RESEARCH PAPERS

Models and techniques for hotel revenue management using a rolling horizon

This paper is dedicated to the loving memory of Richard Freling

Paul Goldman, Richard Freling, Kevin Pak* and Nanda Piersma
Received (in revised form): 7th July, 2002

*Econometric Institute, Erasmus University Rotterdam, PO Box 1738,
3000 DR Rotterdam, The Netherlands
Tel: +31 10 4088 931; E-mail: kpak@few.eur.nl

Paul Goldman *studied econometrics and operations research at the Erasmus University Rotterdam and wrote his Masters thesis about hotel revenue management. Since 2000, he has been working as a consultant at ORTEC Consultants, where revenue management is one of the key businesses.*

Richard Freling *obtained his PhD from the Erasmus University Rotterdam in 1997. He worked for some years for a software corporation before returning to the university as Associate Professor in 1999. He was published in journals such as* Transportation Science *and the* European Journal of Operational Research*. It is with deep sadness that we have to announce that Richard passed away on 29th January, 2002, at the age of 34. This paper constitutes joint work that was finished just before he became very ill.*

Kevin Pak *studied econometrics and operations research at the Erasmus University Rotterdam. In 2000, he became a PhD student with revenue management as his research topic.*

Nanda Piersma *obtained her PhD from the University of Amsterdam in 1993 and has been affiliated to the Erasmus University Rotterdam since then as Associate Professor in Operations Research. She has published in operations research and computer science journals such as* European Journal of Operational Research, Journal of Combinatorial Optimization *and* Computers and Operations Research*. In 2002, she left academia to become programme manager at the business school HES Amsterdam.*

ABSTRACT
KEYWORDS: revenue management, yield management, mathematical programming

This paper studies decision rules for accepting reservations for stays in a hotel based on deterministic and stochastic mathematical programming techniques. Booking control strategies are constructed, which include ideas for nesting, booking limits and bid prices. Multiple day stays are taken into account. Instead of optimising a decision period consisting of a fixed set of target booking days, this study simultaneously

Journal of Revenue and Pricing Management, Vol. 1, No. 3, 2002, pp. 207–219

optimises the complete range of target booking dates that are open for booking at the moment of optimisation. This yields a rolling horizon of overlapping decision periods, which will conveniently capture the effects of overlapping stays.

INTRODUCTION

Hotels offer the same rooms to different types of guests. While hotel managers would like to fill their hotels with highly profitable guests as much as possible, it is generally also necessary to allow for less profitable guests in order to prevent rooms from remaining vacant. The number of low-profit guests in the hotel should, however, be managed carefully such that all high-profit guests can still be allocated. An important decision to be made is whether to accept a booking request and generate revenue now, or to reject it in anticipation of a more profitable booking request in the future. Because this decision must be made at the time of the booking request and future demand is never certain, the booking control problem contains both dynamic and stochastic elements. Finding the right combination of guests in the hotel such that revenues are maximised is the topic of revenue management.

Revenue management originates from the airline industry, where the seats on a plane can be sold to different types of passengers. In comparison with this problem, hotel revenue management has the distinct feature that booking requests can occur for different lengths of stay and can therefore overlap. Most models for hotel revenue management consider a fixed set of target booking days over which to maximise revenues. In general, such a fixed set of days cannot be determined without missing some of the effects of the overlapping stays. This paper studies booking control policies based on a rolling horizon of decision periods. For each optimisation, all types of stays that span the current decision period are considered. Because of the rolling nature of the decision periods, eventually no overlap between the stays will be left out.

The booking control policies studied in this paper include nested booking limits and bid price methods. A deterministic as well as a stochastic model are used to derive the booking control policies. Every guest is assumed to have a strict preference for a specific type of stay. This means that, whenever a booking request is rejected, it is lost forever and is not turned into a booking request for another type of stay. Further, batch bookings or cancellations and no-shows are not considered.

The organisation of this paper is as follows: the next section gives a short overview of the related literature on hotel revenue management. The deterministic and stochastic mathematical programming models are presented in the third section. Booking control policies based on the mathematical programming models and their application over a rolling horizon are presented in the fourth section. The fifth section sketches the environment of a test case. This environment is used as a basis to simulate arrival processes by which the performances of the different booking control policies are studied. The results of the simulation studies are presented in the sixth section.

LITERATURE

Hotel revenue management has received attention in a number of papers. Bitran and Mondschein (1995) and Bitran and Gilbert (1996) concentrate on the room allocation problem at the targeted booking day itself. The hotel manager has to decide whether or not to accept a guest that requests a room on the target day, taking into account the number of reservations made and the potential number of guests who will show up without reservations (walk-ins). They formulate this problem as a stochastic and dynamic programming model. Bitran and Gilbert also provide three simple heuristics to construct booking control policies that can be used during the booking period.

Weatherford (1995) concentrates completely on the booking control problem. He proposes a heuristic which is called the nested by deterministic model shadow prices (NDSP) method. He formulates a mathematical programming model to obtain booking limits, ie the number of rooms to reserve for each type of guest. These booking limits are nested such that a guest can always make use of the capacity reserved for any less profitable guest. A possible drawback of the model is that it considers demand to be deterministic. Weatherford allows for multiple day stays and maximises the model for a decision period consisting of a fixed set of target booking days. He does not account for overlapping stays outside the decision period. Nevertheless, he shows that taking into account multiple-day stays produces better results than when only single-day stays are considered.

Baker and Collier (1999) compare the performances of five booking control policies: Two simple threshold approaches, Weatherford's NDSP method, an NDSP method that includes overbooking, and a bid price method based on work by Williamson (1992) for the airline industry. Baker and Collier compare the performances of these solution techniques under 36 hotel operating environments by ways of simulation and advise on the best heuristic for each operating environment.

This paper concentrates on the booking control problem, which makes this work comparable with the work of Weatherford (1995) and Baker and Collier (1999). Unlike these previous researches, this work uses the booking control policies over a rolling horizon of decision periods, such that all overlap between the different types of stay can be accounted for. Also, next to the well-known deterministic model, a second mathematical programming model that accounts for the stochastic nature of demand is introduced. Both nested booking limits and bid price control policies are considered. Following Baker and Collier (1999), this paper compares the performances of the different methods by simulation.

MATHEMATICAL FORMULATIONS

This section presents two mathematical programming models for finding the optimal allocation of the rooms over the different types of guests. These models are the deterministic and the stochastic model. The models are defined for use over a fixed decision period. Booking control policies based on the models and the application of these policies over a rolling horizon of decision periods are discussed in the next section.

Deterministic model

The deterministic model considered in this paper is the same as the one Weatherford (1995) uses for his NDSP method. This model replaces demand for each type of stay by an estimation and obtains the optimal allocation of the rooms over the expected demand, ie it treats demand as if it were deterministic and equal to its expectation. To formulate the deterministic model, define a stay in the hotel by (a,L,k), where a is the first night of the stay, L the length of the stay and k the price class. Further, denote the set of stays that make use of night l by N_l, where $N_l, = \{(a,L,k) : a \leqslant l \leqslant a+L1\}$. The deterministic model is then formulated as follows:

maximise

$$\sum_{a,L,k} R_{a,L,k} X_{a,L,k} \qquad (1)$$

subject to

$$\sum_{a,L,k \in N_l} X_{a,L,k} \leq C_l \ \forall l$$

$$X_{a,L,k} \leq d_{a,L,k} \;\; \forall a, L, k$$

$$X_{a,L,k} \geq 0 \;\; \text{integer} \;\; \forall a, L, k$$

where: $X_{a,L,k}$ is the number of rooms allocated to a stay of type (a,L,k), $R_{a,L,k}$ is the revenue obtained from a stay of type (a,L,k), $d_{a,L,k}$ is the expected demand for a stay of type (a,L,k), and C_l, is the capacity (number of rooms) of the hotel available on night l.

The objective of the model is to maximise revenues under the restriction that the total number of reservations for a night does not exceed the capacity of the night. In order to prevent vacant rooms, the number of rooms allocated to each type of stay is restricted by the level of the demand, which in this model is replaced by its expectation.

Although no proof exists that the constraint matrix is totally unimodular, the authors' experience and previous experiences (see Williamson, 1992; De Boer *et al.*, 2002) with the LP relaxation of this model show that, when demand is integer, the LP solutions are often integer. It can be expected that when the LP relaxation produces a fractional solution, it will not take much effort to produce an integer solution by applying branch-and-bound techniques.

Stochastic model

The determimstic model never allocates more rooms to a type of stay than the hotel expects to book for that type of stay. Because demand can deviate from its expectation, however, it can be more profitable to allocate more rooms to the more expensive types of stay. In order to consider this, the stochastic nature of demand has to be taken into account. A stochastic model is presented here which was first introduced by De Boer *et al.* (2002) for the airline industry. For this model, it is assumed that the demand for a type of stay, $D_{a,L,k}$, can take on a limited number of different realisations, which are denoted by $d_{a,L,k,1} < d_{a,L,k,2} < \ldots < d_{a,L,k,N}$. The stochastic model is now formulated as follows:

maximise

$$\sum_{a,L,k} \sum_{j=1}^{N} R_{a,L,k} \Pr(D_{a,L,k} \geq d_{a,L,k,j}) X_{a,L,k,j} \qquad (2)$$

subject to

$$\sum_{a,L,k \in N_l} \sum_{j=1}^{N} X_{a,L,k,j} \leq C_l \;\; \forall l$$

$$X_{a,L,k,1} \leq d_{a,L,k,1} \;\; \forall a, L, k$$

$$X_{a,L,k,j} \leq d_{a,L,k,j} - d_{a,L,k,j-1} \;\; \forall a, L, k \text{and} j = 2, 3, \ldots, N$$

$$X_{a,L,k,j} \geq 0 \;\; \text{integer} \;\; \forall a, L, k \text{and} j = 1, 2, \ldots, N$$

The decision variables, $X_{a,L,k,j}$, each represent the part of the demand that falls in the interval $(d_{a,L,k,j-1}, \; d_{a,L,k,j})$. Notice that $X_{a,L,k,j}$ will only be non-zero when $X_{a,L,k,j-1}$ has reached its upperbound of $d_{a,L,k,j-1}$, since $\Pr(X_{a,L,k} \geqslant d_{a,L,k,j-1}) \geqslant \Pr(X_{a,L,k} \geqslant d_{a,L,k,j})$. Summing the decision variables, $X_{a,L,k,j}$ over all j yields the total number of rooms allocated to the stays of type (a,L,k). As for the deterministic model, the LP relaxation of the stochastic model is solved.

The deterministic model can be obtained from the stochastic model by considering only one demand scenario. The EMR model introduced by Wollmer (1986) for the airline industry can be obtained by considering all possible demand scenarios. De Boer *et al.* (2002) show, however, that for the airline industry, three or four demand scenarios suffice to capture most of the extra revenue generated by considering the stochastic nature of the demand. A scenario is said to occur whenever the demand exceeds the level of the demand of the scenario. For each scenario j defined for type of stay (a,L,k), the level of the demand, ie $d_{a,L,k,j}$, and the probability that the scenario occurs, ie $\Pr(D_{a,L,k} \geqslant d_{a,L,k,j})$, have to be determined. The remainder of this paper will denote the scenario probabilities by $p_{a,L,k,j}$. Note that these probabilities are not mutually exclusive and therefore do not sum to 1. In fact, $p_{a,L,k,1}$ encompasses $p_{a,L,k,2}$ and $p_{a,L,k,2}$ encompasses $p_{a,L,k,3}$ and so on, such that $p_{a,L,k,1} \geqslant p_{a,L,k,2} \geqslant \ldots \geqslant p_{a,L,k,N}$.

BOOKING CONTROL POLICIES

This section discusses booking control policies based on the models presented in the previous section. Nested booking limits and bid price control policies are constructed. Further, it is discussed how these booking control policies can be used over a rolling horizon of decision period.

Nested booking limits

The number of rooms allocated to each type of stay by the models from the previous section can easily be interpreted as booking limits. These limits can be used as the maximum number of booking requests to accept for each type of stay during the booking period. It is never optimal, however, to reject a booking request when there are still rooms available for other less profitable types of stay, even if its own booking limit has been reached. Therefore, each type of stay should be allowed to tap into the rooms allocated to any less profitable type of stay. This is called nesting. In order to form nested booking limits, the different types of stay need to be ranked by their contribution to the overall revenue of the hotel. When such a ranking is determined, a nested booking limit for a type of stay can be set equal to the sum of the number of rooms allocated to that and every other, lower ranked type of stay.

It is not trivial what measurement to use to determine a nesting order of the different types of stay. Using the price class does not take into account the length of the stay. Such a measurement will rank guests who are willing to pay more for one night above guests who are willing to pay a little less for multiple nights, whereas the overall revenue generated by the multiple-night stay will most likely be higher. Nesting by the complete revenue generated by the stay does take into account the length of the stay. But this measurement does not account for the load factors of the different nights. Certain nights can be very busy and always fully booked, whereas other nights can be mainly vacant. A stay that occupies many busy nights should be valued differently from a stay that uses mainly nights with a lot of vacant rooms. One way to take into account all these aspects is to use the shadow prices obtained from the underlying allocation model. The shadow price corresponding to the capacity restriction for a night reflects the expected gain that can be obtained if one additional room were available on that night. It can be interpreted as the value of a room. Adding the shadow prices of all nights used by a stay gives an indication of the opportunity costs of the stay. A measurement for nesting is then obtained by subtracting these opportunity costs from the revenue generated by the stay. Thus, a nesting order is based on

$$\bar{R}_{a,L,k} = R_{a,L,k} - \sum_{a,L,k \in N_l} s_l \qquad (3)$$

where s_l denotes the shadow price of the capacity constraint for night l. Nested booking limits can now be constructed easily.

Bid prices

The second type of booking control policy studied in this paper is the bid price policy. This method directly links the opportunity costs of a stay to the acceptance/rejection decision. Bid prices are constructed for every night to reflect the opportunity costs of renting a room on that night. As before, the bid price of a night is estimated by the shadow price of the capacity constraint corresponding to that night. A booking request is only accepted if the revenue it generates is above the sum of the bid prices of the nights it uses, thus, if its revenue is more than its opportunity costs.

Rolling horizon

The mathematical programming models presented earlier in this paper provide an allocation of the rooms for a fixed decision period. They will be used, however, over a rolling horizon of decision periods. Assume that booking requests cannot be made more than F days in advance, and that the longest possible stay in the hotel consists of M days. The stays corresponding to the booking requests that come in at day t can then start at day t at the earliest and at day $t + F$ at the latest. The latest possible booking request will end at day $t + F + M$. Therefore, if a booking control policy is determined at day t, the decision period considered, is given by the time interval $[t, t + F + M]$. Within this decision period, all overlap between the different types of stay are taken into account, except for the overlap at the end of the interval corresponding to the stays that fall partly outside the decision period. But only the types of stay for which booking has just opened fall into this category. It can be expected that the total level of booking requests for these types of stay will not yet be such that booking requests will have to be rejected. By the time critical decisions have to be made for these types of stay, the decision period will have rolled forward and captured all overlap for these types of stay.

The booking control policy is constructed at different points in time. Every time a new policy is constructed, the decision period rolls forward. The booking limits and bid prices for the types of stay already open for booking are adjusted, and new booking limits and bid prices for the types of stay that have just opened up for booking are added.

TEST CASE

The performances of the different booking control policies are tested by way of simulation. This section discusses the simulation environment which is chosen such that it reflects the situation described by a hotel in The Netherlands. A hotel with a total capacity of 150 identical rooms is considered. These rooms can be rented out in ten different price classes, described in Table 1. The maximal length of a stay is considered to be seven days and a booking request can come in at most 90 days in advance. Overbooking is not allowed and cancellations, no-shows or group bookings are not considered. Further, it is assumed that the demand for the different price classes is independent and unique. This means that a rejected booking request is lost forever and cannot be recaptured in another price class. The arrivals of booking requests are simulated by a non-homogeneous Poisson process with intensities dependent on the price class, the starting day of the stay (eg Monday, Tuesday, ...) and the time until the target booking day. Different booking patterns are modelled in order to allow for

Table 1: Price classes

	Class	*Price*[a]
1	Tourist Rate Tours & Groups	\$50
2	Tourist Rate Low Budget	\$75
3	Tourist Rate Packages	\$110
4	Tourist Rate Medium Budget	\$120
5	Rack Rate	\$250
6	Corporate Rate, liaison corporation	\$75
7	Corporate Rate, management	\$125
8	Corporate Rate, salesperson	\$100
9	Corporate Rate, MCI	\$175
10	Corporate Rate, other	\$150

[a]Originally, all revenues in this research were measured in Dutch guilders. For sake of simplicity, this currency is substituted for the US dollar on a one-to-one rate

the different price classes to account for low tourist classes to book early in the booking process and high corporate classes to book at the end of the booking process among others. Further, some days, eg Friday, are allowed to be more busy than other days, eg Thursday. In order to let the arrival intensities fluctuate over time, the booking period is divided into ten smaller periods of nine days, each with a constant arrival intensity. As in Baker and Collier (1999) and Bitran and Mondschein (1995), the length of the stay is not considered to influence the arrival intensity. Instead, the length of the stay of each arrival is modelled by a logistic distribution with a parameter dependent on the price class and the starting day of the stay. The arrival intensities and the parameters for the logistic distribution are chosen to reflect a busy period in the hotel in which, on average, the total demand exceeds the capacity of the hotel. This is the situation in which revenue management produces the highest gains in revenue. The average level of the demand per day expressed as a percentage of the capacity is presented in Figure 1. Note that one booking request can generate demand for multiple days. Figure 1 shows that, in this test case, Friday is the busiest day of the week with an average demand which exceeds capacity by more than 60 per cent. On Thursday, the average demand is equal to the capacity of the hotel, which makes it the least busy day of the week. The exact arrival intensities and parameters for the logistic distribution used for the simulation will be made available to the interested reader upon request.

The performances of the different booking control policies are compared over a six-week period. Because the hotel is empty at the beginning of the simulation, a start-up period is used to fill up the hotel such that the overlap of the stays already in the hotel at the beginning of the six-week period can be considered. To make sure that no stay that could have arrived before the start-up period will overlap with any stay considered for the evaluation, the start-up period is chosen to consist of two weeks. Likewise, a cool-down period of two weeks is also used. The first day of the start-up period is denoted by $t = 1$.

Figure 1: Average demand per day as percentage of capacity

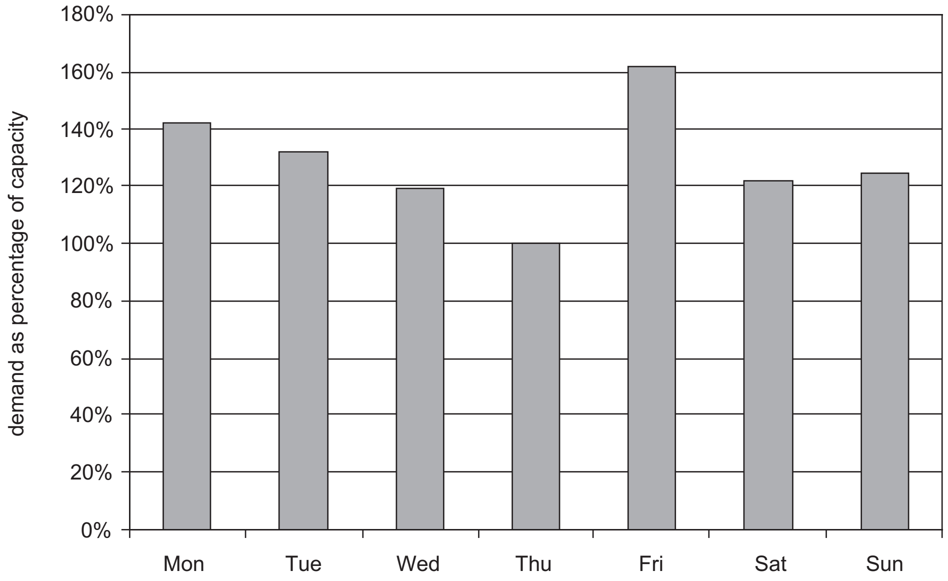
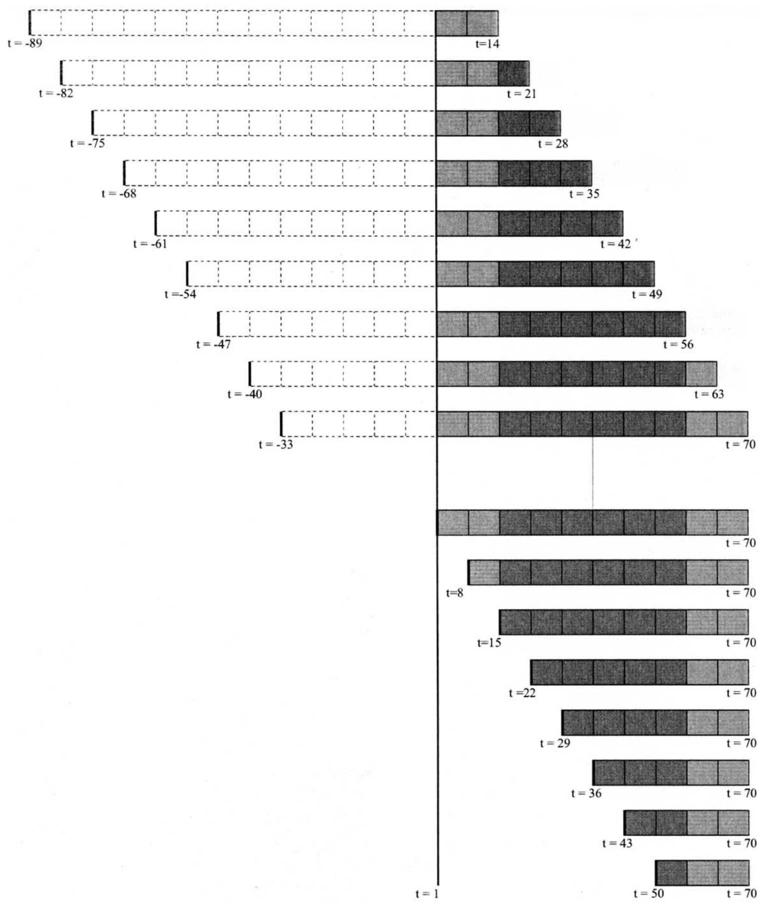


Figure 2: Illustration of the rolling decision periods in the test case



Because a booking request can be made 90 days in advance, the process starts at = –89. At that moment, booking control policies are derived for the decision period $t = 1$ until $t = 14$. A new booking control policy is constructed weekly, such that a next optimisation takes place at –82, which produces booking limits and bid prices for the decision period $t = 1$ until $t = 21$. This way, every week the decision period is extended until it eventually encompasses the maximum number of 104 days. The simulation only encompasses a period of six weeks plus two times two weeks to start up and cool down, however, such that the maximum length of the decision period will be 70 days. A graphical illustration of the rolling decision periods is given in Figure 2. In this figure, the start-up and cool-down periods are shaded light and the actual evaluation period is shaded dark.

The simulations and booking control policies are programmed in C++ on a Pentium III 500 computer. Cplex 6.5 is used to optimise the mathematical programming models. For the maximum decision period of 70 days, optimisation of the deterministic model takes up 0.4 seconds and optimisation of the stochastic model with three demand scenarios takes up 2.5 seconds. In conclusion, the aspects that define the simulation environment in Table 2 are summarised.

RESULTS

By combining the deterministic and stochastic programming models with the two methods to construct booking control policies from the models, the following four booking control policies are obtained:

- deterministic nested booking limits (DNBL)
- deterministic bid prices (DBP)
- stochastic nested booking limits (SNBL)
- stochastic bid prices (SBP).

The performances of these four methods are evaluated when they are applied to the simulated environment discussed above. The performances of the booking control policies are measured over 100 simulated arrival processes. The results are compared with the performances of a simple first-come-first-serve (FCFS) policy and with the optimal acceptance policy which can be determined with hindsight.

Based on the findings of de Boer *et al.* (2002), who show that for the airline industry three or four demand scenarios suffice to capture most of the extra revenue generated by considering the stochastic nature of the demand, only three demand scenarios will be considered for the stochastic model: a low, an average and a high demand scenario. The level of the average demand scenario is defined as the expected demand and the levels of the low and high demand scenarios are defined as k times the standard deviation away from the expected demand. In this study the cases $k = 0.5$ and $k = 1$ are considered. For determining the scenario probabilities, the variable $Z_{a,L,k}$ is defined as follows:

$$Z_{a,L,k} = \frac{D_{a,L,k} - \mu_{a,L,k}}{\sigma_{a,L,k}} \tag{4}$$

where $\mu_{a,L,k}$ is the mean and $\sigma_{a,L,k}$ is the standard deviation of $D_{a,L,k}$. Then, by the central limit theorem, $Z_{a,L,k}$ is approximately distributed as a standard normal distribution. Now, because $D_{a,L,k} \geqslant \mu_{a,L,k} + k\sigma_{a,L,k}$ can be written as $Z_{a,L,k} \geqslant k$, one can approximate the probability that the demand exceeds the average demand plus k times the standard deviation by $N(-k)$, where N(.) is the distribution function of the standard normal distribution. For k is 0.5 or 1, this results in the scenario probabilities presented in Table 3. Note, that for each type of stay, the levels of the demand scenarios are defined proportionally to the probability distribution of the demand. Therefore, the scenario probabilities are the same for all types of stay and one p_j can be used for the numerous $p_{a,L,k,j}$ for $j = 1, 2, 3$.

The results for the optimal, FCFS, DNBL, DBP and both versions of the SNBL and SBP booking control policies, which are denoted by SNBL(0.5), SNBL(1), SBP(0.5) and SBP(1), are presented in Table 4.

Table 2: Specification of the test case

Ten price classes
150 identical rooms
Max. 7-day stay
Max. 90-days in advance booking
Six week evaluation period
Demand exceeds capacity
Demand for the price classes is independent
Different booking patterns for the different price classes
No demand recapturing
No overbooking
No cancellations and no-shows
No group bookings

Table 3: Scenario probabilities

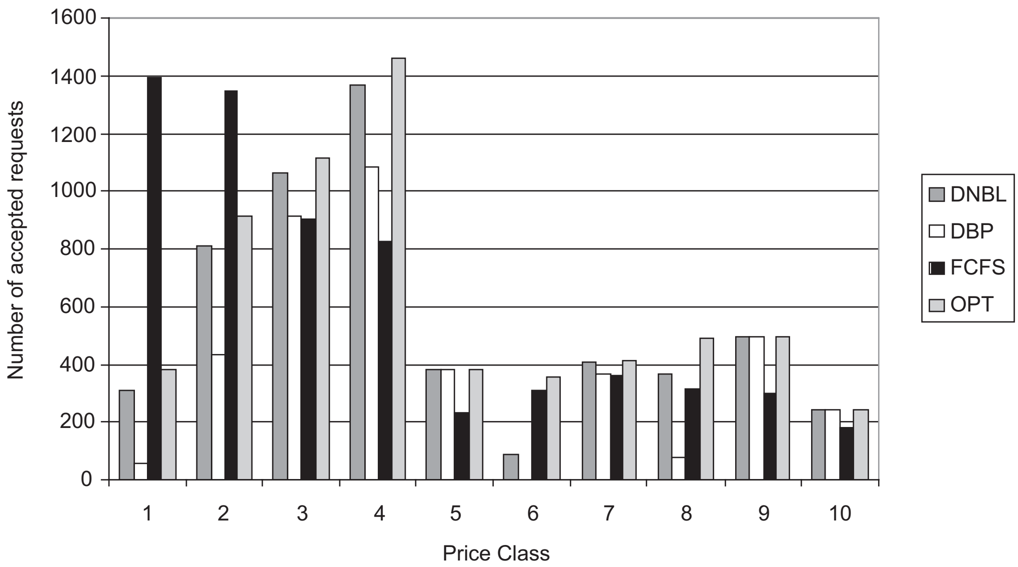
	k = 0.5	*k* = *1*
p_1 (=N(k))	0.6915	0.8413
p_2 (=N(0))	0.5	0.5
p_3 (=N(–*k*))	0.3085	0.1587

Table 4: Performances of the booking control policies

	Average revenue	*Standard deviation*	*Minimum*	*Maximum*	*Percentage optimal*
Optimal	727,565	8,479	704,140	748,100	100
FCFS	606,236	6,317	591,895	620,885	83.3
DNBL	666,642	10,898	640,370	688,590	91.6
SNBL(0.5)	691,588	8,540	668,460	712,065	95.1
SNBL(1)	681,580	6,670	661,330	696,710	93.7
DBP	535,312	10,408	510,515	559,730	73.6
SBP(0.5)	680,754	7,859	663,945	697,830	93.6
SBP(1)	668,835	8,319	649,425	690,380	91.9

Table 4 shows that there is a substantial gap between the average revenue obtained by the FCFS policy and the maximal revenue that can be obtained. Averaged over the 100 simulation runs, the FCFS policy obtains a revenue of 83.3 per cent of the optimal revenue. The DNBL policy is capable of reducing this revenue gap by half its size. On average, it reaches up to 91.6 per cent of the optimal revenue. The DBP policy, however, does not perform well at all. In fact, it performs even worse than the FCFS policy. From this, one suspects that the deterministic model does not provide the right bid prices for the problem. In Figure 3, the average number of booking requests accepted by the DNBL, DBP, FCFS and optimal policies during the six-week period are shown for each price class. It is easy to see that the FCFS policy accepts too many booking requests for the low price classes, because of which it cannot accept many requests for the higher price classes. The DBP policy, in

Figure 3: The average number of booking requests accepted for each price class by the DNBL, DBP, FCFS and optimal policies



contrast, seems to overprotect the high price classes. It accepts very few booking requests for the low price classes, ie classes 1, 2, 3, 4, 6 and 8, and cannot fully compensate for this by accepting requests for the higher price classes. This indicates that the bid prices are set too high. The DNBL policy accepts more low price class requests than the DBP policy. Also for this policy, however, the number of accepted requests for the low price classes is still systematically below the optimal level. From this, it can be concluded that the booking control policies based on the deterministic model tend to overprotect the high price classes.

Table 4 also shows that the performances of the booking limits and bid price policies improve if the stochastic model is used to construct the policies. Especially, the bid price policy seems to benefit from the use of the stochastic model. Whereas the DBP policy performs even worse than the FCFS policy, both the SBP(0.5) and SBP(1) policies perform well. They perform better than the DNBL policy, which was the best policy based on the deterministic model. The best results are obtained, however, when the booking limits policy is constructed based on the stochastic model. The SNBL(0.5) policy produces the best results of all the policies presented in this research and obtains a revenue that is on average within 5 per cent of the optimal revenue. The average number of booking requests accepted during the six-week period for each price class is shown in Figure 4 for the DNBL, SNBL(0.5), SNBL(1) and optimal policies, and in Figure 5 for the DBP, SBP(0.5), SBP(1) and optimal policies.

These figures show that the policies based on the stochastic model do not overprotect the low price classes. In fact, they tend to accept too many low-price-class booking requests because of which they cannot accept all the high-price-class requests. The stochastic model takes into account that it can be wise to allocate a room for a low-price-class request that is certain to occur instead of a high-price-class request for which the probability is very small that it will occur. If the difference in price between the low and high price classes is large, it will be worthwhile to protect rooms for the high price classes.

Figure 4: The average number of booking requests accepted for each price class by the DNBL, SNBL(0.5), SNBL(1) and optimal policies

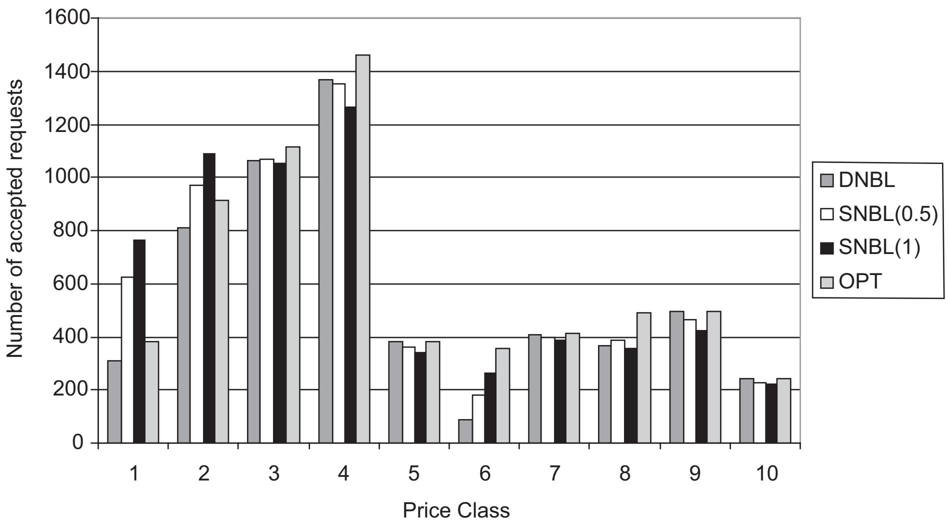
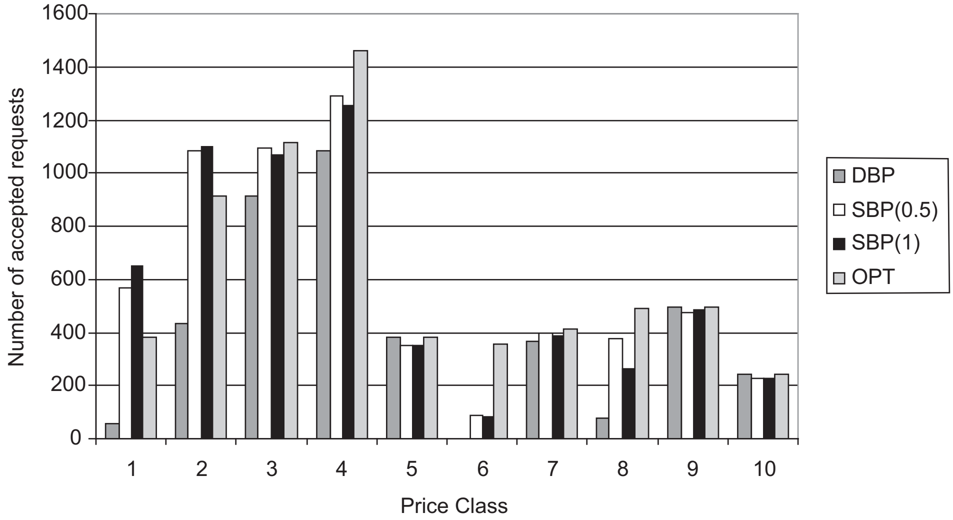


Figure 5: The average number of booking requests accepted for each price class by the DBP, SBP(0.5), SBP(1) and optimal policies



In this test case the price differences are not very large, however, the stochastic model does not protect many rooms for the high price classes.

Still the question remains why a booking control policy that underprotects the high price classes is capable of generating more revenue than a policy that overprotects the high price classes. Intuitively, there are two reasons for this: first, because a room that is protected for a high-price-class request which does not materialise does not generate any profit at all, whereas every room that is given to a low-price-class request generates at least some revenue; secondly, because the nesting policies incorporated in the booking control policies to some extent already compensate for underprotection of the high price classes. Therefore, in this setting where the price differences are not very large, it seems better to underprotect the high price classes than to overprotect them. Obviously, the best policy would be one that neither over- nor underprotects the high price classes. The authors believe that the stochastic model provides the possibility of constructing a booking control policy which approaches this kind of behaviour. In order to do so, the stochastic model should then be defined with more demand scenarios, such that the stochastic nature of the demand can be captured more accurately. Extending the stochastic model with more scenarios would, however, also increase the computational time of the model. De Boer *et al.* (2002) perform an analysis of the performances and computation times of various configurations of the stochastic model to construct booking control policies for the airline industry. It would be interesting to see whether these results hold for the hotel industry. Next to the number of demand scenarios, the spread of the scenarios is also important in the stochastic model. In this test case, both policies that make use of the stochastic model produce better results when k is set to 0.5 than when it is set to 1. This indicates that, when only three scenarios are included in the model, the scenarios should not be chosen too far apart.

Finally, it should be mentioned that results were also obtained for two other simulated demand patterns, one for which there was more high-price-class demand and less low-price-class demand and another for which there was less high-price-class demand and more low-price-class demand. Obviously, the average revenues obtained for the first case are higher and for the second case lower than for the original simulation pattern. The relative performances of the different booking control policies, however, remain the same as discussed above. This indicates that the findings discussed in this section are not exclusive for the arrival pattern used, but hold for other arrival patterns as well.

CONCLUSION

This paper studied four booking control policies for hotel revenue management and showed how to apply them over a rolling horizon of decision periods. Next to the well-known deterministic model, it also looked at a stochastic model for constructing nested booking limits and bid prices. The performances of the different booking control policies are evaluated in a simulated environment. The results show that the booking control policies based on the deterministic model tend to overprotect the high price classes. The booking control policies based on the stochastic model accept more low-price-class booking requests and obtain higher average revenues. The bid prices policy especially benefits from the use of the stochastic model. The booking control policy that performs best of all is the nested booking limits policy based on the stochastic model. The average revenue associated with this policy is within 5 per cent of the optimal revenue that can be obtained.

Following the results presented in this paper, the stochastic model formulated seems an interesting object for further research. An extensive study on the number of demand scenarios and the spreads between the scenarios to use in different hotel and demand environments would be very interesting. Further, research opportunities lie in extending the booking control policies presented here to include aspects left out of this research such as overbooking and group bookings. From the airline revenue management literature, however, it is known that some aspects of the revenue management problem, such as competition and the recapturing of rejected demand with other price classes, can be particularly hard to incorporate into a model.

References

Baker, T. K. and Collier, D. A. (1999) 'A comparative revenue analysis of hotel yield management heuristics', *Decision Sciences*, **30**, 239–263.

Bitran, G. R. and Gilbert, S. M. (1996) 'Managing hotel reservations with uncertain Arrivals', *Operations Research*, **44**, 35–49.

Bitran, C. R. and Mondschein, S. (1995) 'An application of yield management to the hotel industry considering multiple day stays', *Operations Research*, **43**, 427–443.

De Boer, S.V., Freling, R. and Piersma, N. (2002) 'Stochastic programming for multiple-leg network revenue management', *European Journal of Operational Research*, **137**, 72–92.

Weatherford, L. R. (1995) 'Length of stay heuristics: do they really make a difference', *Cornell Hotel and Restaurant Administration Quarterly,* **366**, 70–79.

Williamson, E. L. (1992) 'Airline Network Seat Inventory Control: Methodologies and Revenue Impacts', PhD thesis, Flight Transportation Laboratory, Massachusetts Institute of Technology, Cambridge, MA.

Wollmer, R. D. (1986) 'A Hub–Spoke Seat Management Model', unpublished company report, Douglas Aircraft Company, McDonnell Douglas Corporation, Long Beach, CA.