

An Application of Yield Management  
to the Hotel Industry

by  
Gabriel R. Bitran  
Susana V. Mondschein

WP #3585-93

July 1993

# 1 INTRODUCTION

In this paper we study optimal selling strategies when a producer faces a stochastic and dynamic arrival of customers from different market segments, considering a fixed capacity and a finite planning horizon. We formulate the problem as a stochastic, dynamic programming model and characterize the optimal policies as functions of the capacity and the time left until the end of the planning horizon. We consider two features that enrich the problem: (i) we make no assumptions concerning the particular order between the arrivals of different types of customers and (ii) we allow for multiple types of products and downgrading. This is the case when there exists a natural ordering of all products, so that every product is an acceptable substitute for those that are “worse” than it.

This paper is motivated by the yield management problem that arises at the operational level in the hotel industry. Hotel managers have to decide the best use of the limited number of rooms. As defined by American Airlines (1987) the objective of yield management is “to maximize passenger revenue by selling the right seats to the right customers at the right time,” which can be applied to our problem replacing seats by rooms.

Decisions are made at two levels when solving this problem. The first level is tactical: the manager must decide the maximum number of reservations for each market segment that she accepts at a given moment in time for a particular target date, i.e., given that the manager has already accepted a number of reservations at a certain point in time and considering future stochastic requests and cancellations, she must decide whether or not to accept a specific reservation request. Usually, customers can cancel their reservations without any penalty for doing so. Thus, they do not pay for the rooms until the actual date when they use them.

This tactical problem is closely related with the second level at which decisions are made, namely the operational level. At this level every time a customer requests a room during the target date, the manager has to decide whether or not to rent it to that customer, considering the number of reservations made at the tactical level and potential customers that show up without reservations (*walk-ins*). When making this decision, the manager does not know how many additional customers will arrive that day, and whether among those who arrive there will be some for whom the room currently being requested has a higher value. Special emphasis is given to the manager’s ability to downgrade rooms, for example, the manager may decide to give a customer a suite for the price of a standard room when the latter is not available. The perishability of the product plays a central role in the decision process; rooms that are not rented have a zero opportunity cost. At the tactical level no revenues are collected; it is at the operational level that rooms are actually rented and

revenues are collected.

At the operational level the manager knows the total number of reservations within each market segment. However, the resulting number of reservations that turn into sales is a random variable because of the no shows. Hotels usually have two types of reservations: the 6 p.m. hold and guaranteed reservations. In the 6 p.m. hold reservation case a room is reserved until 6 p.m. and it costs nothing for the customer if she does not show up. On the other hand, in the guaranteed reservation case customers frequently must guarantee their reservations with a credit card. Despite the fact that in practice hotels cannot always force customers to pay for the room if they do not show up, the show-up rate is very large in this case. Therefore, due to the cancellations at the tactical level and no shows at the operational one, managers usually overbook in order to maximize the total expected profit. When doing this they must take into account the trade-offs of having idle capacity due to no shows and cancellations, and turning down customers with the corresponding rejection costs.

One of the key factors for successful management in the hotel industry is market segmentation as stated in Merliss and Lovelock (1980). Usually, a segmentation considers at least three categories: tourists, corporate travelers and groups. However, some hotels have explored a “finer” classification considering more than 10 different categories as for example pure transient, executive service plan, government workers, mini-vacations, corporate groups, airlines, and others. Each of these segments can be characterized by the price customers pay for the room, the type of room that they request and the rejection costs associated to turning them down. In general, this cost is difficult to quantify because it is a combination of monetary and non-monetary costs. For example, if a customer with a reservation for an hotel room is turned down, the rejection cost may be the monetary cost of allocating that customer to another hotel combined with the non-monetary cost associated with the loss in trustworthiness for failing to honor a reservation.

Managers often use downgrading as a complementary method to match the limited capacity of the hotel with the uncertain demand experienced during the target date. Hotels usually have different types of rooms which differ in their quality (size, furniture, service, etc.) as for example suites, deluxe and standard rooms. Hence, there exists a natural order for the rooms where any room can be substituted for all those that are better than it. Therefore, downgrading adds a new degree of flexibility to the room assignment process.

In this paper we focus in the operational level. Most of the literature that studies the tactical problem considers simplified dynamic relationships about what happens during the target date. For example, some assume that managers have perfect information on arrivals during the target

date while others suppose that customers show up in a specific sequence (e.g., customers that pay less show up first). Among the papers that consider the problem at a tactical level, Ladany (1976) proposes a dynamic programming formulation for managing reservations in the hotel industry; at each stage a random number of reservation requests and cancellations are received. He assumes that during the target date, all the customers arrive at the same time. Hence, the manager can do perfect price discrimination at the operational level. Alstrup et. al. (1986) study the booking policy for a single flight leg with two types of customers. They also assume that during the target date, all customers arrive together. Bitran and Gilbert (1992) extend Ladany's analysis to the case where customers do not arrive at the same time, assuming a specific order of arrival among various types of customers. In practice, the arrival of customers from different market segments overlaps during the target date and therefore managers must make the renting decisions in real time, considering the trade offs discussed previously. The models presented in our paper consider this behavior as a central feature in the decision process.

Although in this paper our focus is on the operational problem, the same methodology applies to the tactical problem when cancellations are not allowed. In this case reservations and actual sales become equivalent and there is no distinction between the decisions made at the tactical and operational levels. Most of the literature that studies the tactical problem in the airline and hotel industries does not model stochastic cancellations explicitly. For example, see the references in Weatherford and Bodily (1992) for an extensive research overview in the yield management area. In this context our models make a contribution when considering general patterns for the arrival process of customers from different market segments and multiple types of products with the possibility of downgrading.

One of the first papers that addresses this tactical problem is Littlewood (1972). The author analyzes the problem of selling seats in the airline industry. He determines the maximum number of seats that can be sold to customers paying lower fares in order to maximize expected profits; he assumes two classes of customers where discount customers book their seats first. Optimization is done over a single flight leg. Littlewood's results have been extended in various directions: Belobaba (1989) considers an arbitrary number of fare classes, Brumelle et al. (1990) analyze separately the cases of dependent demand (the demand of discount fare customers gives information about the future demand of full fare customers), goodwill cost and spill rate (the proportion of full fare customers that is turned down), and upgrading (a discount fare customer decides to pay the full fare if there is no available seats). All the models mentioned above assume that different types of customers arrive in a preestablished order.

Hotels generally assign capacity for group requests (tours, conventions and others) a long time

in advance before the target date. It is not unusual to book for conventions with more than 6 months anticipation. These decisions are made at a strategic level and therefore they only reduce the total capacity available at the tactical and operational levels. Given the risk involved in this large block of reservations, rooms usually have to be pre-paid eliminating the problem of no shows. Thus, as a good approximation of reality we consider only single room requests at the operational (or tactical) level.

We also consider single night stays. Despite the fact that multiple day reservations occur in practice, there is evidence that shows that individual dates of a multiple-day reservation cancel or show-up independently, see Bitran, Gilbert, and Leong (1992). Additionally, there are only some critical days during the week when the hotel is likely to be fully booked. Therefore, the effort to assign optimally the hotel's capacity should be concentrated on those days.

In this paper we present three models that differ in the degree of complexity in terms of the assumptions considered. The first model determines the optimal policy for renting rooms when there is only one type of room and there is no reservations made at the beginning of the target date (customers show up without making previous reservations). Then, we extend the results for the case of multiple types of rooms with the possibility of downgrading. These two models provide a basis for building the final model that considers multiple types of rooms and a given number of reservations at the beginning of the target date. The arrivals of different types of customers is stochastic. We make no assumptions concerning the particular order between the arrivals of different types of customers. We determine whether, at a given moment in time, the hotel manager should rent a room to a particular type of customer, and, should the answer be positive, what room should be rented.

We derive an optimal selling policy for this problem, which has a number of interesting properties. It is characterized by a collection of capacity threshold vectors that evolves over time (every vector in the set has as many components as the number of products considered). It is optimal to satisfy a customer's request as long as the actual capacity vector at the moment the request is made has all components larger than or equal to at least one of the vectors in the current set of threshold capacities. Furthermore, the collection of capacity threshold vectors evolves over time in such a way that, for every class of customers and every actual capacity vector, there exists an instant in time beyond which it is optimal to satisfy the customer's request.

The computational effort required to solve the problem can be significantly reduced using the properties described above, making problems with one type of room tractable. However, when real size problems with several types of rooms are considered, the computational burden becomes

enormous even using this approach. Ad-hoc heuristics are developed to handle those cases.

The remainder of this paper is organized as follows. In Section 2 we introduce a dynamic programming formulation for the general problem described above. We also characterize the optimal policy for renting rooms in this section. In Section 3 we find a family of upper bounds for the optimal solution. These bounds are useful when comparing various heuristics to solve the general problem. In Section 4 we describe the heuristics developed to solve the general optimization problem. Section 5 contains the computational experiments that show the performance of the different heuristics. Some of these experiments are performed using real data from a medium size hotel. Finally, in Section 6, we present the conclusions.

## 2 MATHEMATICAL FORMULATION

In this section we present the mathematical formulation of the problem of finding the optimal policy for renting hotel rooms to various classes of customers. We specify three different models, which vary in the degree of complexity in terms of the number of products' types and customers' classes considered. The first model considers multiple classes of customers that request a single type of room at different prices. The second model incorporates multiple types of rooms products with the possibility of downgrading. Finally, we extend the latter model to the case where reservations are allowed prior to the target date. The first two models are also appropriate to solve the tactical problem when cancellations prior to the target date are not a major factor.

The Poisson process is a natural model to represent the arrival process of customers to the hotel. This process is commonly used in the literature; for example Alstrup et al. (1986) claim that airline ticket requests have this distribution. Rothstein (1974) and Bitran and Gilbert (1992) use a Poisson distribution to represent customer requests in the hotel industry. Generally, the arrival rate of customers from each market segment varies during the day. For example, just few customers with 6 p.m. hold reservations arrive after 6 p.m. Therefore, in this paper we consider a general non homogeneous Poisson process to describe the arrival process of customers with arrival rates that depend on the market segment and time.

A special consideration is incorporated for those market segments that have reservations because the total number of arrivals is bounded by the total number of reservations. Therefore, we use a truncated Poisson process to describe the arrival process from these customer classes where the interarrival time is exponential and there is no more arrivals in the cases when all customers show up.

In what follows we present a stochastic and dynamic programming formulation, where the stages

correspond to the periods in which the planning horizon is divided. The state of the system, at a given time, is determined by the current capacity, the class of the customer requesting a room, and the number of pending reservations (if reservations are allowed). We divide the planning horizon into time intervals small enough so that we can assume that the number of arrivals in each interval is either zero or one. This approximation is based on an exact formulation which considers that decisions are only made at the random instants when customers arrive. Due to the complexity for solving this continuous time formulation, we formulate an approximate model where decisions are made at discrete intervals of time. Reducing the length of the time intervals we can get as close as we want to the continuous time formulation.

Products are classified into ordered types  $s \in \{1, \dots, m\}$ , where product 1 is the “best product” (product 1 can substitute all other products) and  $m$  is the “worst.” There are  $n$  classes of customers; in the most general case, a class is determined by the product type requested by the customer, the price that she pays, the rejection cost, and whether or not the customer has a reservation. Several classes of customers can request the same type of product. Therefore, we define  $A_s$  as the set of all types of customers that request product  $s$ . Before presenting the mathematical models, we introduce the following notation:

1.  $\pi_i$ : Price associated to a class  $i$  customer.
2.  $c_i$ : Rejection cost associated to a class  $i$  customer.
3.  $C$ : Total number of rooms. For the case of multiple types of rooms,  $C$  is an  $m$ -dimensional vector with  $C(s)$  equal to the number of available class  $s$  rooms.
4.  $T$ : Planning horizon. For the operational problem  $T$  corresponds to the target date.
5.  $R$ : Vector with pending reservations for each class of customer.
6.  $e_p$ : Vector with a one in the  $p^{\text{th}}$  position and zeros elsewhere.
7.  $E_x[f(x, y)]$ : expected value of  $f(x, y)$  with respect to the random variable  $x$ .

## 2.1 Model Without Reservations

In what follows, we analyze the case where cancellations are not allowed. Therefore, the operational and tactical problems in the hotel industry become indistinguishables. We first consider the single type room case, and later we extend the results to the multiple types of rooms case where downgrading is possible.

### 2.1.1 Single product case

We define the function  $F_t(c, i)$  as the maximum expected profit from period  $t$  onwards if there are  $c$  available rooms and a class  $i$  customer requests a product at time  $t$  ( $i = 0$  corresponds to no arrivals). Without loss of generality we assume that the planning horizon is divided in time intervals of length equal to one. Therefore, the mathematical formulation corresponds to:

$$\text{PROBLEM SP}(C): \quad E_i[F_0(C, i)],$$

where:

1) If  $c > 0, i \neq 0$ :

$$F_t(c, i) = \max \begin{cases} \text{Rent a room:} & \pi_i + E_j[F_{t+1}(c-1, j)], \\ \text{Reject the customer:} & -c_i + E_j[F_{t+1}(c, j)]. \end{cases}$$

2) If  $c = 0, i \neq 0$ :

$$F_t(c, i) = -c_i + E_j[F_{t+1}(c, j)].$$

3) If  $i = 0$ :

$$F_t(c, i) = E_j[F_{t+1}(c, j)].$$

Boundary conditions:

$$F_T(c, i) = \begin{cases} 0 & \text{if } i = 0, \\ \pi_i & \text{if } i \neq 0 \text{ and } c > 0, \\ -c_i & \text{if } i \neq 0 \text{ and } c = 0. \end{cases}$$

Hence,  $\text{SP}(C)$  is equal to the maximum expected profit during the planning horizon, considering that the initial capacity is equal to  $C$ . Given an arrival and the current capacity, at every time period the model gives the optimal policy of whether or not to accept the request.

In what follows, we characterize the optimal policy given by  $\text{SP}(C)$ . The producer faces a non-trivial decision only when there is a positive capacity and someone requests a room. The objective function in this case is given by,

$$F_t(c, i) = \max\{\pi_i + E_j[F_{t+1}(c-1, j)], -c_i + E_j[F_{t+1}(c, j)]\}.$$

Hence, the hotel rents the room if

$$\pi_i + c_i \geq E_j[F_{t+1}(c, j)] - E_j[F_{t+1}(c-1, j)],$$

and rejects the customer if

$$\pi_i + c_i < E_j[F_{t+1}(c, j)] - E_j[F_{t+1}(c-1, j)].$$



Defining  $\alpha_t(c) = E_j[F_{t+1}(c, j)] - E_j[F_{t+1}(c - 1, j)]$ , we obtain that the hotel's policy for a class  $i$  request at time  $t$ , given a capacity  $c$ , is to rent the room if  $\pi_i + c_i \geq \alpha_t(c)$  and reject the customer if  $\pi_i + c_i < \alpha_t(c)$ .

The following proposition shows that associated with every instant in time  $t$ , there exists a threshold  $C_{it}^*$  such that a class  $i$  customer is accepted as long as the current capacity is larger than or equal to this threshold. The threshold  $C_{it}^*$  is determined by equating the expected profits from accepting and rejecting the customer's request, or equivalently:  $\alpha_t(C_{it}^*) = \pi_i + c_i$ . The threshold  $C_{it}^*$  decreases when the opportunity cost of accepting the customer's request,  $\pi_i + c_i$ , increases, i.e., if for a given capacity the optimal decision is to accept a specific type of customer then it is also optimal to accept any type of customer with a higher opportunity cost.

**Proposition 1** *The function  $\alpha_t(c)$  is a non-increasing function of  $c$ :*

$$\alpha_t(c - 1) \geq \alpha_t(c).$$

PROOF: See Appendix 1. This is a particular case of Proposition 3. ■

The following proposition shows that, for a given capacity  $c$  and type  $i$  customer, there exists a threshold time  $\tau_{ic}$  beyond which the customer's request is accepted. The threshold  $\tau_{ic}$  is obtained equating  $\alpha_t(c)$  and  $\pi_i + c_i$ . As we show in proposition 2,  $\alpha_t(c)$  is a non-increasing function of  $t$ . Hence, if we take two classes of customers  $i$  and  $j$  with  $\pi_i + c_i \geq \pi_j + c_j$ , then  $\tau_{ic} \leq \tau_{jc}$ , i.e., the hotel rents the room to customers with a higher opportunity cost before renting it to customers that yield a lower opportunity cost.

**Proposition 2** *The function  $\alpha_t(c)$  is decreasing in  $t$ :*

$$\alpha_t(c) \geq \alpha_{t+1}(c)$$

PROOF: See Appendix 1. This is a particular case of Proposition 4. ■

The simple structure of the optimal policy can be summarized as follows. For a given instant in time and customer class, there exists a capacity threshold such that a request within that class is accepted as long as the current capacity is larger than the threshold. Similarly, there exists a time threshold for every capacity and customer class, such that a request within that class that happens after this time threshold is always accepted.

The main characteristic of the optimal policy is its simplicity. This feature has an important consequence in terms of its practicality, because, in general, managers are more open to incorporate rules that are intuitive and easy to verify.

### 2.1.2 The multiple product case

The next model corresponds to the case of multiple types of rooms with the possibility of downgrading. Similarly to the previous case, cancellations are not allowed. We define  $F_t(c, i)$  as the maximum expected profit from period  $t$  onwards when there are  $c$  available rooms and a class  $i$  customer arrives at time  $t$  ( $i = 0$  corresponds to no arrivals). In this case  $c$  is a vector with as many components as room types.

The mathematical formulation for this model corresponds to:

$$\text{PROBLEM MP}(C): \quad E_i[F_0(C, i)],$$

where, assuming that a class  $i$  customer requests a type  $s$  room, i.e.  $i \in A_s$ , we have:

1) If  $i \neq 0$  and there exists  $c(k) > 0$  s.t.  $k \leq s$  :

$$F_t(c, i) = \max \begin{cases} \text{Rent a room: } \max_{1 \leq p \leq s} \pi_i + E_j[F_{t+1}(c - e_p, j)], \\ \text{Reject the customer: } -c_i + E_j[F_{t+1}(c, j)]. \end{cases}$$

2) If  $i \neq 0$  and  $c(k) = 0$  for all  $k \leq s$  :

$$F_t(c, i) = -c_i + E_j[F_{t+1}(c, j)].$$

3) If  $i = 0$

$$F_t(c, i) = E_j[F_{t+1}(c, j)].$$

Boundary conditions:

$$F_T(c, i) = \begin{cases} \pi_i & \text{if } \exists k \in \{1, 2, \dots, k\} \text{ s.t. } c(k) > 0 \text{ and } i \neq 0, \\ -c_i & \text{if } \forall k \leq s, c(k) = 0 \text{ and } i \neq 0, \\ 0 & \text{if } i = 0. \end{cases}$$

We observe that in the multiple product case the manager not only has to decide whether or not to accept the request but also what room to rent. However, using the property that the objective function is a non-decreasing function of the capacity (see Lemma 1 in Appendix 1) it is easy to prove that if the optimal decision is to rent a room to a class  $i$  customer that requests a type  $s$  room, then the hotel must rent the “lowest level” available room the customer is willing to accept (this room is given by  $p = \max\{1, 2, \dots, s\}$  s.t.  $c(p) > 0$ ). That is, if the hotel decides to rent a room to a customer, it rents the room satisfying the customer’s needs that is closest to the room requested by the customer.

In what follows we derive the properties that characterize the optimal policy. Similarly to the single product case, the manager faces a non trivial decision when a type  $i$  customer requests room  $s$  and there is an available room that satisfies this request. Then, the optimal decision is given by:

$$F_t(c, i) = \max\{\pi_i + E_j[F_{t+1}(c - e_p, j)], -c_i + E_j[F_{t+1}(c, j)]\},$$

where  $p = \max\{1, 2, \dots, s\}$  subject to  $c(p) > 0$ . Hence, the hotel rents the room if:

$$\pi_i + c_i \geq E_j[F_{t+1}(c, j)] - E_j[F_{t+1}(c - e_p, j)]$$

and rejects the customer otherwise.

Defining  $\alpha_t(c, p) = E_j[F_{t+1}(c, j)] - E_j[F_{t+1}(c - e_p, j)]$ , the optimal policy consists of renting the room if  $\pi_i + c_i \geq \alpha_t(c, p)$ , and rejecting the customer if  $\pi_i + c_i < \alpha_t(c, p)$ . The following two propositions characterize the optimal policy.

**Proposition 3** : *The function  $\alpha_t(c, p)$  is non-increasing as a function of  $c$ :*

$$\alpha_t(c - e_l, p) \geq \alpha_t(c, p) \quad \forall c, p, l, t.$$

PROOF: See Appendix 1. ■

**Proposition 4** *The function  $\alpha_t(c, p)$  is non-increasing as a function of  $t$ :*

$$\alpha_t(c, p) \geq \alpha_{t+1}(c, p) \quad \forall c, p, t.$$

PROOF: See Appendix 1. ■

We illustrate the properties described above with the following example. Consider a hotel that has two types of rooms, where type 1 rooms can substitute type 2 rooms. For a given period of time, and a given class  $i$  of customers, figure 1 shows the collection of capacity threshold vectors. This corresponds to the curve that separates the acceptance and the rejection regions. If the current capacity is in the acceptance region, the manager accepts the class  $i$  request. Otherwise, she rejects the request. Assuming that class  $i$  customers request a type 2 room, downgrading only takes place when there is no type 2 rooms available in the acceptance region. As time goes by, the curve that corresponds to the set of capacity threshold vectors moves down, and the rejection region becomes smaller.

The two preceding propositions show how the results obtained for the single product case extend to the multiple product case with downgrading. The optimal policy for renting rooms in this case is characterized by a collection of capacity threshold vectors that evolves over time (every vector in the set has as many components as the number of room types considered). It is optimal to satisfy a customer's request as long as the capacity vector at the moment the request is made has all components larger than or equal to at least one of the vectors in the current set of threshold

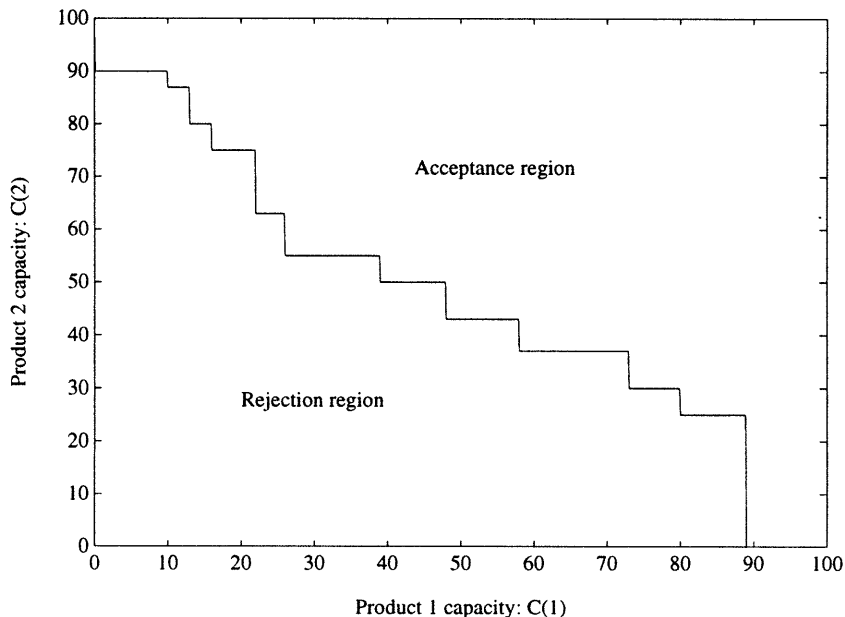


Figure 1: Example of a collection of capacity threshold vectors

capacities. Furthermore, the collection of capacity threshold vectors evolves over time in such a way that, for every class of customers and every actual capacity vector, there exists an instant in time beyond which it is optimal to satisfy the customer's request.

Proposition 3 implies that if a customer's request is accepted when the capacity is equal to  $c - e_l$  then it is also accepted when the capacity is equal to  $c$ , for all values of  $c$  and  $l$ . Suppose that the request is satisfied with a type  $p$  room when the capacity is  $c - e_l$ . When incrementing the capacity to  $c$  this request will be satisfied with the same type  $p$  room or with a type  $l$  room if this room is closer to the customer's need. This property is obtained by observing that:

$$\alpha_t(c, p) \geq \alpha_t(c, l) \quad \forall p \leq l.$$

The above inequality uses the fact that the function  $F_t(c, i)$  is non-increasing in  $c$  (see Lemma 1 in Appendix 1).

## 2.2 Model with Reservations

In this section we solve the operational problem that arises during the target date where the manager knows the total number of reservations within each market segment. The demand associated to these customers is stochastic because some of them do not show up, even though they have reservations. Additionally, during the target date, the hotel receives requests of walk-ins, i.e. customers that show up without making a previous reservation.

At every instant in time, the demand associated to customers that have reservations depends on the number of pending reservations, i.e., the number of reservations that have not been already requested. Thus, in this case, the dynamic, stochastic programming model includes an additional state variable that corresponds to the number of pending reservations,  $R$ , which has as many components as the number of customer classes.

We define  $F_t(c, R, i)$  as the maximum expected profit from period  $t$  onwards, if the hotel starts at period  $t$  with  $c$  available rooms,  $R$  pending reservations and a class  $i$  customer requests a room. We recall that the customer class determines the price, the rejection cost, the room that the customers requests, and whether she has a reservation or not. We define  $\delta_i$  equal to 1 when class  $i$  corresponds to customers with reservations and  $\delta_i$  equal to 0 otherwise.

Considering the general multiple product case, the mathematical formulation is equal to:

$$\text{PROBLEM MPR}(C, R): \quad E_i[F_0(C, R, i)],$$

where, assuming that a class  $i$  customer requests a type  $s$  product, i.e.  $i \in A_s$ , we have:

1) If  $i \neq 0$  and there exists  $c(k) > 0$  s.t.  $k \leq s$  :

$$F_t(c, R, i) = \max \begin{cases} \text{Rent a room: } \max_{1 \leq p \leq s} (\pi_i + E_j[F_{t+1}(c - e_p, R - \delta_i e_i, j)]), \\ \text{Reject the customer: } -c_i + E_j[F_{t+1}(c, R - \delta_i e_i, j)]. \end{cases}$$

2) If  $i \neq 0$  and  $c(k) = 0$  for all  $k \leq s$  :

$$F_t(c, R, i) = -c_i + E_j[F_{t+1}(c, R - \delta_i e_i, j)].$$

3) If  $i = 0$

$$F_t(c, R, i) = E_j[F_{t+1}(c, R, j)].$$

Boundary conditions:

$$F_T(c, R, i) = \begin{cases} \pi_i & \text{if } \exists k \in \{1, 2, \dots, k\} \text{ s.t. } c(k) > 0 \text{ and } i \neq 0, \\ -c_i & \text{if } \forall k \leq s, c(k) = 0 \text{ and } i \neq 0, \\ 0 & \text{if } i = 0. \end{cases}$$

Similarly to the previous cases, we define:

$$\alpha_t(c, p, R) = E_j[F_{t+1}(c, R, j) - F_{t+1}(c - e_p, R, j)].$$

The hotel rents a room to a class  $i$  customer if  $\pi_i + c_i \geq \alpha_t(c, p, R - \delta_i e_i)$ , and rejects the customer otherwise.

The following proposition shows that, similarly to the case without reservations, for a given customer class, instant in time, and number of pending reservations, there is a set of capacity threshold vectors such that the customer is accepted as long as the current capacity vector has all the components larger than or equal to at least one of the threshold vectors.

**Proposition 5** *The function  $\alpha_t(c, p, R)$  is non increasing as a function of the capacity:*

$$\alpha_t(c, p, R) \leq \alpha_t(c - e_l, p, R) \quad \forall c, t, R, p, l.$$

PROOF: See Appendix 1. ■

We are not able to extend the result that the function  $\alpha_t(c, p, R)$  is decreasing as a function of time, using the same proof that we have used for the case without reservations, because the number of pending reservations varies as a function of the current arrival. For the particular case where the probability of an arrival in the next unit of time depends on the number of pending reservations but not on  $t$ , we can obtain the result that  $\alpha_t(c, p, R)$  is decreasing as a function of time. This proof is done by induction using the hypothesis:

$$F_t(c, R, i) - F_t(c - e_p, R, i) \geq F_{t+1}(c, R, i) - F_{t+1}(c - e_p, R, i).$$

Using the assumption that the probability of an arrival in the next unit of time depends only on the number of pending reservations, we take the expected value on both sides of the induction hypothesis, and obtain the desired result. We conjecture that the previous result still holds for the case where the arrival probabilities depends on  $t$ . However, new approaches to prove this result are needed.

The optimal policy for the general problem (multiple products, several types of customers and reservations) can be found solving the dynamic programming formulation. However, this approach takes too long for most “real size problems.” The computational effort can be significantly reduced using the properties characterizing the optimal policy derived in this section, making problems with one product tractable. However, when real size problems with several types of products are considered, the computational burden becomes intractable even using this approach. Therefore, it is necessary to develop ad-hoc heuristics that provide good approximations for the optimal policy. In the next section we present a family of upper bounds that will be useful to measure the quality of these heuristics.

### 3 UPPER BOUNDS

In this section we describe a family of upper bounds for the optimization problem described Section 2, considering multiple types of rooms and reservations. These upper bounds are based on the general idea that expected optimal profits increase when the amount of information available about future demand becomes larger. For example, if at the beginning of the planning horizon the manager knows the total number of requests that will take place, she can make a better decision than the decision she would make without this information. Before presenting the model for the upper bound, we define the following additional notation:

1.  $R^k$ : vector of pending reservations at the beginning of period  $k$ .
2.  $s^k(R)$ : vector with arrivals in period  $k$  if the initial vector of reservations is  $R$ . The component  $s_i^k(R)$  is a random variable that represents the number of class  $i$  requests in period  $k$ .
3.  $B_p$ : set of customer classes that request a room type better than or equal to  $p$ .
4.  $x^k$ : vector of decision variables. The component  $x_i^k$  corresponds to the number of class  $i$  customers accepted in period  $k$ .

The following problem provides an upper bound for the problem  $\text{MPR}(C, R)$  (multiple product case with reservations).

$$\text{PROBLEM UB}_K(C^1, R^1): \quad E_{s^1(R^1)}[G_1(C^1, R^1, s^1(R^1))],$$

where:

$$\begin{aligned}
 G_k(C^k, R^k, s^k(R^k)) &= \max_{x_1^k, \dots, x_n^k} \sum_i \pi_i x_i^k - \sum_i c_i (s_i^k(R^k) - x_i^k) + \\
 &\quad E_{s^{k+1}(R^{k+1})}[G_{k+1}(C^{k+1}, R^{k+1}, s^{k+1}(R^{k+1}))] \\
 \text{s.t.} \quad x_i^k &\leq s_i^k(R^k), \quad \forall i = 1, \dots, n, & (1) \\
 \sum_{i \in B_p} x_i^k &\leq \sum_{i=1}^p C^k(i), \quad \forall p = 1, \dots, m, & (2) \\
 R^{k+1} &= R^k - f_1(s^k(R^k)), & (3) \\
 C^{k+1} &= C^k - f_2(x^k), & (4) \\
 x_i^k &\geq 0 \text{ and integer}, \quad \forall i = 1, \dots, n. & (5)
 \end{aligned}$$

The boundary condition is given by:

$$G_K(C^K, R^K, s^K(R^K)) = \max_{x_1^K, \dots, x_n^K} \sum_i \pi_i x_i^K - \sum_i c_i (s_i^K(R^K) - x_i^K)$$

$$\begin{aligned}
\text{s.t.} \quad & x_i^K \leq s_i^K(R^K), \quad \forall i = 1, \dots, n, \\
& \sum_{i \in \mathcal{B}_p} x_i^K \leq \sum_{i=1}^p C^K(i), \quad \forall p = 1, \dots, m, \\
& x_i^K \geq 0 \text{ and integer}, \quad \forall i = 1, \dots, n.
\end{aligned}$$

The maximum number of customers that can be accepted in period  $k$  is bounded by the number of requests received in that period. This condition is given by constraint (1), for each class of customers. The set of constraints (2) corresponds to the capacity constraint for each class of room, considering that they can be downgraded. Constraints (3) are conservation constraints for the number of pending reservations, i.e.,  $f_1(s^k(R^k))$  is equal to zero for all components associated to customers without reservations and equal to the corresponding coordinate of  $s^k(R^k)$  for all components associated to customers with reservations. Finally, constraints (4) are conservation constraints for the capacity. Thus, the  $j^{\text{th}}$  component of  $f_2(x^k)$  is equal to the number of type  $j$  products sold in period  $k$  considering that some of them were downgraded.

The index  $K$  of the family of upper bounds is given by the number of periods in which the planning horizon is divided. To prove that the problem  $\text{UB}_K(C, R)$  provides an upper bound for  $\text{MPR}(C, R)$  we have to apply several times the property that the maximum of the expected value is less than or equal to the expected value of the maximum. Instead of giving a formal proof, we explain the intuition underlying these upper bounds. The problem  $\text{UB}_1(C, R)$  corresponds to the extreme case where the manager has perfect information about the number of arrivals for each class of customers in the planning period. Therefore, she can make a better decision than the decision she would make with less information. In problem  $\text{UB}_K(C, R)$  we have divided the planning horizon in  $K$  periods. At the beginning of every period the manager has perfect information about the customers that show up in that period. Hence, she can make a decision better than the decision she would make knowing the current arrival and the probability of potential arrivals in the future. In the limit, when  $K$  is equal to the number of periods considered in the original problem (periods with at most one arrival), the upper bound is equal to the optimal solution of  $\text{MPR}(C, R)$ .

## 4 HEURISTICS

In this section we develop heuristics for finding “good” feasible solutions for the optimization problems formulated in Section 2. These heuristics are based on the characterization of the optimal policies studied in Section 2.

We introduce the following additional notation:

1.  $\pi_i$  = price associated to a class  $i$  customer.



2.  $c_i$  = rejection cost of a class  $i$  customer.
3.  $D_i = \{j/\pi_j + c_j > \pi_i + c_i\}$ .
4.  $\text{pref}(i)$  = product type that a class  $i$  customer requests.
5.  $c$  = vector of current capacity.
6.  $\lambda_i(t)$  = arrival rate of class  $i$  customers at time  $t$ .
7.  $T$  = \ planning horizon.
8.  $t$  = current time.

#### 4.1 Heuristic 1

This heuristic works as follows. If at time  $t$  a customer requests a room, the manager calculates the expected number of arrivals that have an opportunity cost (price plus rejection cost) larger than the current request's opportunity cost, from  $t$  to the end of the planning horizon. The current capacity is assigned optimally to this expected number of arrivals. If after the assignment there is a remaining room that satisfies the current request, then the room is rented to the customer. Otherwise, the manager rejects the request.

In order to assign optimally the rooms to the expected number of arrivals, the heuristic satisfies first the expected demand of customers with the largest opportunity cost, downgrading if it is necessary. Then it satisfies the expected demand associated to the second largest opportunity cost and so forth.

**DESCRIPTION OF HEURISTIC 1 (HEUR1):** Suppose a class  $i$  customer requests a type  $s$  room at time  $t$ . The following algorithm is used to determine whether or not to accept the customer's request:

**Step 0: Initialization**

$$\text{Cap} = c.$$

**Step 1: Determining the remaining capacity after assigning the rooms to the expected number of requests from  $t$  to  $T$  that have a larger opportunity cost than the current request's opportunity cost.**

$S_i = D_i$   
 1.1  $j = \max_{k \in S_i} \{\pi_k + c_k\}$   
 1.2  $\text{Exp\_arr} = \int_t^T \lambda_j(\tau) d\tau$   
 1.3  $p = \text{pref}(j)$   
     If  $(\text{Cap}(p) > \text{Exp\_arr})$  then  
          $\text{Cap}(p) = \text{Cap}(p) - \text{Exp\_arr}$   
          $\text{Exp\_arr} = 0$   
     else  
          $\text{Exp\_arr} = \text{Exp\_arr} - \text{Cap}(p)$   
          $\text{Cap}(p) = 0$   
     endif  
     if  $(p > 1)$  then  $p = p - 1$ , goto 1.3  
      $S_i = S_i - \{j\}$ ,  
     if  $S_i \neq \emptyset$  goto 1.1,  
     else goto Step 2.

**Step 2:** Checking if there is an available room for the class  $i$  request.

$p = s$   
 2.1 if  $(\text{Cap}(p) > 0)$  then  
     Optimal decision  $\leftarrow$  give product  $p$  to the current request.  
     go to step 3  
 endif  
 if  $(p > 1)$  then  $p = p - 1$ , goto 2.1  
 else, Optimal decision  $\leftarrow$  reject the current request.

**Step 3:** STOP.

## 4.2 Heuristic 2

This heuristic is similar to the heuristic described above. The only difference is that instead of considering the expected number of arrivals from  $t$  to the end of the planning horizon, the heuristic considers a number of arrivals such that with 90% probability the actual number of arrivals is less than it. HEUR2 is used to denote heuristic 2.

### 4.3 Heuristic 3

This heuristic is developed for the single product case. It calculates the marginal expected profit of the room that satisfies the current request. If this marginal value is larger than the certain opportunity cost associated to the current request, the manager rents the room. She rejects the request otherwise.

In order to compute the marginal expected value, we consider the customers that have a larger opportunity cost than the current request's net profit. For those customers, we calculate the probability that the number of requests is larger than or equal to the current capacity. The weighted opportunity cost for those customers times the above probability corresponds to the expected profit for this marginal room.

Description of Heuristic 3 (HEUR3): Suppose there is a class  $i$  request at time  $t$ . The following algorithm is used to determine whether or not to accept the customer's request:

**Step 0:** Initialization

$$\text{Cap} = C,$$

**Step 1:** Computing the weighted opportunity cost

$$\text{Weighted\_prof} = \sum_{j \in D_i} (\pi_j + c_j) \times \lambda_j / \sum_{j \in D_i} \lambda_j$$

**Step 2:** Computing the probability that there are more or equal number of requests than the current capacity.

$$\text{Prob}_{it} = \Pr\{\sum_{j \in D_i} \text{class } j \text{ requests in } (T - t) \geq \text{Cap} \}$$

**Step 3:**

If  $(\text{Prob}_{it} \times \text{Weighted\_prof} > \pi_i + c_i)$  then  
Optimal decision  $\leftarrow$  reject the current request  
else  
Optimal decision  $\leftarrow$  accept the current request  
Endif

**Step 4:** STOP.

### 4.4 Heuristic 4

In this heuristic the manager satisfies all the requests as long as there is an available product that satisfies the customer's need. The decisions are made without considering any additional

information. This simple heuristic is only used for the purpose of comparison with the other heuristics. HEUR4 is used to denote heuristic 4.

## 5 COMPUTATIONAL EXPERIMENTS

In this section we use Monte Carlo simulations to evaluate the performance of the heuristics described in Section 4. We assume that customers arrive according to a Poisson process, and every time a customer requests a room, the manager makes a decision following the rules given by the heuristics. We simulate the arrival process during an operational day and compute the total profit obtained under the application of the different heuristics. Averaging the profits given by repeated simulations, we obtain a statistical estimate of the expected profit.

We consider three groups of computational experiments. The first two sets of experiments correspond to relatively small applications that allow us to compute the optimal solutions. The parameters used in the third application are based on information given by the manager of a medium size hotel in Santiago, Chile. In this real application, we are not able to compute the optimal solution, therefore we compare the performance of the heuristics with the upper bounds for the general optimization problem.

A statistical estimate of the value of the upper bounds is also computed using Monte Carlo simulations. Recall that the upper bounds correspond to the expected value of an optimization problem. The expectation is taken over all possible arrival combinations and therefore it is infeasible to compute its exact value.

The criterium to stop the simulations is given by a standard deviation of less than 0.1% of the expected value. In the experiments we compute two upper bounds. The first one, UB1, corresponds to the case where all the requests are known at the beginning of the planning horizon. In the second upper bound, UB2, the planning horizon is divided in two periods; the requests during each segment are known at their beginning.

In what follows, we define the booking index,  $OB$ , to measure the hotel's capacity as a function of the expected number of requests.

Single product case:  $OB$  is equal to the total expected number of arrivals over the total capacity.

$$OB = \frac{\sum_{i=1}^n \int_{t=1}^T \lambda_i(t) dt}{C}$$

Multiple product case: first we compute the booking index for each type of product,  $OB_s$ , using the cumulative capacity and the cumulative number of arrivals. Then, we calculate  $OB$  as

the weighted booking index, where the weights correspond to the expected number of arrivals that request each type of product.

$$EA_s = \sum_{i \in A_s} \int_{t=1}^T \lambda_i(t) dt$$

$$OB_s = \frac{\sum_{j=1}^s EA_j}{\sum_{j=1}^s C(j)}$$

and finally,

$$OB = \sum_{s=1}^m OB_s \frac{EA_s}{\sum_{j=1}^m EA_j}$$

In the first group of computational experiments, we consider a single type of room, and 3 classes of customers that arrive in a Poisson manner with arrival rates equal to 5, 3, and 2 customers per hour. The prices that they pay are \$200, \$120, and \$85 respectively and there is no rejection costs. We can think on these customers as walk-ins that do not have a previous commitment with the hotel. The planning horizon is 12 hours. The performance of the heuristics is shown in table 1:

Table 1: Single Product Case

Capacity # rooms	$OB * 100$	HEUR1 w/r opt. sol.	HEUR3 w/r opt. sol.	HEUR4 w/r opt. sol.	HEUR1 w/r UB2	HEUR3 w/r UB2	HEUR4 w/r UB2
50	240	99.9%	99.9%	76.7%	99.8 %	99.8%	76.6 %
60	200	98.9%	98.4%	78.5%	98.2 %	97.7%	78.0 %
70	171	98.5%	98.1%	82.2%	97.4 %	97.0%	81.3 %
80	150	99.1%	99.0%	86.0%	98.0 %	97.8%	85.0 %
90	133	99.6%	99.6%	89.5%	98.5 %	98.5%	88.5 %
100	120	99.6%	99.6%	93.3%	98.5 %	98.6%	92.3 %
110	109	99.9%	100.0%	96.9%	99.0 %	99.1%	96.0 %

The first column in table 1 is the hotel's capacity (number of available rooms). The next three columns have the performance of the heuristics compared with the optimal solution. Finally, in the last three columns appear the performance of the heuristics with respect to the upper bound UB2.

The performance of HEUR4 (to give a room to all the requests as long as there is an available room) improves as long as the capacity increases. Naturally, if the capacity is very large, all the heuristics must lead to the optimal solution. However, for the cases where the capacity is an issue, HEUR4 performs much worse than the other heuristics. Therefore, in those cases, it is sensible to use additional information as for example the current capacity and the probability distribution of future arrivals. We also observe that HEUR1 and HEUR3 perform almost the same; for very small and large capacities the solutions given by the heuristics are near to the optimal solutions. When the capacity is small these two heuristics rent the rooms to the highest opportunity cost customers which, intuitively, corresponds to the optimal strategy. On the other hand, when the capacity is large enough the optimal strategy is to accept all the requests which is also the policy given by the heuristics. The exact behavior can be seen in Figure 2 in Appendix 2. The expected profits appear in Table 4 in Appendix 2.

In the second group of experiments we consider 2 types of rooms with the possibility of downgrading. We also consider 3 types of customers with arrival rates equal to 2, 3 and 5 customers per hour, prices equal to \$200, \$120, and \$85, and room preferences equal to 1, 2 and 2 respectively. No rejection costs are considered and the planning horizon is 12 hours.

Table 2: Multiple Product Case with Downgrading

Capacity (# rooms)	$OB * 100$	HEUR1 w/r optimal solution	HEUR2 w/r optimal solution	HEUR4 w/r optimal solution	HEUR1 w/r UB2	HEUR2 w/r UB2	HEUR4 w/r UB2
(5,30)	850	99.8%	84.1%	85.6%	99.7%	84.0%	85.5%
(10,30)	528	99.8%	84.3%	84.3%	99.7%	84.2%	84.2%
(10,45)	463	98.5%	87.1%	90.1%	98.0%	86.6%	89.6%
(10,50)	448	98.7%	90.0%	91.9%	98.1%	89.5%	91.4%
(10,60)	425	99.0%	94.5%	94.6%	98.5%	94.1%	94.1%
(15,70)	305	99.4%	96.2%	96.4%	99.0%	95.8%	96.0%
(20,80)	240	99.6%	97.9%	97.8%	99.2%	97.5%	97.4%
(20,90)	231	99.9%	99.5%	99.3%	99.7%	99.2%	99.1%
(25,95)	195	100.0%	100.0%	100.0%	99.9%	100.0%	99.9%

The first column in Table 2 is the hotel’s capacity (number of rooms). The rooms are ordered according to their quality; thus the first number in the capacity vector corresponds to the suite rooms. The next three columns correspond to the performance of HEUR1, HEUR2, and HEUR4 compared with the optimal solution. Finally, the last three columns show the performance of these three heuristics with respect to the upper bound UB2. HEUR2 and HEUR4 have a similar behavior, HEUR4 being slightly better. However, these two heuristics have, in all the cases, a worse performance than heuristic 1. HEUR2 follows highly conservative rules, because a customer is accepted only if there is at most 10% probability for renting the room to a higher opportunity cost. Therefore, it is worthwhile to apply HEUR1 because it gives better solutions to the optimization problem. We also observe from Table 2 that the performance of HEUR1 is almost optimal in both extremes; i.e. for very small and large capacities. When capacity is very small, it is always better not to downgrade rooms because better quality rooms will be rented to the highest opportunity cost customers. On the other hand, all the heuristics perform close to the optimal when the capacity increases to the limit that all the requests should be accepted. The expected profits for these experiments can be found in Appendix 2, Table 5 and the graphic behavior of heuristic 1 as a function of the capacity in Figure 3, Appendix 2.

For the last set of experiments, we use parameters based on data of a medium size hotel in Santiago, Chile. The hotel works with three main classes of customers: (i) guaranteed reservation customers that are guaranteed a room based on a previous arrangement, (ii) “6 p.m. hold” customers that are guaranteed a room based on a previous arrangement only if they arrive before 6 p.m., and (iii) “walk-ins” that request a room during the target date without a reservation. Within these categories, customers request standard and suite rooms. Therefore, the hotel faces the demand of 6 classes of customers. The average percentage of requests that the hotel receives in a day is equal to 81% of guaranteed reservation customers, 14% of 6 p.m. hold customers, and 5% of walk-ins.

The parameters used in the following simulations can be found in Appendix 2. In this case it is not feasible to compute the value of the optimal objective function, because it takes too long. We only compute the upper bounds UB1 and UB2 and the value of the objective function given by the heuristics 1, 2 and 4.

Table 3 shows the performance of HEUR1, HEUR2 and HEUR4 compared with the upper bounds given by UB1 and UB2. We observe that HEUR2 and HEUR4 have a similar behavior, HEUR2 being slightly better. Both heuristics improve their performances when the capacity increases, and in the limit when the capacity is large in comparison with the demand, they perform close to the optimal solution. However, HEUR1 has a better performance than the other heuristics

Table 3: A Real Case

Capacity (# rooms)	$OB * 100$	HEUR1 w/r UB1	HEUR2 w/r UB1	HEUR4 w/r UB1	HEUR1 w/r UB2	HEUR2 w/r UB2	HEUR4 w/r UB2
(20,130)	186	100.0%	83.9%	81.9%	100.0%	83.9%	82.0%
(20,150)	168	99.8%	87.1%	86.0%	99.9%	87.2%	86.1%
(20,190)	143	99.3%	93.3%	92.7%	99.8%	93.8%	93.2%
(30,200)	122	98.6%	95.1%	94.3%	99.2%	95.6%	94.8%
(40,210)	108	98.2%	97.0%	96.5%	98.7%	97.4%	96.9%
(40,220)	105	98.6%	98.1%	97.6%	99.0%	95.5%	98.0%
(60,220)	93	99.4%	99.3%	99.2%	99.6%	99.5%	99.4%
(80,220)	84	99.6%	99.7%	99.6%	99.8%	99.9%	99.8%

in all the cases considered. We also observe that the worst performance of HEUR1 is 98.7% compared with UB2, which is close to the optimal considering that the optimal solution is less than or equal to the upper bound.

## 6 CONCLUSIONS

This paper has studied optimal policies for renting hotel rooms to various classes of customers. We have considered stochastic and dynamic arrival of customers from different market segments (airline employees, government workers, executives, tourists, etc.), no assumptions concerning the particular order between the arrivals of different classes of customers, and multiple types of rooms with the possibility of downgrading. The limited capacity and the finite planning horizon play a central role when determining the optimal selling strategies.

We have characterized the optimal policies as follows: given a period of time, if a request is accepted for a certain capacity, then it is also accepted for any larger capacity. Furthermore, for every class of customers and every capacity vector, there exists an instant in time beyond which it is optimal to satisfy the customer's request.

Exact optimal policies can be computed for the single product case. For the general case with multiple types of rooms, we have developed ad-hoc heuristics, based on the qualitative properties of the optimal policies, to find approximations for the optimal solution. Computational results showed that their performances are close to the optimal solution for a wide variety of realistic scenarios.



The main feature of these heuristics is their simplicity which is extremely important in terms of their practicality; managers are more open to use rules that are intuitive and easy to implement.

Perhaps the most important result of this paper is the simple characterization of the optimal selling policies in a realistic framework where customers from different market segments arrive simultaneously. We also added a new dimension to this problem when considering multiple types of rooms with the possibility of downgrading; a practice widely used in the hotel industry.

## APPENDIX 1

**Lemma 1** *The function  $F_t(c, i)$  is monotonically increasing as a function of the capacity:*

$$F_t(c, i) \geq F_t(c - e_p, i), \quad \forall c, p, t, i,$$

and

$$F_t(c - e_k, i) \geq F_t(c - e_p, i), \quad \forall k \geq p, \text{ and } \forall c, t, i.$$

The single product case corresponds to the particular case where  $c$  is a scalar and  $e_p = e_k = 1$ .

**PROOF:** Suppose we have an optimal policy when we start with a capacity equal to  $c - e_p$  at time  $t$ . This policy is also feasible when we start with a capacity equal to  $c$  at time  $t$  (and gives the same value for the objective function), but it is not necessarily optimal. Therefore, the maximum expected profit from time  $t$  onwards starting with capacity  $c - e_p$  is less than or equal to the maximum expected profit starting with capacity  $c$ . Hence,

$$F_t(c, i) \geq F_t(c - e_p, i), \quad \forall c, p, t, i.$$

The same argument can be used when we replace  $c$  by  $c - e_k$  with  $k \geq p$ , because product  $p$  can be downgraded to be used as product  $k$ . ■

**Lemma 2** *The objective function satisfies the following inequality:*

$$F_t(c + e_l + e_p, i) - F_t(c + e_l, i) \leq F_t(c + e_p, i) - F_t(c, i), \quad \forall t, i, c, p, l. \quad (6)$$

**PROOF:** In what follows we use induction in  $t$  to prove the lemma for the single product case. The objective function for the last period in the planning horizon is given by:

$$F_T(c, i) = \begin{cases} \pi_i & \text{if } c > 0 \text{ and } i \neq 0, \\ -c_i & \text{if } c = 0 \text{ and } i \neq 0, \\ 0 & \text{if } i = 0. \end{cases}$$

We observe that given  $i$ ,  $F_T(c, i)$  trivially satisfies the inequality (6), i.e.,

$$F_T(c + 2, i) - F_T(c + 1, i) \leq F_T(c + 1, i) - F_T(c, i) \quad \forall c.$$

In the induction hypothesis we assume that the inequality (6) is satisfied at period  $t + 1$  and we prove that it is also satisfied at period  $t$ . The objective functions at period  $t$  are given by (assuming that  $i \neq 0$  and  $c > 0$ ):

$$F_t(c, i) = \begin{cases} \pi_i + E_j[F_{t+1}(c - 1, j)] & (a), \\ -c_i + E_j[F_{t+1}(c, j)] & (b). \end{cases}$$

$$F_t(c + 1, i) = \begin{cases} \pi_i + E_j[F_{t+1}(c, j)], & (c), \\ -c_i + E_j[F_{t+1}(c + 1, j)] & (d). \end{cases}$$

$$F_t(c + 2, i) = \begin{cases} \pi_i + E_j[F_{t+1}(c + 1, j)] & (e), \\ -c_i + E_j[F_{t+1}(c + 2, j)] & (f). \end{cases}$$

In order to prove property (6) we have to consider all possible values for the functions above. Thus, we have to evaluate the 8 combinations.

1. Combination (a) + (c) + (e): Replacing the objective functions for their corresponding values in inequality (6), we get,

$$\pi_i + E_j[F_{t+1}(c + 1, j)] - \pi_i - E_j[F_{t+1}(c, j)] \leq \pi_i + E_j[F_{t+1}(c, j)] - \pi_i - E_j[F_{t+1}(c - 1, j)],$$

which is true by the induction hypothesis. The same proof holds for combinations (b) + (c) + (e), (b) + (d) + (e), and (b) + (d) + (f).

Combination (a) + (d) + (f): In this case, when the capacity is  $c$  we accept the request, i.e.,

$$c_i + \pi_i \geq E_j[F_{t+1}(c, j)] + E_j[F_{t+1}(c - 1, j)],$$

and when the capacity is  $c + 1$ , we reject the request, i.e.,

$$c_i + \pi_i \leq E_j[F_{t+1}(c + 1, j)] + E_j[F_{t+1}(c, j)].$$

Both inequalities together lead to:

$$E_j[F_{t+1}(c + 1, j)] + E_j[F_{t+1}(c, j)] \geq E_j[F_{t+1}(c, j)] + E_j[F_{t+1}(c - 1, j)],$$

which contradicts the induction hypothesis. Hence, the combination (a) + (d) + (f) is infeasible. Combinations (b) + (c) + (e), (b) + (d) + (e), and (b) + (d) + (f) are also infeasible. Therefore,

the inequality (6) holds for period  $t$ . When  $c = 0$  the same proof holds. The case when  $i = 0$  the inequality (6) holds trivially. ■

A similar proof holds for the case with types of rooms. We believe that the same proof can be generalized for the multiple product case with more than two types of rooms. This lemma says that the smaller the current inventory, the larger the profit associated to an extra room. The intuition behind this lemma is that the opportunity cost of an extra room decreases when the capacity increases because the stochastic process that determines the total demand does not change.

**Lemma 3** *The function  $F_t(c, R, i)$  is monotonically increasing as the function of the capacity and it satisfies the property given in Lemma 2.*

PROOF: The proof is an extension of the multiple product case without reservations. ■

### Proof of Proposition 3

In what follows we present the general proof for the multiple product case. By Lemma 2, we know that the following inequality holds:

$$F_{t+1}(c - e_l, j) - F_{t+1}(c - e_p - e_l, j) \geq F_{t+1}(c, j) - F_{t+1}(c - e_p, j) \quad \forall t, i, c, p, l.$$

The probability of a class  $i$  arrival in the next unit of time depends only on  $t$ , hence we take the expected value in both sides and the inequality still holds:

$$E_j[F_{t+1}(c - e_l, j) - F_{t+1}(c - e_l - e_p, j)] \geq E_j[F_{t+1}(c, j) - F_{t+1}(c - e_p, j)].$$

Thus,

$$\alpha_t(c - e_l, p) \geq \alpha_t(c, p). \quad \blacksquare$$

### Proof of Proposition 4

Assume that a type  $l$  room is sold to a class  $i$  customer when the capacity is equal to  $c$ . Assuming that  $l \neq p$ , the same room  $l$  is available for a class  $i$  customer when the initial capacity is  $c - e_p$ , then:

$$\begin{aligned} F_{t+1}(c, i) - F_{t+1}(c - e_p, i) &= \max\{\pi_i + E_j[F_{t+2}(c - e_l, j)], -c_i + E_j[F_{t+2}(c, j)]\} \\ &\quad - \max\{\pi_i + E_j[F_{t+2}(c - e_p - e_l, j)], -c_i + E_j[F_{t+2}(c - e_p, j)]\} \end{aligned}$$

We analyze the following two cases:

Case 1:

$$\max\{\pi_i + E_j[F_{t+2}(c - e_p - e_l, j)]; -c_i + E_j[F_{t+2}(c - e_p, j)]\} = -c_i + E_j[F_{t+2}(c - e_p, j)].$$

In this case,

$$\begin{aligned} F_{t+1}(c, i) - F_{t+1}(c - e_p, i) &\geq -c_i + E_j[F_{t+2}(c, j)] - (-c_i + E_j[F_{t+2}(c - e_p, j)]), \\ F_{t+1}(c, i) - F_{t+1}(c - e_p, i) &\geq E_j[F_{t+2}(c, j)] - E_j[F_{t+2}(c - e_p, j)] = \alpha_{t+1}(c, p) \\ F_{t+1}(c, i) - F_{t+1}(c - e_p, i) &\geq \alpha_{t+1}(c, p). \end{aligned} \tag{7}$$

Case 2:

$$\max\{\pi_i + E_j[F_{t+2}(c - e_p - e_l, j)]; -c_i + E_j[F_{t+2}(c - e_p, j)]\} = \pi_i + E_j[F_{t+2}(c - e_p - e_l, j)].$$

In this case,

$$\begin{aligned} F_{t+1}(c, i) - F_{t+1}(c - e_p, i) &\geq \pi_i + E_j[F_{t+2}(c - e_l, j)] - (\pi_i + E_j[F_{t+2}(c - e_p - e_l, j)]), \\ F_{t+1}(c, i) - F_{t+1}(c - e_p, i) &\geq E_j[F_{t+2}(c - e_l, j)] - E_j[F_{t+2}(c - e_p - e_l, j)] = \alpha_{t+1}(c - e_l, p) \\ F_{t+1}(c, i) - F_{t+1}(c - e_p, i) &\geq \alpha_{t+1}(c - e_l, p). \end{aligned} \tag{8}$$

Using Proposition 3, we obtain:

$$F_{t+1}(c, i) - F_{t+1}(c - e_p, i) \geq \alpha_{t+1}(c - e_l, p) \geq \alpha_{t+1}(c, p).$$

Equations (7) and (8) yield:

$$F_{t+1}(c, i) - F_{t+1}(c - e_p, i) \geq \alpha_{t+1}(c, p).$$

For  $i = 0$ , the previous result holds trivially. Therefore, using the fact that the probability of a new arrival depends only on  $t$ , we take expected value on both sides of the inequality above, obtaining the desired result:

$$\begin{aligned} E_i[F_{t+1}(c, i) - F_{t+1}(c - e_p, i)] &\geq \alpha_{t+1}(c, p), \\ \alpha_t(c, p) &\geq \alpha_{t+1}(c, p). \end{aligned}$$

For the special case when  $p = l$ , we have two cases:

1. If the initial capacity  $c - e_p$  has an available type  $p$  room, then the proof above holds without any change.

2. If the initial capacity  $c - e_p$  does not have any type  $p$  room available, then, if the current request is accepted the customer receives a type  $k$  room. This room is better than a type  $p$  room, i.e.,  $k < p$ . In this case, the proof changes slightly, so we present the part of the proof that differs from the previous case. By definition we have:

$$F_{t+1}(c, i) - F_{t+1}(c - e_p, i) = \max\{\pi_i + E_j[F_{t+2}(c - e_p, j)]; -c_i + E_j[F_{t+2}(c, j)]\} \\ - \max\{\pi_i + E_j[F_{t+2}(c - e_p - e_k, j)]; -c_i + E_j[F_{t+2}(c - e_p, j)]\}.$$

Similarly to the previous proof, we have to analyze two cases. The first one is analogous to the Case 1 above. The second case changes to the following:

$$\max\{\pi_i + E_j[F_{t+2}(c - e_p - e_k, j)]; -c_i + E_j[F_{t+2}(c - e_p, j)]\} = \pi_i + E_j[F_{t+2}(c - e_p - e_k, j)].$$

Hence,

$$F_{t+1}(c, i) - F_{t+1}(c - e_p, i) \geq \pi_i + E_j[F_{t+2}(c - e_p, j)] - (\pi_i + E_j[F_{t+2}(c - e_p - e_k, j)]),$$

$$F_{t+1}(c, i) - F_{t+1}(c - e_p, i) \geq E_j[F_{t+2}(c - e_p, j)] - E_j[F_{t+2}(c - e_p - e_k, j)] = \alpha_{t+1}(c - e_p, k).$$

Using Proposition 3, we obtain:

$$F_{t+1}(c, i) - F_{t+1}(c - e_p, i) \geq \alpha_{t+1}(c - e_p, k) \geq \alpha_{t+1}(c, k).$$

Finally, using that the function  $F_{t+1}(c)$  is increasing as a function of the capacity, we have:

$$\alpha_{t+1}(c, k) \geq \alpha_{t+1}(c, p) \quad \text{if } k < p.$$

Therefore,

$$F_{t+1}(c, i) - F_{t+1}(c - e_p, i) \geq \alpha_{t+1}(c, p).$$

The rest of the proof follows the same as in the previous case. ■

## Proof of Proposition 5

Considering that the probability of a new arrival during the next time interval of length  $\Delta t$  depends only on  $t$  and on the number of pending reservations, we apply the expected value on both sides of the inequality given in Lemma 3 to obtain the desired result. ■

## APPENDIX 2

Table 4 shows the results obtained for a single product case and 3 classes of customers with arrival rates equal to 5, 3, and 2 customers per hour. The room prices are \$200, \$120, and \$85 respectively and there is no rejection costs. The planning horizon is 12 hours.

Table 4: Expected Profits for the Single Product Case.

<i>(# rooms)</i>	<i>Opt. Sol.</i>	UB1	UB2	HEUR1	HEUR3	HEUR4
50	9964	9975	9974	9956	9950	7644
60	11682	11764	11757	11550	11489	9171
70	13021	13182	13170	12821	12776	10703
80	14226	14407	14389	14103	14078	12228
90	15378	15563	15542	15309	15312	13760
100	16372	16593	16549	16299	16309	15274
110	17221	17433	17381	17200	17219	16689

Table 5 contains the expected profits for the case of 2 types of rooms, 3 types of customers with arrival rates equal to 2, 3, and 5 customers per hour, prices equal to \$200, \$120, and \$85, and room preferences equal to 1, 2, and 2 respectively. No rejection costs are considered and the planning horizon is 12 hours.

Table 5: Expected Profits for the Multiple Product Case.

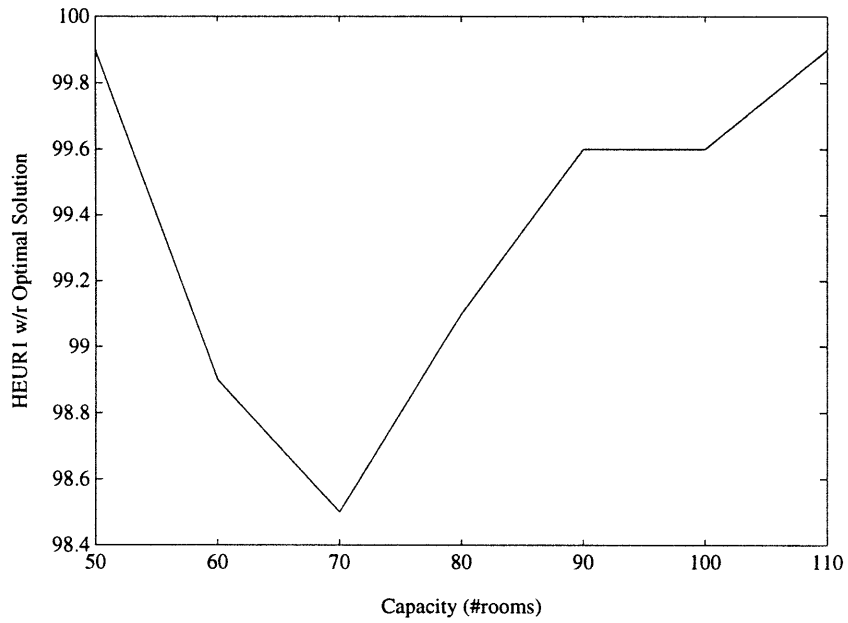
<i>(# rooms)</i>	<i>Opt. Sol.</i>	UB1	UB2	HEUR1	HEUR2	HUER4
( 5,30)	4578	4583	4583	4569	3850	3919
(10,30)	5578	5583	5585	5568	4593	4704
(10,45)	7036	7078	7075	6933	6127	6339
(10,50)	7463	7509	7506	7366	6717	6857
(10,60)	8314	8360	8357	8233	7860	7867
(15,70)	10158	10204	10202	10095	9775	9794
(20,80)	11934	11982	11977	11880	11679	11667
(20,90)	12672	12705	12697	12664	12603	12588
(25,95)	13615	13652	13625	13613	13623	13614

Table 6 contains the expected profits for a real application. The planning horizon is a operational day that starts at 8 a.m. and finishes at 12 p.m. The parameters correspond to:

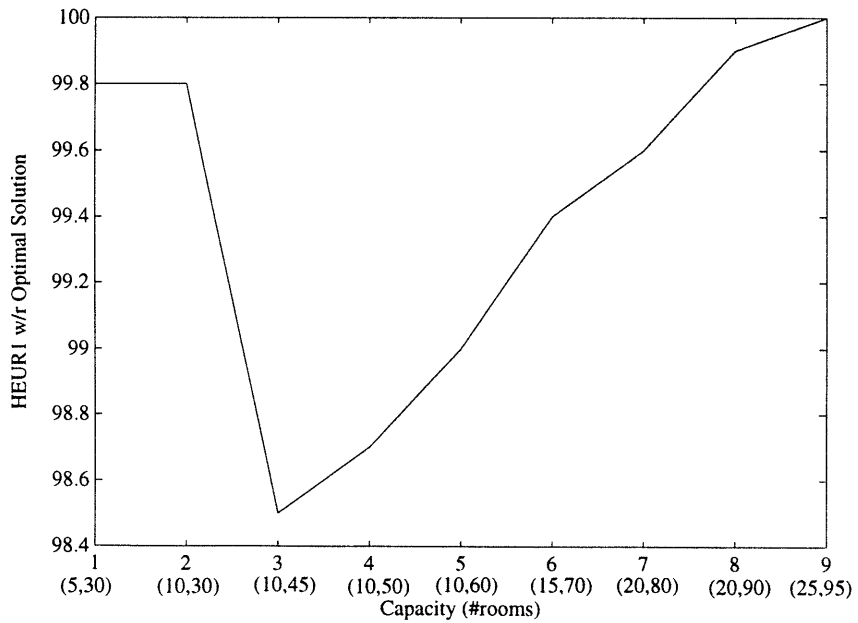
1. Number of customers : 6
2. Number of products : 2
3. Prices : (190, 200, 250, 70, 80, 110)
4. Rejection costs : (190, 100, 0 , 70, 40, 0)
5. Room preferences : (1, 1, 1, 2, 2, 2)
6. periods in the planning horizon : 8-10,10-12,12-16,16-18,18-20, 20-21,21-24
7. Arrival rate (guarantee) : (3.1, 53.9, 3.1, 3.1, 3.1, 53.9, 3.1) customers/hour.
8. Arrival rate (6 p.m. ) : (.46, 16.7, .46, .46, 0, 0, 0) customers/hour.
9. Arrival rate (walk-ins) : (.32, .32, .32, 3.6, 3.6, 3.6, 3.6) customers/hour.

Table 6: Expected Profit for a Real Case.

(# rooms)	UB1	UB2	HEUR1	HEUR2	HEUR4
(20,130)	5924	5923	5925	4970	4854
(20,150)	8740	8728	8722	7609	7512
(20,190)	13889	13813	13788	12953	12876
(30,200)	18725	18618	18465	17804	17656
(40,210)	22701	22601	22300	22014	21902
(40,220)	23548	23463	23228	23099	22990
(60,220)	25544	25491	25390	25374	25349
(80,220)	25849	25789	25747	25770	25744



**Figure 2: Performance of HEUR1 for the single product case**



**Figure 3: Performance of HEUR1 for the multiple product case**



## REFERENCES

1. ALSTRUP, J., BOAS, S., MADSEN, O., VIDAL, R., (1986): "Booking policy for flights with two types of passengers", *European Journal of Operational Research* **27**, 274-288.
2. American Airlines Annual Report, (1987). The Art of Managing Yield, 22-25.
3. BELOBABA, P.P., (1989): "Application of a probabilistic decision model to airline seat inventory control", *Operations Research* **37**, 183-197.
4. BITRAN, G., GILBERT, S. and LEONG, T-Y. (1992): "Hotel Sales and Reservations Planning", *The Service Productivity and Quality Challenge*.
5. BITRAN, G. and GILBERT, S. (1992): "Managing Hotel Reservations with Uncertain Arrivals", *Working Paper, Case Western Reserve University, Weatherhead School of Management, Department of Operations Research*.
6. BRUMELLE, S.L., MCGILL, J.I., OUM, T.H., SAWAKI, K. and TRETHERWAY, M.W. (1990): "Allocation of airline seats between stochastically dependent demands", *Transportation Science* **24**, No. 3, 183-192.
7. GILBERT, S.M. (1991): "Management of Co-production Processes with Random Yields: Applications in Manufacturing and Services." *Thesis to obtain the Ph.d. degree in Management, MIT*.
8. LADANY, S. (1976): "Dynamic Operating Rules for Motel Reservations", *Decision Sciences*, **7**, 829-840.
9. LITTLEWOOD, K. (1972): "Forecasting and Control of Passenger Bookings." *AGIFORS Symp. Proc.* **12**, 95-117.
10. MERLISS, P.P. and LOVELOCK C., (1980): "The Parker House: Sales and Reservations Planning." Harvard Business School case 9-580-152.
11. ROTHSTEIN, M. (1974): "Hotel Overbooking as a Markovian Sequential Decision Process." *Decision Sciences*, **5**, 389-404.
12. WEATHERFORD, L. and BODILY, S. (1992): "A Taxonomy and Research Overview of Perishable-Asset Revenue Management: Yield Management, Overbooking, and Pricing." *Operations Research*, Vol. 40, **5**, 831-844.