ 280	55	15
3. B-C full-fare	\$ 400	75	10
4. B-C discount	\$ 300	80	13
5. A-C full-fare	\$ 700	65	8
6. A-C discount	\$ 550	85	12

The opportunity cost for flight 1 *is* \$150 *and for flight* 2 *is* \$300. *As we can see here, the expected demand exceeds capacity on both flights, thus we can't use Greedy Heuristic. So, we should do the initialization of virtual nesting:*

I) Initialization:

Step 1.*Indexing Here we have decided to use 3 buckets for each flight.*

		Flight 1		Flight 2	
Product	Fare	Net Leg Fare	Bucket	Net Leg Fare	Bucket
1.	\$350	\$350	2	-	
2.	\$280	\$280	3	-	
3.	\$400	-		\$400	2
4.	\$300	-		\$300	3
5.	\$700	\$400	1	\$550	1
6.	\$550	\$250	3	\$400	2

Step 2.*For each leg bucket, estimate the mean and the standard deviation of the total forecasted demand and a corresponding fare.*

Flight 1				
Bucket	Products	Mean	SD	Fare
1	5	65	8	\$700
2	1	40	5	\$350
3	2,6	140	19.209	\$261.7857

For bucket 3 *we have:*

$$mean = 55 + 85 = 140$$

$$SD = \sqrt{15^2 + 12^2} \simeq 19.209$$

$$fare = \frac{55 * 280 + 85 * 250}{140} \simeq 261.7857$$

Flight 2				
Bucket	Products	Mean	SD	Fare
1	5	65	8	$700
2	3,6	160	15.62	$400
3	4	80	13	$300

For bucket 2 we have:

$$mean = 75 + 85 = 160$$

$$SD = \sqrt{10^2 + 12^2} \simeq 15.62$$

$$fare = \frac{400 * 75 + 400 * 85}{160} = 400$$

Step 3. *Using EMSR-a and EMSR-b, for the mean and the standard deviation of demand by bucket and flight capacity, we get the following protection levels for each bucket on each flight:*

Protection levels	Flight 1		Flight 2	
	EMSR-a	EMSR-b	EMSR-a	EMSR-b
y_1	65	65	63.55	63.55
y_2	104.23	105.9	215.9	219.8

II) Operation:

The operation starts. Suppose we have the following booking request and cancellations:

1. *Request 2 tickets for A-B full-fare*

2. *Request 6 tickets for A-C discount*

3. *Request 6 tickets for A-C full-fare*

4. *Cancellation 2 tickets for A-C full-fare*

5. *Request 3 tickets for B-C full-fare*

Firstly, we should calculate the nested booking limits:

Flight 1

$$b_1 = C = 100$$

$$b_2 = C - 65 = 35$$

$$b_3 = C - 104.23 = C - 100 = 0$$

Flight 2

$$b_1 = C = 250$$

$$b_2 = C - 65 = 185$$

$$b_3 = C - 215.9 = 34.1$$

A request for 2 tickets for the product "A-B full-fare" arrives. We see that this product belongs to bucket 2 at flight 1. Since, we have the requested demand ($b_2 = 35$) we accept the request and we reduce each booking limit at flight 1 by 2.
Then, a request for 6 tickets for the product "A-C discount" arrives. This product belongs to bucket 3 at flight 1, which, as we can see, has booking limit $b_3 = -2$, thus we should reject the request.
Continuing, we receive another request for 6 tickets for the product "A-C full-fare", which belongs at bucket 1 at flight 1 and 2. In order to accept this request we should have available capacity in both flights, so we check the booking limit for bucket 1 in both flights, which is 98 and 250, respectively. Thus, we accept the request and reduce all booking limits by 6.
Now, a cancellation for 2 tickets at product "A-C full-fare" arrives. Thus, we increase all booking limits by 2, since this product belongs in both flights.
Finally, a request for 3 tickets for the product "B-C full-fare" arrives, which belongs at bucket 2 at flight 2. We accept it ($b_2 = 182$) and reduce all booking limits by 3.

Requests/Cancellations	Accept/Reject	Flight 1			Flight 2		
		b_1	b_2	b_3	b_1	b_2	b_3
		100	35	0	250	185	34.1
2 tickets for A-B full-fare	A	98	33	-2	250	185	34.1
6 tickets for A-C discount	R	98	33	-2	250	185	34.1
6 tickets for A-C full-fare	A	93	27	-8	245	180	29.1
2 tickets for A-C full-fare	-	95	29	-6	247	182	31.1
3 tickets for B-C full-fare	A	95	29	-6	244	179	28.1

3.2.5 Bid Price controls

Contrary to nesting controls, single resource bid-price controls are easy to translate directly into a network setting. Here, in network case, a bid-price control sets a threshold price (bid price) for each resource in the network defined as:

$$\text{bid price} = \begin{array}{c}\text{estimate of the marginal cost to the network of consuming} \\ \text{the next incremental unit of the resource's capacity}\end{array}$$

When a request for a product comes in, if the revenue of the request exceeds the sum of the bid prices, the request is accepted else, it is rejected.

One can notice that the bid-price controls get over a lot of the problems we had in virtual nesting. Namely, first of all the number of parameters involved is minimal, since one only has to specify a single value for each resource and not for each product as we had in virtual nesting. Furthermore, they have a natural economic interpretation as the marginal cost to the network of each resource, so business analysts can easily understand and manage network revenue management systems. Finally, as we saw above, evaluating each request only requires a simple comparison of revenue to the sum of bid prices for the requested resources, apart from the fact that the transaction processing task is quick, this comparison provides a measure of the net benefit of accepting a request.

Bid prices can have very good revenue performance in simulation studies and can even be shown to be theoretically near-optimal under certain conditions. Despite these advantages, bid-price controls had gained some bad criticism. The debate about bid prices being "unsafe" was discussed in Chapter 2 in the single-resource context, initially one deals with the same issues in network case. Namely, in bid-price control if there is a rush of discount demand, the entire inventory may be sold out; this is not the case regarding the nesting controls because of their inherent limits on discount availability. An other criticism, concerns the overall revenue performance of bid-price methods. Indeed, bid-prices can perform badly, worse than virtual nesting and sometimes worse than even single-resource control methods.

The bid-price disadvantages described above affect only the most simple implementation of bid-price controls. Bid-prices and capacity are dependent, so bid-prices should be updated with every change in capacity, every sale or cancellation, and ideally they should be updated over time as well. One way to implement these updates is through dynamic bid-price controls, i.e. capacity-time dependent tables.

3.3 Dynamic programming approach for optimal network controls

We will describe the basic model of the network allocation problem. The network has m resources which can be used to supply n products. We define

$$a_{ij} = \begin{cases} 1, & \text{if resource } i \text{ is used by product } j, \text{ and} \\ 0, & \text{otherwise.} \end{cases}$$

Moreover, let $A = [a_{ij}]$ be the $m \times n$ incidence matrix. Thus, A_j, the j^{th} column of A, is the incidence vector for product j, namely has an entry of one in every resource that is used by product j. Equally, A^i, the i^{th} row, denote the set of products that use resource i. In other words, the notation $i \in A_j$ indicates that resource i is used by product j, and likewise $j \in A^i$ that the product j uses resource i.

We define the vector $x = (x_1, x_2, ..., x_m)$, where x_i is the capacity for the resource i. This vector describes the state of the network. If product j is sold, the state of the network changes to $x - A_j$ (i.e. $x_i - a_{ij}$), namely we abstract from the capacity of the resource i one if product j uses resource i. To simplify our analysis, we will ignore cancellations or no-shows and also we will omit the possibility of booking requests for multiple units of capacity.

We discretize the time, i.e., we divide the time horizon into small slots such that within each time slot at most one request for a product can arrive. We assume that there are T periods and index t represents period (slot) t. The time of service is $t = T$.

Demand in period t is modelled as the realization of a single random vector $R(t) = (R_1(t), ..., R_n(t))$.

$$R_j(t) = \begin{cases} r_j, & \text{if a request for product } j \text{ occurred, and} \\ 0, & \text{if no request for product } j \text{ occurred,} \end{cases}$$

where $R_j(t) > 0$ is the associated revenue. A realization $R(t) = \underline{0}$ (all components equal to zero) indicates that no request from any product occurred at time t. Since the probability of more than one request is negligible, at most one component of $R(t)$ can be positive.

Once again we deal with the quantity-based revenue management decision:

"Do we or do we not accept the current request?"

We define the n-vector $u(t)$, which denotes this decision. The component $u_j(t) = u_j(t, x, r_j)$ depends on the current time t, the current remaining capacity x and the current request's price r_j. Specifically,

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

Since we can accept at most one request in any period and resources cannot be oversold, if the current seat inventory is x, then $u(t)$ is restricted to the set $\mathcal{U}(x) = \{u \in E_n : Au \leq x\}$, where $E_n = \{e_0, .., e_n\}$ and e_j is the j^{th} unit n-vector.

3.3.1 The structure of the optimal controls

In order to formulate a dynamic program to determine optimal decisions $u^*(t, x, r)$, we have the following theorem:

Theorem 4. ***(Bellman equation)*** *Let $V_t(x)$ denote the maximum expected revenue-to-go given remaining capacity x in period t. Then $V_t(x)$ must satisfy the Bellman equation:*

$$V_t(x) = E\left[\max_{u \in \mathcal{U}(x)} \left\{R^T(t)u(t, x, R(t)) + V_{t-1}(x - Au)\right\}\right],$$

with the boundary condition $V_{T+1}(x) = 0$, $\forall x$.

Comment 4. *One can notice that $V_t(x)$ is finite for every finite x.*

Corollary 3. *An optimal control u^* satisfies:*

$$u_j^*(t, x, r_j) = \begin{cases} 1, & r_j \geqslant V_{t+1}(x) - V_{t+1}(x - A_j) \text{ and } A_j \leqslant x \\ 0, & \textit{otherwise.} \end{cases}$$

Namely, an optimal policy is to accept revenue r_j for product j if and only if we have sufficient remaining capacity and $r_j \geqslant V_{t+1}(x) - V_{t+1}(x - A_j)$, where $R_j(t) = r_j$ is the revenue value of the request for product j.

Comment 5. *Intuitively, an optimal control is to accept a revenue of* r_j *for a given product only when it exceeds the opportunity cost of the reduction in resource capacities required to satisfy the request, which is reflected exactly in the above equation.*

Chapter 4

Overbooking

4.1 Introduction

In the aspect that we saw Revenue Management in the previous chapters, it was mainly concerned with how best to price tickets or allocate capacity. However, overbooking is concerned how we can increase capacity utilization. Overbooking occurs when an airline with limited capacity sells more units than it can offer. The reason that airlines book more seats than they actually have, is obviously to avoid revenue lost due to no-showing passengers or cancellations. It has been shown that, in financial terms, overbooking is among the most successful of Revenue Management practices. Indeed, without a form of overbooking a big percentage of the seats from cancellations or no-shows will remain unsold.

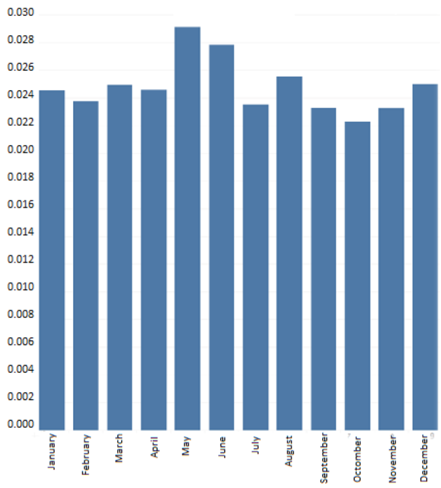


Figure 4.1: Monthly rate of no-shown passengers
https://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=8903812&fileOId=8903813

Consider a flight with 100 seats that has a higher demand than 100. Also, assume that passengers on this flight on average have a 10% no-show or cancellation rate. If the airline never overbooks, it would, on average, depart with 10 empty seats on every flight and at the same time deny reservations to passengers who wanted to be on this flight. If a cancellation occurs last minute and the passengers have refundable ticket this will cause to the airline the lost of the tickets' fee and also it will be very hard to find another passenger. On the other hand, a no-show will not cost to the airline, but they maybe had denied selling tickets to other passengers, which causes bad reputation for the airline.

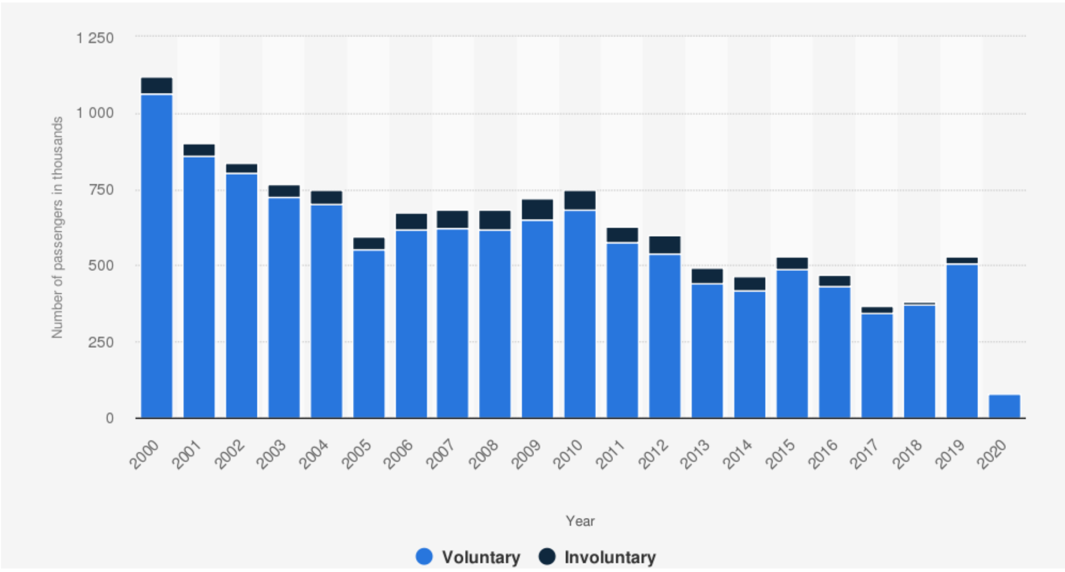


Figure 4.2: Number of passengers denied boarding from 2000-2020
https://www.statista.com/statistics/186198/passengers-voluntary-and-involuntary-not-boarded-since-1990/

Overbooking has a lot of risks. The biggest challenge in overbooking is to control the level of reservations to balance the potential risks of denied service against the rewards of increased sales. Obviously, when overbooking too much, there is a penalty, which is usually more expensive for the airlines than to have an empty seat. Moreover, it involves dealing with the resulting legal and regulatory issues. Theoretically, this involves controlling parameters of a probability distribution, which introduces somewhat unique methodology that is not encountered in other areas of Revenue Management.

We use the following terminology:

No show is a booking that does not appear on the day of service.

Cancellation is a booking that is cancelled before the day of service.

Denied boarding is a booking that shows up on the day of service and does not receive service because of overbooking.

4.1.1 Reservation Mechanism

We can consider a fee reservation as a contract between a customer and the airline. When a customer books a fee, he directly has the claim to use the service he has paid for. Namely, he has the right to use the ticket in the date for which he made the reservation at a fixed price. Moreover, often the customer has also the option to cancel before the flight and many times they pay extras for this specific service. The cancellation option in a reservation gives customers the benefit of locking in availability in advance and many times in low prices, and also having the flexibility to cancel and refund if their plans or preferences change.

Most of the customers prefer reservations with a cancellation option, in order not to risk losing their money. But this type of reservations cause a two-sided risk for the airline. Firstly, in the case that customers show up they should honour the reservation or provide a suitable reimbursement. In the case where customers cancel or not show-up the airline should cope the opportunity cost of wasted capacity. They have two ways to counterbalance the economic losses. The first one is to impose penalties in the cases of cancellations in order to make this option less desirable. Obviously, if the penalties are too large this will make the cancellation choice pointless. The point of these penalties is to make the customers and the firms to share the risk of cancellations, otherwise customers could book many different fees and in the end keep the one they want.

Another approach is airlines to offer more seats than the capacity they really have. This of course has the risk the number of show up customers to be more than the capacity. In essence, this is the **planned overbooking**. Choosing the overbooking strategy makes the airline to face several important problems. One is confronting the legal implication of failing to deliver a service. Furthermore, they should have developed strategies in order to deal with the denied service situations, i.e., offering customers another flight soon in order to go to their destination and also offer them cheaper tickets. In many cases, where customers should stay in a hotel, the airline should pay for these expenses, too. So, a denied service implies costs to the company.

4.1.2 Legal issues in Overbooking

Since 1965, overbooking was an officially approved practice with the requirement of being "carefully controlled". This happens because of some studies, which have been conducted, shown that there is a no-show rate of 1 out of every 10 passengers booked contrary to the small denied boarding rate of 7.69 per $10,000$ passengers. No shows created real economic problems for the airlines. Through carefully controlled overbooking the airlines can reduce the number of empty seats and at the same time serve the public interest by accommodating more passenger.

Legally, overbooking involves the risk of potentially failing to deliver on a contract to provide service. Thus, for the airlines to be legally right, they should involve the following overbooking notification statement on all airline tickets:

> *"Airline flights may be overbooked, and there is a slight chance that a seat will not be available on a flight for which a person has a confirmed reservation. If the flight is overbooked, no one will be denied a seat until airline personnel first ask for volunteers willing to give up their reservation in exchange for a payment of the airline's choosing. If there are not enough volunteers the airline will deny boarding to other persons in accordance with its particular boarding priority. With few exceptions' persons denied boarding involuntarily are entitled to compensation. The complete rules for the payment of compensation and each airline's boarding priorities are available at all airport ticket counters and boarding locations. Some airlines do not apply these consumer protections to travel from some foreign countries, although other consumer protections may be available. Check with your airline or your travel agent."*

Moreover, there were the following regulations for overbooking:

1. Denied boarding compensation was of 200% of the coupon.
2. Airlines were required to seek volunteers first before denying boarding to any passenger involuntarily.
3. The travelling public was to be notified of the deliberate overbooking practices of the airlines.
4. A statement warning passengers that their flight may be overbooked and informing them of their rights was to be printed on every ticket.

These basic rules are still in existence since 1974, where a passenger was denied boarding, who sued the airline and won with the claim that the airline did not inform passengers of its practice of overbooking. Thereafter, airlines in order to find volunteers to give up their seats on oversold flights they use vouchers and payments. As a result, involuntary denied boardings have been decreased nowadays comparing to the time when overbooking was a secret practice.

4.1.3 Managing over-sales occurrences

In the event of over-sales they should choose the customers carefully, since managing the compensations and the selection of customers can have a significant impact on the expenses for the airline when they denied a service. Also, it affects the customers' perception for the airline.

As for compensation, in the case of denied service legally the company should give the customer a monetary compensation. But this compensation does not attract customers in order they to give up their seat. It has been proved that it is more effective to offer customers a substitute service plus supplementary services that may make the short-run delay in their schedule more acceptable. Compensation that is targeted to substitute for the denied service and perhaps improve it somewhat in most cases is less expensive for a provider and it seems to be more desirable from the customers than monetary compensation.

Selecting which customers are to be denied service also can have a significant impact on both the firm's direct costs and customer's goodwill, i.e., they should target on customers that they could delay their plans, namely not business travellers. From a legal standpoint, airlines current regulations state that:

> "Every carrier shall establish priority rules and criteria for determining which passengers holding confirmed reserved space shall be denied boarding on an oversold flight in the event that an insufficient number of volunteers come forward. Such rules and criteria shall not make, give, or cause any advantage to any particular person or subject any particular person to any unjust or unreasonable prejudice or disadvantage in any respect whatsoever."

The default option for allocating service to customers is usually to do it on a first-come, first-serve (FCFS) basis. A FCFS allocation is perceived as fair and encourages customers to arrive on time. But airlines do not always use this way of allocation.

According to some researches young, student travellers are eager to receive a nice hotel room and a good meal in exchange for taking a flight the following day. Therefore, gate agents see which passengers have arrived for the flight and selectively target specific passengers for denied-boarding offers, which is possible on the grounds that customers gather to the gate before departure.

An alternative method of managing over-sales is to conduct an auction to attract volunteers to give up their reservations in exchange for monetary or other compensation. At 1968, when this idea appears for the first time, the airlines object to the scheme and none of them offers to experiment with it on even a single flight. The scheme continued to flounder until 1977 when an economist adopted it under the heading of a “volunteer” denied-boarding plan. The happiest result of the volunteer plan is that airlines now have a fair and efficient way to avoid denying seats to people who have a pressing need to make their flights as planned. Moreover, it enables airlines to reduce costs while maintaining customer goodwill and thereby protecting future revenue. This practice is now widespread in airlines and familiar to most travellers.

To sum up, overbooking practices takes time to establish, because consumers have to get used to and accept overbooking practices, and providers in turn have to learn how to develop strategies and operational practices that make overbooking as painless as possible for customers. Moreover, overbooking is a well-developed and refined practice, it remains a primary source of dissatisfaction for customers. Even at its best, overbooking is a somewhat awkward compromise between economic efficiency and service quality.

4.2 Static Overbooking Models

In this section, we will analyse the methodology for making overbooking decisions based on static overbooking models, which is the simplest and most widely used methodology. In practice, static models are used to compute overbooking limits, which are then used as inputs to capacity-allocation models. The models simply determine the maximum number of fees to offer at the current time given estimates of cancellation probabilities from the current time until the day of service. These static overbooking models are typically re-solved periodically to account for changes in the cancellation and no-show probabilities over time, resulting in overbooking limits that decline over time, since while we are approaching the date of the flight is less possible a cancellation to arise. The current overbooking limit gives the maximum number of reservations one will accept at any time.

There are two types of events that affect the overbooking decisions, *cancellations* and *no-shows*. While both result in a situation where a reservation does not “survive” to the time of service their difference is related to the time each event occurs. Essentially, a no-show is a cancellation, which happened on the time of service. The main difference, between cancellation and no-show, is that with a cancellation, the firm has an opportunity to possibly replace the cancelled reservation, since they have more available time to find another customer in contrast to a no-show. Moreover, many fees offer to the customer the capability to cancel and take his money back, so the firm will lose these money if they don't find replacement. Contrary, in a no-show there is no refund, so the seat will be empty but paid. Under a static model, all that matters is the probability that a reservation survives to the time of service.

4.2.1 The Binomial Model

The simplest static model is based on a binomial model of cancellations in which a no-show is treated simply as a cancellation that occurs at the day of service. We make the following assumptions:

1. Customers cancel independently of one another.
2. Each customer has the same probability of cancelling.
3. The cancellation probability depends only on the time remaining to service and is independent of the age of the reservation.

Let t denote the time remaining until service, C denote the physical capacity, u denote the number of reservations on hand and q the probability that a reservation currently on hand shows up at the time of service, so $1 - q$ is the probability that a customer cancels prior to the time of service. In general the more time remaining the more likely it is that customers cancel before the time of service, so q decreases in the remaining time t.

Under the assumptions stated above and suppressing the notation regarding the remaining time t, the number of customers who show up at the time of service given there are u reservations on hand, denoted $Z(u)$ "the show demand", is binomially distributed with p.m.f.

$$p_u(z) = P_r[Z(u) = z] = \begin{pmatrix} u \\ z \end{pmatrix} q^z (1-q)^{u-z},\ z = 0, 1, .., u,$$

c.d.f.

$$F_u(z) = P_r[Z(u) \le z] = \sum_{k=0}^{z} \begin{pmatrix} u \\ k \end{pmatrix} q^k (1-q)^{u-k},\ z = 0, 1, .., u,$$

and mean and variance

$$E[Z(u)] = qu \quad \& \quad Var[Z(u)] = uq(1-q).$$

It is convenient to work with the complement of the distribution F_u, denoted by $\overline{F}_u$, which is defined by

$$\overline{F}_u(z) = 1 - F_u(z) = P_r[Z(u) > z].$$

Several studies have showed that group-cancellation behaviour does invalidate the binomial model for certain cabins on certain flights, overall its concluded that the binomial model adequately fits the data.

Overbooking Based on service level criteria

There are two types of service levels. Type 1 service level is the probability of over-sale at the time of service. Given that there are u reservations on hand, this probability is denoted $s_1(u)$ and is given by

$$s_1(u) = \overline{F}_u(C).$$

A matlab function that calculates Type 1 service level is shown in Figure 4.3.

```
function [s1] = s1(C,q,u)
  %s1 fuction computes the type 1 service level
  %for the binomial model
  %C is the capacity
  %q is the probability that a reservation currently on
  %hand shows up at the time of service
  %u is the number of reservations on hand
  s1=1-binocdf(C,u,q);
  %the second argument at binocdf is the number of trials
endfunction
```

Figure 4.3: Matlab function for Type 1 service level

A more intuitive measure of service is the Type 2 service level, which is the fraction of customers who are denied service and denoted by $s_2(u)$. This fraction is given by

$$\begin{aligned} s_2(u) &= \frac{E[(Z(u) - C)^+]}{E[Z(u)]} \\ &= \frac{\sum_{k=C+1}^{u}(k - C)p_u(k)}{uq} \\ &= \frac{\sum_{k=C+1}^{u} kp_u(k) - C\sum_{k=C+1}^{u} p_u(k)}{uq} \end{aligned}$$

$$
\begin{aligned}
&= \frac{\sum_{k=C+1}^{u} k \begin{pmatrix} u \\ k \end{pmatrix} q^k (1-q)^{u-k} - C\overline{F}_u(C)}{uq} \\
&= \frac{\sum_{k=C+1}^{u} u \begin{pmatrix} u-1 \\ k-1 \end{pmatrix} q q^{k-1} (1-q)^{(u-1)-(k-1)} - C\overline{F}_u(C)}{uq} \\
&= \frac{uq \sum_{k'=C}^{u-1} \begin{pmatrix} u-1 \\ k' \end{pmatrix} q^{k'} (1-q)^{(u-1)-k'} - C\overline{F}_u(C)}{uq} \\
&= \overline{F}_{u-1}(C-1) - \frac{C}{qu}\overline{F}_u(C).
\end{aligned}
$$

A matlab function that calculates Type 2 service level is shown in Figure 4.4.

```
function [s2] = s2(C,q,u)
  %s2 fuction computes the type 2 service level
  %for the binomial model
  %C is the capacity
  %q is the probability that a reservation currently on
  %hand shows up at the time of service
  %u is the number of reservations on hand
  s2=s1(C-1,q,u-1)-(C/(q*u))*s1(C,q,u);
endfunction
```

Figure 4.4: Matlab function for Type 2 service level

Setting an overbooking limit of u ensures that, at most, a fraction of $s_2(u)$ customer will be denied service. In practice, we must first specify the service level we want, namely the fraction of customers that will be denied the service, and then find the largest booking level u^* satisfying this standard. The resulting u^* is the **overbooking limit**. The quantity $u^* - C$ (the excess over capacity) is typically referred to as the **overbooking pad**.

Quantities $s_1(u^*)$ and $s_2(u^*)$ predict the service level in cases in which demand exceeds u^*, but service will be better if demand is strictly less than u^*. Thus, we are considering a worst-case service level, i.e. demand exceeding the overbooking limit. For this reason, we may result in low overbooking limits if average service levels are what matters.

The average service level can be easily determined given a distribution of reservation demand. Let the random variable D denote the number of reservations. Then by the renewal-reward theorem, the average Type 2 service given an overbooking level u

is given by

$$
\begin{aligned}
\overline{s}_2(u) &= \frac{E[\min\{D,u\}s_2(\min\{D,u\})]}{E[\min\{D,u\}]} \\
&= \frac{\sum_{d=0}^{u-1} dP_r[D=d]s_2(d) + uP_r[D \geq u]s_2(u)}{\sum_{d=0}^{u-1} dP_r[D=d] + uP_r[D \geq u]}.
\end{aligned}
$$

One then searches for the largest value of u that provides an average service level $\overline{s}_2(u)$ that is within a given standard.

A matlab function that calculates the average service level is shown in Figure 4.5.

```
function [s_2] = s_2(C,D,q,u,n)
  %s_2 fuction computes the average type 2 service level
  %for the binomial model
  %u is the overbooking level
  %C is the capacity
  %D is a 1xn matrix giving the distribution of the demand
  %n is the number of columns for matrix D
  %q is the probability that a reservation currently on
  %hand shows up at the time of service
  sum1=0;
  sum2=0;
  for d=1:(u-1)
    sum1=sum1+d*D(1,d+1)*s2(C,q,d);
    sum2=sum2+d*D(1,d+1);
  endfor
  dem_cdf = 0
  for j=u:(n-1)
    dem_cdf = dem_cdf + D(1,j+1);
  endfor
  sum1=sum1+u*dem_cdf*s2(C,q,u);
  sum2=sum2+u*dem_cdf;
  s_2=sum1/sum2;
endfunction
```

Figure 4.5: Matlab function for the average Type 2 service level

As we said above, we want to find the overbooking limit, i.e., the max u^*, for which our service criteria are fulfilled. So, a matlab function has been developed (Figure 4.6) that has the desired service level as input and the resulting optimal overbooking limit as output.

Example 12. *With $C = 100$ we have used the algorithm for finding max u for different cancellation probabilities and service levels. The distribution for the requests is given from following vector:*

$$
\mathrm{D} = \begin{bmatrix} 0 & \cdots & 0 & \underbrace{1/30}_{106} & \cdots & \underbrace{1/30}_{135} \end{bmatrix}
$$

In the following table we can see the maximum overbooking levels for each case and service type.

C (capacity)	q (probability of cancellation)	t (threshold service level)	Max u for each service type		
			s_1	s_2	$\overline{s}_2$
100	0.2	0.01	113	122	131
100	0.15	0.01	108	116	119
100	0.1	0.01	104	110	111
100	0.2	0.001	110	116	118
100	0.15	0.001	106	111	111
100	0.1	0.001	102	106	106

Let's take a closer look at our results:

1. *Generally, we can see that s_1 gives tighten booking limits comparing to s_2 and $\overline{s}_2$. The last one gives the bigger ones.*

2. *We can see that while the probability of cancellation decreases, i.e., we approaching the flight day, u decreases, as well. This result makes sense since there is big probability everyone to show up, so we want to deny service in the less customers possible.*

3. *Also, as the threshold decreases, u decreases again, which is expected, too, since we want to deny service in less customers, so we should sell less fees.*

Overbooking Based on Economic Criteria

In the previous part we saw how we can calculate overbooking limit based on service level. Here we will analyse another way for finding overbooking limits. Here, we want to maximize revenues and minimize losses. Thus, there is a need to estimate the revenue loss from not accepting additional reservations and the cost of denied service.

Suppose the number of shows is z and let $c(z)$ denote the denied-service cost. We shall assume that it is an increasing convex function of z. A common assumption in practice is

$$c(z) = h(z - C)^+, \tag{4.1}$$

where $h > 0$ constant, but undoubtedly an more realistic one is to assume strictly increasing marginal costs. This derives from the fact that for each additional customer,

```
function [maxu] = maxu(C,q,t,D,n,service_type)
#maxu fuction computes the maximum u
#so service level is under our threshold
#which is the variable t
#service type takes the values 1,2,3
#for s1,s2,s_2 respectively
u = C;
if service_type == 1
  while s1(C,q,u)<=t
    u = u + 1;
  endwhile
elseif service_type == 2
      while s2(C,q,u)<=t
        u = u + 1;
      endwhile
else
      while s_2(C,D,q,u,n)<=t
        u = u + 1;
      endwhile
endif
maxu = u-1;
endfunction
```

Figure 4.6: Matlab function for finding the max u for each service Type

who is denied service, the compensation should be increased.

Let r denote the fixed marginal revenue generated by accepting an additional reservation. Then the total expected profit from having u reservations on hand is given by

$$\pi(u) = ru - E[c(Z(u))],$$

where, as before, the random variable $Z(u)$ denotes the number of customers who show up on the day of service out of u reservations.

Proposition 2. *For the binomial model if $c(z)$ is convex, then $\pi(u)$ is concave in u.*

The following definitions and lemma will help with the proof of the above proposition.

Definition 3. *$X(\theta)$ is stochastically convex in the sample-path sense if for any four values θ_i for $i = 1, 2, 3, 4$ satisfying $\theta_2 - \theta_1 = \theta_4 - \theta_3$ and $\theta_4 \geq \max\{\theta_2, \theta_3\}$, there exist random variable X_i for $i = 1, 2, 3, 4$, such that X_i is equal in distribution to $X(\theta_i)$ and*

$$X_4(\omega) - X_3(\omega) \geq X_2(\omega) - X_1(\omega), \ \omega \in \Omega.$$

Lemma 2. *The sum of Bernoulli random variables is stochastically convex.*

Proof. Let $X(k) = \sum_{i=1}^{k} Y_i$, where Y_i are i.i.d. Bernoulli random variables. Let θ_i, for $i = 1, 2, 3, 4$, be integers satisfying $\theta_2 - \theta_1 = \theta_4 - \theta_3$ and $\theta_4 \geq \max\{\theta_2, \theta_3\}$. Note that $\theta_1 \leq \max\{\theta_2, \theta_3\}$ (else $\theta_4 < \max\{\theta_2, \theta_3\}$) and define

$$X_1 = \sum_{i=1}^{\theta_1} Y_i$$

$$X_3 = \sum_{i=1}^{\theta_3} Y_i$$

$$X_4 = \sum_{i=1}^{\theta_4} Y_i$$

$$X_2 = X_1 + (X_4 - X_3)$$

Note X_i is equal in distribution to $X(\theta_i)$ since each is the sum of θ_i i.i.d. Bernoulli random variables, and by construction

$$X_4 - X_3 = X_2 - X_1,$$

so $X(\theta)$ is stochastically convex in the sample path sense.
We know that showing $X(\theta)$ is stochastically convex in the sample path sense, implies that is stochastically convex. □

Definition 4. *If $X(k)$ is stochastically convex, then for any real valued, convex function $g(x)$, $E[g(X(k))]$ is convex in k.*

Proof. (for proposition 2)
We know that $Z(u)$ is Binomially distributed, namely is the sum of u i.i.d Bernoulli random variables, so $Z(u)$ is stochastically convex. Moreover $c(z)$ is a convex function, thus the $E[c(Z(u))]$ is convex in u. □

Note 1. *As a result we have that a maximizing u^* is the largest value of u satisfying*

$$\pi(u) - \pi(u-1) \geq 0$$

$$ru - E[c(Z(u))] - r(u-1) + E[c(Z(u-1))] \geq 0$$

$$E[c(Z(u))] - E[c(Z(u-1))] \leq r.$$

Calculating overbooking limits based on economic criteria has some difficulties because of the fact that it requires the estimation of the marginal revenues and costs. In the case of one class, marginal revenue is simply the common price, but for n-classes is more complex to estimate. A heuristic approach is to use the weighted-average revenue. But this approach faces the problem that generally the marginal revenue produced by an additional unit of capacity is not equal to the weighted-average revenue. Furthermore, the marginal revenue is typically decreasing while the available capacity decreases, thus the linearity assumption for the marginal revenue is violated.

As for the denied-boarding cost, in the case of monetary compensations, for example refund of the purchase price, is easy to quantify. On the other hand, if auctions are used to determine compensation, then this must be taken under consideration. As we can understand, in the case of auctions, it is more difficult to estimate marginal cost, since we should estimate the compensation. Moreover, vouchers for free service in the future require a more careful accounting, since the actual cost of providing the service is often less than the face value of the voucher.

The last component of calculating overbooking limits according to economic criteria is the goodwill loss of upsetting a customer. This is the impact that a denied-service will have in the customers future behaviour. For example, a customer may not trust this airline again, because he will be afraid of the possibility to be denied-service again. Another customer may give a bad review for the airline, which is also harmful for the company. As we can understand, due to the fact that this component is directly linked to the customer is the most difficult to quantify. However, it is usually worth an attempt to make this calculation to at least get the correct order of the importance of goodwill losses.

4.2.2 Static Model Approximations

As we saw in the previous section, binomial model is quite simple, but here we will analyse even simpler expressions for the overbooking limits.

Deterministic Approximation

At the deterministic approximation we set the overbooking limit so that the average show demand is exactly equal to the capacity, namely,

$$u^* = \frac{C}{q}.$$

This approximation is really simplistic, however there are several Revenue Management implementations that use it. Indeed, absent detailed service standards or cost information, a value around the deterministic level is not an unreasonable heuristic to use.

Normal Approximation

Another common approximation is to use the normal distribution, in which $F_u(x)$ is replaced by the normal distribution with mean, μ, and variance, σ^2, chosen to match the binomial, i.e.,

$$\mu_u = qu$$

$$\sigma_u^2 = uq(1-q).$$

The Type 1 service level is then approximated by

$$s_1(u) = 1 - \Phi(z_u),$$

where $z_u = \frac{C-\mu_u}{\sigma_u}$ and $\Phi(z)$ is the standard normal distribution.
A matlab function that calculates Type 1 service level for the normal approximation is shown in Figure 4.7.

```
function [sl_norm] = sl_norm(C,q,u)
  %sl_norm fuction computes the type 1 service level
  %for the standard normal approximation
  %C is the capacity
  %q is the probability that a reservation currently on
  %hand shows up at the time of service
  %u is the number of reservations on hand
  m = (1-q)*u;
  s = sqrt(u*q*(1-q));
  z = (C-m)/s;
  sl_norm=1-normcdf(z);
endfunction
```

Figure 4.7: Matlab function for Type 1 service level

The p.d.f. of the standard normal distribution is

$$\phi(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}}$$

and the c.d.f is

$$\Phi(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx.$$

Lemma 3. *The Type 2 service level is then approximated by*

$$s_2(u) = \frac{\sigma_u}{\mu_u}[\phi(z_u) - z_u(1 - \Phi(z_u))].$$

Proof. This follows from the fact that if the random variable $Z \sim N(\mu, \sigma^2)$, then

$$\begin{aligned}
E[(Z-C)^+] &= \int_C^\infty (Z-C)\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(Z-\mu)^2}{2\sigma^2}}dZ \\
&= \sigma\int_{\frac{C-\mu}{\sigma}}^\infty (z'\sigma + \mu - C)\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{z'^2}{2}}dz',\ where\ z' = \frac{Z-\mu}{\sigma} \\
&= \sigma\left[\int_{\frac{C-\mu}{\sigma}}^\infty z'\sigma\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{z'^2}{2}}dz' - \int_{\frac{C-\mu}{\sigma}}^\infty (C-\mu)\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{z'^2}{2}}dz'\right] \\
&= \sigma\left[-\int_{\frac{C-\mu}{\sigma}}^\infty \frac{2z'}{2}\frac{1}{\sqrt{2\pi}}e^{-\frac{z'^2}{2}}dz' - \frac{C-\mu}{\sigma}\left[1 - \Phi\left(\frac{C-\mu}{\sigma}\right)\right]\right] \\
&= \sigma\left[-\left[\frac{1}{\sqrt{2\pi}}e^{-\frac{z'^2}{2}}\right]_{\frac{C-\mu}{\sigma}}^\infty - \frac{C-\mu}{\sigma}\left[1 - \Phi\left(\frac{C-\mu}{\sigma}\right)\right]\right] \\
&= \sigma\left[\phi\left(\frac{C-\mu}{\sigma}\right) - \frac{C-\mu}{\sigma}\left[1 - \Phi\left(\frac{C-\mu}{\sigma}\right)\right]\right] \\
&= \sigma\left[\phi(z) - z(1 - \Phi(z))\right],
\end{aligned}$$

where $z = \dfrac{(C-\mu)}{\sigma}$. Thus,

$$\begin{aligned}
s_2(u) &= \frac{E[(Z(u) - C)^+]}{E[Z(u)]} \\
&= \frac{\sigma_u(\phi(z_u) - z_u(1 - \Phi(z_u))}{\mu_u}.
\end{aligned}$$

□

A matlab function that calculates Type 2 service level for the normal approximation is shown in Figure 4.8.

Example 13. *Implementing again Example 12 for the normal approximation, we got the following results:*
Compared to the results from Example 12 we can see that for the Type 1 service level we got smaller overbooking limits and for Type 2 service level we got the same as before.

```
function [s2_norm] = s2_norm(C,q,u)
  %s2_norm fuction computes the type 2 service level
  %for the standard normal approximation
  %C is the capacity
  %q is the probability that a reservation currently on
  %hand shows up at the time of service
  %u is the number of reservations on hand
  m = (1-q)*u;
  s = sqrt(u*q*(1-q));
  z = (C-m)/s;
  s2_norm=(s/m)*(normpdf(z)-z*s1_norm(C,q,u));
endfunction
```

Figure 4.8: Matlab function for Type 2 service level

			Max u for each service type	
C (capacity)	q (probability of cancellation)	t (threshold service level)	s_1	s_2
100	0.2	0.01	112	122
100	0.15	0.01	107	116
100	0.1	0.01	103	110
100	0.2	0.001	108	116
100	0.15	0.001	104	110
100	0.1	0.001	100	106

In Figure 4.9 we have a matlab function for finding the max u for each service Type for the normal approximation
The economic-based overbooking limit (??) for the constant-marginal-cost function (4.1) is approximated by choosing u^* to satisfy

$$\Phi_{u^*}(C) = 1 - \frac{r}{qh}. \tag{4.2}$$

4.2.3 Group Cancellations

As mixed-classes so groups, affect cancellation models. There is a positive correlation in cancellations between the members of a group, namely if a group decides to cancel then there is not only one cancellation, since all the members of the group will cancel. There are two techniques, we use in practice, to adjust for group size.

The first one, is to simply inflate the variance of the show distribution by a factor that accounts for group size. For the normal approximation to the binomial model the variance estimate is modified as $\sigma_u^2 = kuq(1-q)$, where k is a factor to account for group cancellations.

```
function [maxu_norm] = maxu_norm(C,q,t,service_type)
#maxu_norm fuction computes the maximum u
#for the normal approximation
#so service level is under our threshold
#which is the variable t
#service type takes the values 1,2,3
#for s1_norm,s2_norm
u = C;
if service_type == 1
  while s1_norm(C,q,u)<=t
    u = u + 1;
  endwhile
else service_type == 2
  while s2_norm(C,q,u)<=t
    u = u + 1;
  endwhile
endif
maxu_norm = u-1;
endfunction
```

Figure 4.9: Matlab function for finding the max

The other technique is based on moment-generating functions.

Definition 5. *A moment-generating function for a random variable X is*

$$G_x(t) = E[e^{tX}].$$

Note 2. *We know that*

$$E[X^n] = \frac{d}{dt} G_x(t)\Big|_{t=0}.$$

Let u be the number of total reservations and u_i denote the number of groups of size i, $i = 1, 2, .., n$. Then, $u = \sum_{i=1}^{n} u_i i$. Also, $Z(u)$ is the number of survivals from u total reservation, $Z_i(u_i)$ the number of survivals from u_i reservations of size i and q_i the probability of a group of size i to survive. Then, assuming that $Z_i(u_i)$ are independent random variables for $i = 1, 2, .., n$, the generating function of $Z(u)$ is

$$\begin{aligned}
G_{Z(u)}(t) &= E\left[e^{tZ(u)}\right] \\
&= E\left[e^{t\sum_{i=1}^{n} iZ_i(u_i)}\right] \\
&= E\left[e^{tZ_1(u_1)} e^{t2Z_2(u_2)} ... e^{tnZ_n(u_n)}\right] \\
&= \prod_{i=1}^{n} E\left[e^{tiZ_i(u_i)}\right] \\
&= \prod_{i=1}^{n} P_{z_i}(u_i)(e^{it}) \\
&= \prod_{i=1}^{n} (1 - q_i - q_i e^{it})^{u_i}
\end{aligned}$$

4.3 Combined Capacity-Control and Overbooking Models

We now have to analyse the overbooking problem taking into consideration the interaction of overbooking decisions with capacity controls. Due to the fact that including cancellations and no-shows is too difficult theoretically, one makes the following set of assumptions:

1. The cancellation and no-show probabilities are the same for all customers.

2. Cancellations and no-shows are mutually independent across customers and in any period are independent of the time a reservation was accepted.

3. The refunds and denied service costs are the same for all customers.

From the above man can notice that the number of no-shows and the costs incurred are only a function of the total number of reservations on hand, so we only need to retain a single state variable.

As a result of the fact that cancellation options and penalties are often linked directly to a booking class, i.e. rates and costs can be vary significantly from one class to the other, the assumptions 1 and 2 are restrictive. Ideally, we should take account of these differences when we make allocation decisions, however, this significantly complicates the problem. Moreover, we have already see that reservations from people in groups typically cancel at the same time, so the assumption of independence among customers is unrealistic.

Mainly, the overbooking problem is separated from the capacity-allocation problem, because in a approximate static overbooking model we are able to relax some or all parts of the above assumptions.

4.3.1 Exact Methods for No-Shows Under Assumptions

Here we consider that there are only no-shows without cancellations. We define as q_0 the probability of a show-up, thus $1 - q_0$ is the no-show probability. Due to the above assumptions, we know that q_0 is assumed to be the same for all customers and independent of when the reservation was made.

Given x reservations on hand at the time of service, we denote as

$$Z_0(x) = \sum_{i=1}^{x} Z_i$$

the number of customers who show up at time zero, where

$$Z_i = \begin{cases} 1, \text{if customer } i \text{ shows up for service} \\ 0, \text{otherwise.} \end{cases}$$

As it emerges from our assumptions, $Z_0(x)$ is a Binomial (q_0,x) random variable, with

$$P_r[Z_0(x) = z] = \begin{pmatrix} x \\ z \end{pmatrix} q_0^z (1 - q_0)^{x-z},\ z = 0, 1, ..., x,$$

and, moreover, the total cost of denied service is only a function of the show demand. Let $c(z)$ denote the overbooking cost given z. We will require the $c(z)$ be increasing and convex with $c(0) = 0$, which is quite natural since the marginal cost of denying service to customers tends to increase with the number denied.

Specifically, we assume that we have a simple linear cost per denied customer $c(z) = h(z - C)^+$. Suppose we have x reservations on hand at the time of service, then the expected cost of service is given by

$$V_0(x) = E[-c(Z(x))],\ x \geq 0.$$

Proposition 3. *$V_0(x)$ is concave in x if $c(\cdot)$ is convex.*

Proof. We know that if $c(\cdot)$ is convex, then $-c(\cdot)$ is concave. Thus, $E[-c(\cdot)]$ is concave.

□

Static Model

Here we will use again the static n-class model from Chapter 2. Recall: in an n-class model the classes are ordered $r_1 > r_2 > ... > r_n$ and we assume that first arrives the class with the lowest revenues and last the one with the highest revenue. Classes and stages are indexed by j. Now, the state variable, x, is defined to be the number of reservations on-hand and not the remaining capacity.

The Bellman equation (2.2) for the static model with no-shows is the following

$$V_j(x) = E\left[\max_{0 \leq u \leq D_j} r_j u + V_{j-1}(x + u)\right],$$

with boundary conditions $V_0(x) = E[-c(Z(x))]$, $x \geq 0$, where here $V_j(x)$ is now interpreted as the expected *net benefit*, i.e. expected revenue minus the expected terminal cost, of operating the system from period j onward given that there are x reservations on hand. In this case, $V_j(x)$ is a decreasing function of x, since the more reservations we have on hand now, the fewer future opportunities to collect revenue and/or the higher the expected future terminal costs.

As in Chapter 2, considering that $V_0(x)$ is concave, we can show that the value function $V_j(x)$ is concave in x for all j and x. In this case we will express the optimal policy using booking limits, due to the fact that we don't have any capacity constraint. The optimal nested booking limits are given by

$$b_j^* = \min\{x \geq 0 : r_j < \Delta V_{j-1}(x)\},\ j = 1, ..., n-1,$$

where $\Delta V_{j-1}(x) = V_{j-1}(x) - V_{j-1}(x+1)$ now is the marginal opportunity cost of holding another reservation in Stage $j-1$. It is then optimal to accept Class j if and only if the number of reservations on hand, x, is strictly less than b_j^*, i.e. $x < b_j^*$.

4.3.2 Class-dependent no-show refunds

By relaxing one or more assumptions, then the problem becomes significantly difficult, because of the increase in dimensionality of the dynamic problem. For instance, if no-show rates or costs depend on customer class or the time of purchase (or both), then we have to retain a state variable for each class or each time period (or both). However, it occurs that class-dependent refunds can be easily integrated through an appropriate change of variable.

Thus, we relax this assumption but we still keep all the others. Suppose a customer of Class j who no-shows in period 0 is given a refund h_{j_0}, with $h_{j_0} < r_j$. An unsophisticated way is to keep in track each class separately so that refunds can be properly awarded at the time of service. We know that a given customer no-shows is completely independent of all other decisions and events in the system, thus we can in fact charge for the expected refund at the time the reservation is accepted. Namely, if we accept a customer of Class j, they will yield an reduced revenue of

$$\widehat{r}_j = r_j - (1 - q_0)h_{j_0}$$

independent of everything else in the system. Therefore we simply use $\widehat{r}_j$ in place of r_j. Man can notice that depending on the refund, the ordering of $\widehat{r}_j$ may be different than r_j, thus it's possible the optimal policy in this case may reject the high gross revenue customer in favour of the high net revenue one.

4.3.3 Exact Methods for Cancellations Under Assumptions

The cancellation model is quite more difficult than the no-shows one, but under our assumptions we can manage it.

Static Model

Let q_j denote the probability that a reservation in the system at the start of period j survives to period $j+1$. Then $1-q_j$ is the probability that a reservation cancels in period j, which are the same and independent for all customers and are independent of the age of the reservation. Let $Z_j(x)$ denote the number of reservations that survive period j given that there are x reservations on-hand, thus $x - Z_j(x)$ are the number of cancellations.

We alter the Bellman equation (2.2) for the static model, so it accounts for cancellations:

$$V_j(x) = E\left[\max_{0\leq u\leq D_j}\{r_j y + H_{j-1}(x+u)\}\right],$$

with boundary conditions $V_0(x) = E[-c(Z(x))]$, $x \geq 0$, where

$$H_{j-1}(x) = E[V_{j-1}(Z_j(x))] = \sum_{z=0}^{x}\left(\begin{array}{c} x \\ z \end{array}\right) q_k^z(1-q_k)^{x-z}V_{j-1}(z),$$

is the expected value function after cancellations in period j.

Comment 6. *We know that if $V_{j-1}(z)$ is concave in z, then $H_{j-1}(x)$ is concave in x and hence the value function $V_j(x)$ is concave in x.*

As before is optimal to use nested booking limits. The optimal booking limits are given by

$$b_j^* = \min\{x \geq 0 : r_j < H_{j-1}(x) - H_{j-1}(x+1)\},\ j = 1, ..., n-1,$$

where it is optimal to accept Class j if and only if the number of reservations on hand, x, is strictly less than b_j^*.

4.3.4 Class-dependent cancellation refunds

As in no-shows case, relaxing the assumption that no-show rates or costs depend on customer class or the time of purchase (or both) requires expanding the state space. Again, a change of accounting can be used to allow for class-dependent refunds.

Again we hold all the assumptions except the one for class-dependent. Suppose a customer of Class j who cancels in period t is given a refund h_{j_t}, with $h_{j_t} < r_j$. In the same manner as the no-show case, we can charge for the expected refund at the time the reservation is accepted, rather than at the time of service. We define $G_t(j)$ the expected refund given to a Class j reservation from period t through to the time of service, and is given by:

$$G_t(j) = (1 - q_t)h_{j_t} + q_t G_{t-1}(j),\ t\ = 1, 2, ..., T,$$

with boundary condition

$$G_0(j) = (1 - q_0)h_{j_0}.$$

We then form the reduced revenue

$$\widehat{r}_{j_t} = r_i - G_t(j)$$

again we simply use $\widehat{r}_{i_t}$ in place of r_i and the ordering may be different.

Comment 7. *In both cases, no-shows and cancellations, we should use the* $\widehat{r}_{i_t}$*, because even if though a class gives a higher current revenue, much of that revenue may be forfeited on average, so the net benefit of accepting it can be quite different from the gross revenue.*

Bibliography

[1] P. P. Belobaba. Air Travel Demand and Airline Seat Inventory Management. PhD thesis, Flight Transportation Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusettes, 1987.

[2] P. P. Belobaba. Airline yield management: An overview of seat inventory control. Transportation Science, 21:63−73, 1987.

[3] P. P. Belobaba. Application of a probabilistic decision model to airline seat inventory control. Operations Research, 37:183−197, 1989.

[4] R. Phillips. Pricing and Revenue Optimization. Stanford University Press, 2005

[5] K. T. Talluri, G. J. Van Ryzin. The Theory and Practice of Revenue Management. Kluwer Academic Publishers, 2002