POMS PRODUCTION AND OPERATIONS MANAGEMENT

PRODUCTION AND OPERATIONS MANAGEMENT POMS

Vol. 23, No. 5, May 2014, pp. 788–801
ISSN 1059-1478 | EISSN 1937-5956 | 14 | 2305 | 0788

DOI 10.1111/poms.12065

Appointment Scheduling with No-Shows and Overbooking

Christos Zacharias, Michael Pinedo
Stern School of Business, New York University, New York, New York 10012, USA,
czachari@stern.nyu.edu, mpinedo@stern.nyu.edu

We study an overbooking model for scheduling arrivals at a medical facility under no-show behavior, with patients having different no-show probabilities and different weights. The scheduler has to assign the patients to time slots in such a way that she minimizes the expected weighted sum of the patients' waiting times and the doctor's idle time and overtime. We first consider the static problem, where the set of patients to be scheduled and their characteristics are known in advance. We partially characterize the optimal schedule and introduce a new sequencing rule that schedules patients according to a single index that is a function of their characteristics. Then we apply our theoretical results and conclusions from numerical experiments to sequential scheduling procedures. We propose a heuristic solution to the sequential scheduling problem, where requests for appointments come in gradually over time and the scheduler has to assign each patient to one of the remaining slots that are available in the schedule for a given day. We find that the no-show rate and patients' heterogeneity have a significant impact on the optimal schedule and should be taken under consideration.

Key words: appointment scheduling; no-shows; overbooking; sequential scheduling; health care
History: Received: February 2012; Accepted: February 2013 by Sergei Savin, after 4 revisions.

1. Introduction

Appointment schedules are generated on a regular basis in many different settings. Such schedules are, in particular, ubiquitous in health care. There are various sources of variability that make the appointment scheduling problem challenging. The main ones are patients' no-shows, unpunctuality, walk-ins, and random service durations. In this study, we focus on the first source of variability, namely, the no-shows.

Patients' no-show behavior and its negative impact on medical practice have been documented in many studies. To name a few, (a) Defife et al. (2010) report a 21% no-show rate in psychotherapy appointments, (b) Dreiher et al. (2008) report a 30% proportion of non-attendance in an outpatient obstetrics and gynecology clinic, and (c) Rust et al. (1995) report a 31% appointment failure rate in pediatric resident continuity clinics nationally.

Unattended appointments result in under-utilization of a clinic's valuable resources. One way to mitigate the negative impact of no-shows is the practice of overbooking. However, overbooking potentially results in clinic overcrowding, with increased patients' waiting times and physician's overtime. LaGanga and Lawrence (2007) show that appointment overbooking can significantly improve a clinic's performance by increasing patients' access and improving the physician's productivity.

We study an overbooking model for scheduling arrivals at a medical facility, with patients having different no-show probabilities and different weights. The different weights correspond to different customer classes, and the no-show probability assigned to each patient reflects her history in attending scheduled appointments. An optimal appointment schedule balances the trade-offs between the benefits of efficient resource utilization and the costs of patients' waiting time and physician's overtime.

Many studies have appeared in the literature on appointment scheduling, most of them focusing on single-server models. Cayirli and Veral (2003), and Gupta and Denton (2008) provide excellent overviews of the literature, the research challenges, and the opportunities. Hall (2012) provides a comprehensive review of models and methods used for scheduling the delivery of patient care for all parts of the health care system. The analysis may often be based on anyone of a variety of approaches, including stochastic programming, queueing models, and simulation. For the homogeneous customers case, it is of interest to determine the time intervals between consecutive arrivals throughout the working day. When patients have different characteristics, the sequencing of the arrivals is also of interest, that is, the order at which patients are scheduled to arrive at the medical facility. In most cases, finding an optimal schedule is analytically intractable, and, thus, most of the studies in the literature use enumeration, search algorithms, simulation-based techniques, and/or heuristics.

In this study, we consider two appointment scheduling paradigms. First, we assume that the set of customers who have to be scheduled on a particular day and their individual characteristics are available in advance, before the schedule is generated. Each customer can be assigned to anyone of the slots on that day. This paradigm will be referred to in our study as the "offline" scheduling problem. In our second paradigm we assume that requests for appointments come in gradually over time, and the scheduler has to fit each patient into a growing schedule for a given day. This paradigm is referred to as "online" or "sequential" scheduling. Clearly, the online scheduling problem will be in many situations more realistic.

The main contributions of this study are the following. We provide guidelines for the use of overbooking to compensate for no-shows. For the offline problem, we partially characterize the optimal schedule and introduce a new sequencing rule that schedules customers according to a single index, which is a function of their characteristics. Then we apply our theoretical results together with our conclusions from numerical experiments to develop a heuristic solution to the sequential scheduling problem. We demonstrate that the no-show rate and patients' heterogeneity have a significant impact on the optimal schedule and should be taken under consideration.

This study is organized as follows. First, we discuss the related literature. In section 3, we consider the offline scheduling problem with heterogeneous patients, each patient requiring only one time slot of service. In section 4, we extend the heterogeneity of the patients to the service times as well. In section 5, we consider the problem with homogeneous customers and demonstrate our results from numerical and simulation experiments. In section 6, we present a framework for a heuristic procedure for generating schedules in an online environment and provide an example. In section 7, we present our conclusions and suggestions for future research. The proofs are relegated to the Supporting Information of this study (Appendix S1).

2. Related Literature

A literature stream that is closely related to our work considers single-server models and patients' no-show behavior. Kaandorp and Koole (2007), Hassin and Mendel (2008), Klassen and Yoogalingam (2009), and Robinson and Chen (2010) consider the offline problem with homogeneous patients who arrive on time for their scheduled appointments, if they do show up. In Kaandorp and Koole (2007), the potential appointment slots are equally spaced, and services are exponential. Hassin and Mendel (2008) model the clinic as a single server queue with exponential service times. They find that a dome pattern is optimal: appointment intervals are initially short and increasing, they have approximately the same length in the middle of the working day, and they become short again towards the end. Klassen and Yoogalingam (2009) use a simulation optimization approach to determine optimal schedules, assuming that service times are log normal. Robinson and Chen (2010) study an overbooking model with deterministic service times and compare the performances of traditional appointment scheduling systems (patients are scheduled well in advance to fill each day) to open-access policies (meet today's demand today).

Begen and Queyranne (2011), Cayirli et al. (2012), and LaGanga and Lawrence (2012) account for heterogeneity in the no-show probabilities. Begen and Queyranne (2011) consider the problem of scheduling the arrivals of heterogeneous jobs in a given sequence, assuming discrete, random durations. Cayirli et al. (2012) use simulation and non-linear regression and propose an explicit appointment rule that takes into account the different environmental parameters of a medical facility. LaGanga and Lawrence (2012), assuming deterministic service times and that different time slots during the day have different no-show rates, develop a fast and near-optimal solution procedure.

Studies have appeared also in sequential scheduling. Muthuraman and Lawley (2008) study the problem of sequentially scheduling patients with different no-show probabilities. They develop a myopic sequential scheduling algorithm that does not consider future arrivals when making an assignment and the decision to accept or reject a patient does not depend on her type. Zeng et al. (2010) extend the work of Muthuraman and Lawley (2008) and propose two more sophisticated sequential scheduling algorithms. Liu et al. (2010) propose heuristic dynamic policies for scheduling requests for appointments from a homogeneous customer pool, taking into account time-dependent cancelations and no-shows. LaGanga and Lawrence (2012) show how their solution procedure for the offline problem can be adapted to an online environment as well.

Our model, which focuses on the interplay between the different no-show probabilities and the different weights of non-homogeneous customers, has, to our knowledge, not yet been considered in the literature.

3. Heterogeneous Patients: Identical Service Times

We consider a service provider who has in her regular schedule n time slots available to serve customers within one working day. Beyond these n regular slots, she can serve customers in overtime slots as well. Arrivals are driven by scheduled appointments. There are m, $m \geq n$, customers who have to be scheduled throughout the working day. Let $y = m - n$ denote the level of overbooking. The scheduler would like to assign each one of the customers to arrive at the beginning of one of the time slots. Customer $j, j = 1,\ldots,m$, will show up with probability $r_j = 1 - q_j$ at the beginning of the time slot she was assigned; she has a weight w_j and requires one time slot of service.

There are three costs associated with an appointment schedule: customers' waiting cost and doctor's idle time and overtime costs. The objective is to minimize the weighted sum of the three costs. If there are no customers present during one of the regular n time slots, the service provider remains idle, and an idle time cost c_I is incurred. An overtime cost c_O is incurred for each overtime slot in which the server has to remain present at the medical facility to see patients. The scheduler may overbook certain time slots and assign more than one customer to them in order to compensate for the no-show behavior. If several customers are present at the beginning of a time slot due to overbooking, then all but one of these customers have to wait. A waiting cost w_j is incurred for each time slot in which customer j has to wait before starting service.

Assume that a subset of the m customers are known to show up with probability 1, that is, $r_j = 1$. We refer to this subset of customers as set $\mathcal{C}_1$. Let N_1 denote the number of customers in this set, and we assume that $N_1 < n$, so that an overbooking model would make sense. Let $\mathcal{C}_2$ denote the subset of customers with $r_j < 1$ and let N_2 denote the number of customers in this set.

We consider only schedules that assign all m customers to the n regular time slots. That is, no customer is assigned *a priori* at the outset to an overtime slot. However, if the doctor has not served all customers by the end of the nth slot, then she continues working overtime and overtime costs are incurred. In section 4 we allow the option of an *a priori* assignment of patients to slots beyond n. We consider static list policies: all customers in the entire schedule will be processed in the specified predetermined order. In what follows in this section, we establish some of the structural properties of an optimal schedule.

LEMMA 1. *For the class of optimal schedules the following hold:*

(i) Every slot is assigned at least one customer.

(ii) The first overbooking occurs after the last time slot with a customer from set $\mathcal{C}_1$.

(iii) There exists an optimal schedule that assigns all customers from set $\mathcal{C}_1$ to the first N_1 time slots $1,\ldots,N_1$ with no overbooking.

A direct conclusion from (i) of Lemma 1 is that "block-scheduling policies" cannot be optimal in our setting. A block-scheduling policy divides the working day into blocks, and batches of multiple customers are assigned to each one of them. Glowacka et al. (2009) evaluate the performance of such block-schedules. For the rest of the study, we consider only schedules that have at least one customer assigned to every slot.

In any schedule, some slots may be assigned just one customer, whereas other slots may be assigned multiple customers. An overbooked slot will be referred to as a *vertical* segment of the schedule. The customers in a vertical segment are also put in a given order; that is, the order in which they will receive service is predetermined. For reasons that become clear later, a sequence of customers assigned to consecutive slots with no overbooking, including the customer with the lowest priority of the preceding vertical segment, will be referred to as a *horizontal* segment of the schedule. From (i) of Lemma 1, an optimal schedule is a concatenation of alternating vertical and horizontal segments.

Let $\mathcal{A}_1$ be the class of schedules that have the following three properties: (a) at least one customer is assigned to every slot, (b) all customers from $\mathcal{C}_1$ are assigned to the first N_1 time slots $1,\ldots,N_1$ with no overbooking, and (c) all the overbooking occurs at time slot $N_1 + 1$ (the first time slot with a positive probability of the server being idle).

LEMMA 2. *All appointment schedules within class $\mathcal{A}_1$ minimize the total idle time cost and the total overtime cost.*

Consider now the two-stage optimization problem with as primary objective the minimization of the doctor's expected costs (i.e., idle cost plus overtime cost) and with the minimization of the total expected waiting cost of all m customers as a secondary objective. Equivalently, the goal is now to find within class $\mathcal{A}_1$ those schedules that minimize the total expected waiting cost of all customers. Such a schedule can be globally optimal (minimizing the weighted sum of the three costs) if the doctor's cost coefficients, c_I and c_O, are an order of magnitude larger than the customers' weights. In practice, this could correspond to a system where the doctor's availability and the clinic's resources are sufficiently more costly than the cost of having patients waiting. Our next result characterizes such a schedule (Figure 1).

PROPOSITION 1. *An appointment schedule within class $\mathcal{A}_1$ that minimizes the total expected waiting cost of all customers has the following structure:*

(i) *The customers from $\mathcal{C}_1$ are assigned to slots $1, \ldots, N_1$, each slot being assigned one customer.*
(ii) *$m - n + 1$ customers from $\mathcal{C}_2$ are assigned as a batch to slot $N_1 + 1$; they are prioritized in decreasing order of w_j.*
(iii) *The remaining $n - N_1 - 1$ customers are assigned to slots $N_1 + 2$, $N_1 + 3, \ldots, n$ in increasing order of $z_j := w_j(1 - q_j)/q_j = w_j r_j/(1 - r_j)$.*
(iv) *The customer with the lowest weight assigned to slot $N_1 + 1$ has a z_j value that is lower than that of the customer assigned to slot $N_1 + 2$.*

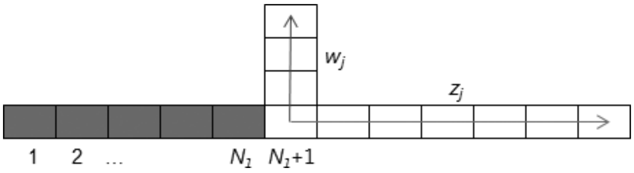
The sequencing rule in which the customers in slots $N_1 + 2, \ldots, n$ are ordered will be referred to as the *Smallest Weighted Probability of Showing up first* (*SWPS*) sequencing rule. The priority rule applied to the batch of customers in slot $N_1 + 1$ will be referred to as the *Largest Weight first* (*LW*) rule.

In contrast to the more conventional scheduling models, jobs with larger weights within the segment $N_1 + 2, \ldots, n$ tend to be scheduled later on in the schedule (this is in contrast to the well-known rule in scheduling theory that orders the jobs in *decreasing* order of their weight over processing time). It is intuitive that in our setting, the customers with larger weight are scheduled more towards the end of this segment. It is to be expected that the customers at the beginning have to wait longer for their service than those towards the end because of a queue that may have been formed by the batch of customers assigned to slot $N_1 + 1$. Toward the end of the segment, the queue of waiting customers is expected to dwindle down because of no-shows.

Note that Proposition 1 does not specify explicitly which subset of customers is assigned as a batch to slot $N_1 + 1$ and which subset of customers is assigned to slots $N_1 + 2, \ldots, n$. However, if we do consider the special case where all the weights are equal, then we can specify the entire schedule unambiguously.

COROLLARY 1. *If $w_j = w$ for all $j = 1,\ldots,m$, then an appointment schedule within class $\mathcal{A}_1$ that minimizes the total expected waiting cost of all customers has the following structure:*

Figure 1 Schedule within Class $\mathcal{A}_1$ that Minimizes the Total Expected Waiting Cost



(i) *The customers from $\mathcal{C}_1$ are assigned to slots $1, \ldots, N_1$, each slot being assigned one customer.*
(ii) *The $m - n + 1$ customers with the highest no-show probabilities from $\mathcal{C}_2$ are assigned as a batch to slot $N_1 + 1$.*
(iii) *The remaining $n - N_1 - 1$ customers are assigned to slots $N_1 + 2, N_1 + 3, \ldots, n$ in decreasing order of their no-show probability q_j.*

The latter conclusion is quite intuitive; the later a customer is assigned in the schedule, the more likely it is to show up and the more likely to encounter a shorter queue. Recall that the schedules within class $\mathcal{A}_1$ minimize the clinic's idle time and overtime costs. Proposition 1 and Corollary 1 characterize the schedules within class $\mathcal{A}_1$ that further minimize the total expected waiting cost of all the customers. In what follows, we build upon Proposition 1 to tackle the original problem, the minimization of the sum of all three costs simultaneously.

Recall that an optimal schedule is a concatenation of alternating vertical and horizontal segments. In Figure 2, a possible schedule has been depicted. Multiple customers have been assigned to slots 4, 6, 9, and 13. Let $\tau_1 < \tau_2 < \cdots$ denote those time slots that have been assigned more than one customer and that contain the vertical segments of the schedule (in Figure 2, $\tau_1 = 4$, $\tau_2 = 6$, etc.). In between time slots τ_ℓ and $\tau_{\ell+1}$, there are slots $(\tau_\ell + 1, \tau_\ell + 2, \ldots, \tau_{\ell+1} - 1)$ that have been assigned only one customer. The subset of customers that have been assigned to slots $\tau_\ell + 1$, $\tau_\ell + 2, \ldots, \tau_{\ell+1} - 1$ constitute the main part of a horizontal segment. Now, add to this subset the last customer that had been assigned to slot τ_ℓ. (Recall that the customers assigned to slot τ_ℓ had been put in a predetermined order.) Let $\mathcal{S}_\ell$ denote the resulting subset and the number of customers in this subset is $\tau_{\ell+1} - \tau_\ell$. (In Figure 2, $\mathcal{S}_1$ consists of two customers.)

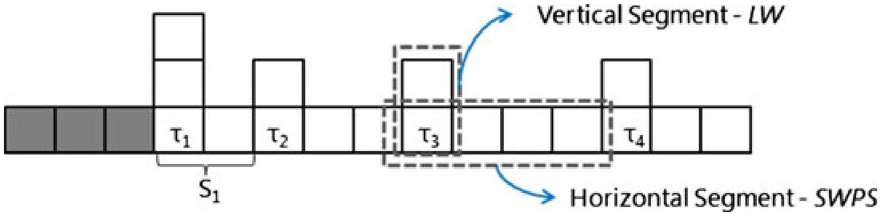
Note that a horizontal segment may at times actually consist of just a single customer. Such an extreme case occurs when there is overbooking in two consecutive time slots; the single customer is then the customer with the lowest priority in the first one of the two slots.

PROPOSITION 2.

(i) *The customers in any vertical segment of an optimal schedule are prioritized in decreasing order of w_j.*
(ii) *The customers in any horizontal segment of an optimal schedule are sequenced in increasing order of z_j.*

The last proposition implies that the optimal schedule is a sequence of the form

Figure 2 Horizontal and Vertical Segments



$$\mathcal{C}_1 - LW - SWPS - LW - SWPS - LW - SWPS\ldots$$

As before, it is of interest to consider the special case in which all weights are equal, where the optimal schedule can be more precisely determined. The optimal schedule becomes now a sequence of Vertical–Horizontal segments where customers are ordered according to the *Smallest Probability of Showing up first* (*SPS*).

Corollary 2. *If $w_j = w$ for all $j = 1,2,\ldots,m$, then all customers in a vertical segment and the immediately following horizontal segment in an optimal schedule are sequenced in decreasing order of their no-show probability q_j.*

Glowacka et al. (2009) use data mining and simulation techniques to predict patients no-shows and evaluate the performance of different scheduling methods. In the scheduling methods they consider, all patients are scheduled in increasing order of their no-show probability. As a conclusion from Corollary 2, such schedules turn out to be not optimal in our model.

The special case where $q_j = q$, for all $j = 1,2,\ldots,m$, *and customers have different weights, does not allow for an aggregation of a vertical segment with the immediately following horizontal segment. In this case, i follows from Proposition 2 that the optimal schedule takes the form*

$$\mathcal{C}_1 - LW - SW - LW - SW - LW - SW\ldots$$

That is, the customers in each vertical segment are prioritized according to the *LW first* rule and the customers in each horizontal segment are sequenced according to the *Smallest Weight* (*SW*) *first* rule.

4. Heterogeneous Patients: Non-identical Service Times

Consider now the more general case where customer j may require p_j time slots of service, p_j being an integer. All other assumptions and notation remain the same. Again, we can consider that the optimal schedule alternates between various types of subsequences. Vertical segments (where the overbooking occurs) are now much harder to define, as the different customers require different numbers of time slots. A segment of contiguous slots in which overbooking occurs may now exhibit a very complicated structure.

However, horizontal segments turn out to be easier to analyze. A horizontal segment consists of a set of customers who are assigned one after another, each one to her own time slot(s). In the following result, we consider an adjacent pairwise interchange between two customers who are scheduled one after the other within a horizontal segment. Assume that the one that goes first is scheduled at the beginning of slot t. Let B be the backlog of time slots that need to be processed by the server at the beginning of slot t, in order to finish all prior jobs. B is a random variable and depends only on the schedule prior to time slot t.

Proposition 3. *Suppose customers j and k are scheduled one after the other within a horizontal segment. If $B = 0$ with probability one, then any sequencing of j and k results in the same cost. If $B > 0$ with positive probability, then customer j should be scheduled before customer k if and only if*

$$\frac{w_j r_j}{(1-r_j)E[\min(p_j, B)]} \leq \frac{w_k r_k}{(1-r_k)E[\min(p_k, B)]}.$$

Consider now the special case where all customers have the same no-show probability q and the same weight w. The customers are distinguished only by the number of time slots they require. The following corollary follows immediately from Proposition 3.

Corollary 3. *If $w_j = w$ and $q_j = q$ for all $j = 1,2,\ldots,m$, then the customers in any horizontal segment have to be scheduled in decreasing order of p_j, that is, according to Longer Processing Time first (LPT).*

Note how the result in this corollary contrasts to a rule that is typically very popular in practice, namely, the *Shortest Processing Time first* (*SPT*) rule.

5. Homogeneous Patients

The previous sections presented some structural results and priority rules for the optimal schedule. However, we were not able to determine the exact sequence of vertical–horizontal segments of the optimal schedule and which subset of customers would be assigned to each segment.

In this section, we consider the problem where all customers have the same characteristics, that is, they all have the same weight w, have the same no-show probability $q = 1 - r$, and require a service of exactly one time slot. Even though it may seem that such special conditions would make the problem easier to analyze, it still turns out to be difficult. However, some theoretical results can still be obtained. The homogeneous customers model is similar to the model studied in LaGanga and Lawrence (2012), where they provide several properties of the optimal schedule. The crucial difference in our scheduling model is that we allow the option of an *a priori* assignment of customers to overtime slots. Proposition 4 demonstrates a condition under which such an option can be optimal.

Let a schedule be denoted by a vector $\bar{a} = (a_1, \ldots, a_m)$, where a_t is the number of customers assigned to slot t, with $\sum_{t=1}^{m} a_t = m$. It is clear that if $\bar{a}^*$ is an optimal schedule and $a_i^* = 0$ for some $i \geq n + 1$, then $a_t^* = 0$ for every $t = i + 1,\ldots,m$. Let T be the maximum t such that $a_t > 0$.

Suppose that $m = n + 1$. As there is one customer more than the number of the time slots, the optimal schedule has either one double booking in one of the first n time slots or has the extra customer scheduled to arrive in the overtime slot $n + 1$. Denote the schedule that assigns the extra customer to slot k, $1 \leq k \leq n$, as $\bar{a}^k$ and the schedule that assigns the extra customer to the overtime slot as $\bar{a}^{n+1}$, and let $z := wr/(1 - r)$.

Proposition 4. *If $m = n + 1$, then the optimal schedule is $\bar{a}^*$, where*

$$\bar{a}^* \begin{cases} = \bar{a}^1 & \text{if } z < c_I + c_O \\ \in \{\bar{a}^1, \bar{a}^2, \ldots, \bar{a}^{n+1}\} & \text{if } z = c_I + c_O \\ = \bar{a}^{n+1} & \text{if } z > c_I + c_O. \end{cases}$$

Proposition 4 completely characterizes the optimal schedule for $m = n + 1$. For larger m, we can derive recursive expressions of the objective function and determine the optimal schedule by enumeration. In order to describe our computational procedure we introduce the following notation.

Let $b(n,p,k)$ be the probability that a binomial (n,p) random variable takes a value equal to k, that is, $b(n,p,k) = \binom{n}{k} p^k (1-p)^{n-k}$. Let $B^j(\bar{a}_t) = B^j(a_1, a_2, \ldots a_t)$ denote the probability of a backlog of j customers at the end of slot t, given that $a_1, a_2, \ldots, a_t$ customers have been assigned to slot $1,2,\ldots,t$, respectively, $1 \leq t \leq T$. As a convention, let $\bar{a}_0 = 0$ and $B^0(\bar{a}_0) = 1$. Then $B^j(\bar{a}_t)$ can be expressed recursively as follows:

$$B^j(\bar{a}_t) = \begin{cases} B^0(\bar{a}_{t-1})[b(a_t, r, 0) + b(a_t, r, 1)] + B^1(\bar{a}_{t-1}) b(a_t, r, 0) & \text{for } j = 0 \\ \sum_{i=0}^{l(\bar{a}_{t-1})} B^i(\bar{a}_{t-1}) b(a_t, r, j - i + 1) & \text{for } 1 \leq j \leq l(\bar{a}_t), \end{cases}$$

where $l(\bar{a}_t) = \sum_{i=1}^{t} a_i - t$ is the maximum possible backlog at the end of slot t, $1 \leq t \leq T$.

Let $W(a,k)$ denote the total expected waiting time of a customers, who are scheduled to arrive in the same given time slot, assuming that there is already a backlog of k customers at the beginning of that slot. Then

$$W(a, k) = r \sum_{i=1}^{a} \sum_{j=0}^{i-1} (k + j) b(i - 1, r, j),$$

and the aggregated expected customers' waiting time under schedule $\bar{a}$ is

$$W(\bar{a}) = \sum_{t=1}^{T} \sum_{j=0}^{l(\bar{a}_{t-1})} B^j(\bar{a}_{t-1}) W(a_t, j).$$

If $I(\bar{a})$ denotes the total expected number of idle slots among slots $1,\ldots,n$, then

$$I(\bar{a}) = \sum_{t=1}^{n} B^0(\bar{a}_{t-1}) b(a_t, r, 0).$$

Let $O(\bar{a})$ denote the expected number of overtime slots. If no customer is assigned *a priori* to an overtime slot, that is, $T = n$, then $O(\bar{a}) = \sum_{j=0}^{l(\bar{a}_n)} j B^j(\bar{a}_n)$. When T is greater than n, there are three possible scenarios for the number of overtime slots. First, the service provider will have to stay at the medical facility for just $T - n - 1$ overtime slots if all the patients assigned to slot T do not show up and there is no backlog of patients from previous slots. Second, the service provider will have to stay at the medical facility for $T - n$ overtime slots if there is only one customer present at slot T. Otherwise, the number of overtime slots is equal to $T - n$ plus the backlog of customers at the end of time slot T. Summarizing,

$$\begin{aligned} O(\bar{a}) &= (T-n-1)B^0(\bar{a}_{T-1})b(a_T,r,0) \\ &\quad + (T-n)[B^0(\bar{a}_{T-1})b(a_T,r,1) \\ &\quad + B^1(\bar{a}_{T-1})b(a_T,r,0)] + \sum_{j=1}^{l(\bar{a}_T)}(T-n+j)B^j(\bar{a}_T) \\ &= \sum_{j=0}^{l(\bar{a}_T)}(T-n+j)B^j(\bar{a}_T) - B^0(\bar{a}_{T-1})b(a_T,r,0),\ T>n. \end{aligned}$$

The objective is to find an optimal schedule $\bar{a}^* = (a_1^*, \ldots, a_m^*)$ that minimizes the total expected cost $V^* = \min_{\bar{a}}\{V(\bar{a}) = c_I I(\bar{a}) + wW(\bar{a}) + c_O O(\bar{a})\}$. Equivalently, $\bar{a}^* = \arg\min_{\bar{a}}\{V(\bar{a}) = I(\bar{a}) + wW(\bar{a}) + c_O O(\bar{a})\}$, by normalizing the objective function with respect to c_I, that is, $c_I = 1$.

Throughout our numerical analysis, we consider homogeneous customers. Following the literature, see Robinson and Chen (2010, 2011), we consider an overtime cost coefficient $c_O = 1.5$ and a waiting cost coefficient w between zero and one. Our analysis is based on a working day consisting of $n = 12$ time slots. The procedure is programmed in MATLAB R2012a. In finding an optimal schedule, we use complete enumeration among the schedules that assign at least one customer to every slot, up to slot T.

5.1. Optimal Schedules

Optimal schedules are presented in Table 1, for different values of the waiting cost w and no-show probabilities q, with the number of customers to be scheduled, m, being subject to optimization. It is evident, and intuitive, that the overbooking level y is decreasing in w and increasing in q. As in LaGanga and Lawrence (2012), we observe that the majority of optimal schedules are *front loaded*, most of the overbooking occurs more towards the beginning of the schedule. It is also evident that, as w increases and as q decreases, the optimal schedule becomes less front loaded. Moreover, we observe that in an optimal schedule (when m is subject to optimization and not fixed) no customer is assigned *a priori* to an overtime slot, whereas this is not the case in our next experiment, where m is kept fixed.

Next, we examine the effect of the no-show probability q while the number of patients m is exogenously determined and not subject to optimization. Figure 3 lists the optimal schedules $\bar{a}^*$ as a function of q. We keep the number of patients m fixed at 18 and the waiting cost at $w = 0.1$. In Figure 4, we decompose the cost associated with the optimal schedule into its three components: idle time, overtime, and waiting cost. For low no-show probabilities, the overtime cost inevitably is very high and dominates the other two costs. For low values of q there is little or no overbooking, and some customers

Figure 3 $\bar{a}^*$ as a Function of q

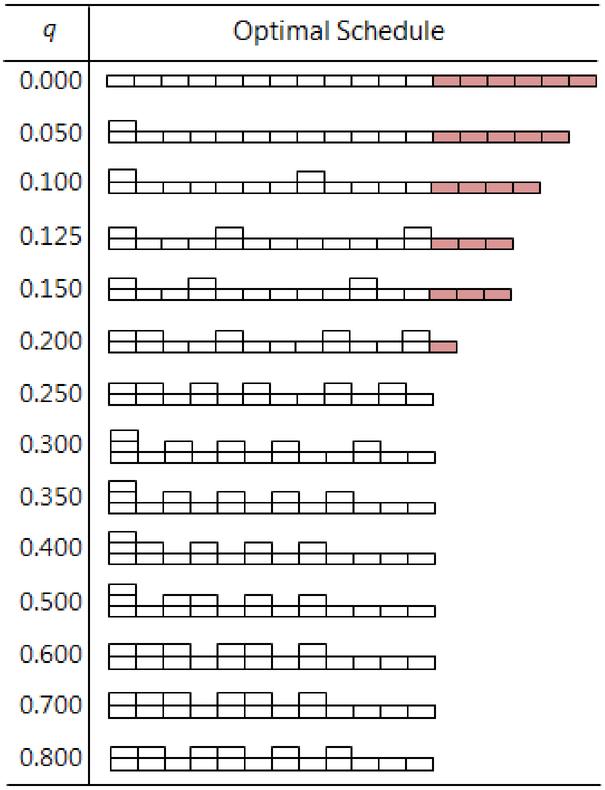


Table 1. Optimal Schedules for Different Weights and No-Show Probabilities

w	*q* = 0.2												*q* = 0.3												*q* = 0.4											
0.01	3	1	1	1	1	1	1	1	1	1	1	1	3	2	1	2	1	1	1	1	1	1	1	1	4	2	1	2	1	2	1	1	1	1	1	1
0.05	2	1	2	1	1	1	1	1	1	1	1	1	2	2	1	1	2	1	1	1	1	1	1	1	3	2	1	2	1	2	1	2	1	1	1	1
0.10	2	1	1	1	1	1	1	1	1	1	1	1	2	1	2	1	1	2	1	1	1	1	1	1	3	1	2	1	2	1	1	2	1	1	1	1
0.15	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	2	1	1	1	1	1	1	1	2	2	1	1	2	1	1	2	1	1	1	1
0.20	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	2	1	1	1	1	1	1	1	2	1	2	1	2	1	1	2	1	1	1	1
0.25	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1	2	1	2	1	2	1	1	2	1	1	1	1
0.30	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	1	1	2	1	1	1	1	1
0.40	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	1	1	1	2	1	1	1	1
0.50	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1
0.60	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1
0.70	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 4 Decomposed Cost as a Function of *q*

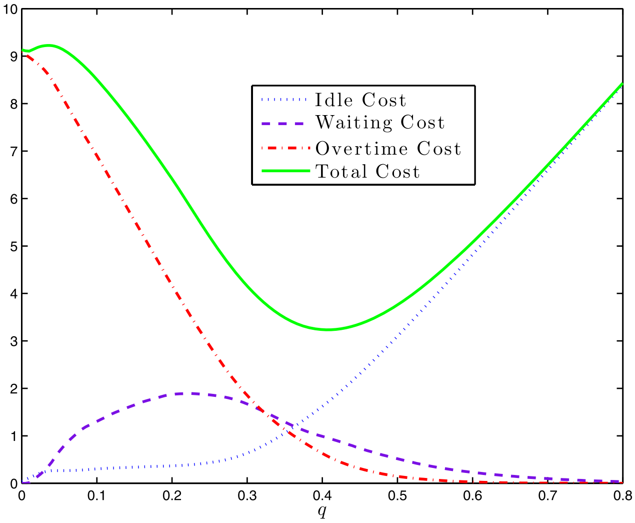
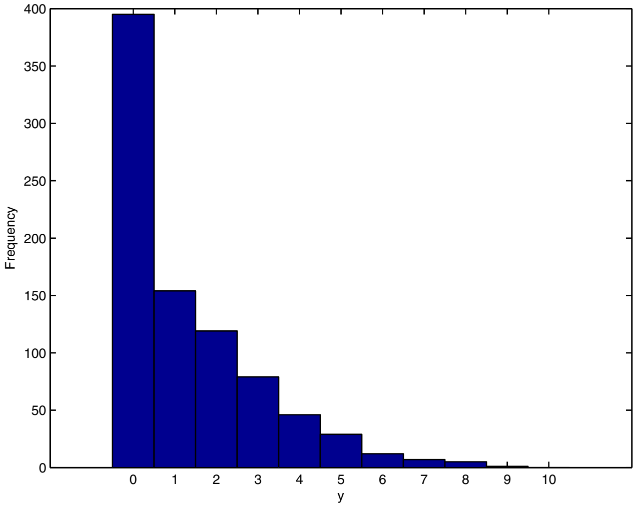


Figure 5 Overbooking Level



are assigned *a priori* to overtime slots; the marginal increase in the waiting cost due to potential overbooking outweighs the marginal decrease in the doctors' costs. As q increases from zero up to a threshold probability $\tilde{q}$ (in our example $\tilde{q} \approx 0.4$), the optimal schedule becomes more front loaded. By shifting appointments to the beginning of the working day, the scheduler manages to reduce the expected number of idle and overtime slots, and at the same time the waiting costs are kept at moderate levels, as the expected number of patients to show up is decreasing. As q increases from $\tilde{q}$ up to one, the overbooking spreads out more evenly throughout the working day and $\bar{a}^*$ becomes less front loaded. The threshold probability $\tilde{q}$ seems to be the minimizer of the total cost $V^*(q)$.

5.2. Optimal Overbooking Level

It is analytically intractable and computationally very complex to determine the optimal overbooking level y, especially when n is large. We use regression models for event count data in order to estimate y in terms of the independent variables n, q, w. The idle cost coefficient c_I is again normalized to one, we set $c_O = 1.5$, and we consider values of w between zero and one. We generate our data by solving numerically the optimal y for different values of n, q, and w. Our data set consists of 847 entries. The frequency histogram for y is shown in Figure 5.

We use *Poisson Regression* (see Greene 2008, Chapter 25) to derive a relationship between y and the independent variables n, q, w. Let $\bar{x} = (n, w, q, 1)$ be the vector of independent variables. We test the model that y has a Poisson distribution with mean $E[y|\bar{x}] = e^{\bar{\beta}'\bar{x}}$, for some coefficient vector $\bar{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)$. In Table 2, we summarize the results of the Poisson regression. The model and the estimated coefficients are statistically significant.

The Pearson's chi-squared test statistic is equal to 184.396 and $P(\chi^2(843) > 184.396) = 1$, concluding that the Pearson's goodness-of-fit test is not statistically significant and therefore the proposed model fits very well. Then, we perform the Negative Binomial regression, which relaxes the assumption of the Poisson distribution, in order to test for overdispersion. The estimate of the dispersion parameter from the Negative Binomial regression is $\hat{\alpha} = 2.32 \times 10^{-8}$, which is very close to zero, concluding that there are no signs of overdispersion and the Poisson assumption is appropriate. Thus, a good estimate of the optimal overbooking level is $y = e^{\beta_1 n + \beta_2 w + \beta_3 q + \beta_4}$, where $\bar{\beta} = (0.15, -2.70, 8.36, -3.26)$.

Table 2 Poisson Regression

	Coefficient	SE	z	$P > \|z\|$	95% CI
n	0.1480	0.0153	9.68	0.0000	[0.118,0.178]
w	−2.7036	0.1988	−13.60	0.0000	[−3.093,−2.314]
q	8.3580	0.2839	29.44	0.0000	[7.802,8.915]
1	−3.2567	0.1838	−17.72	0.0000	[−3.617,−2.896]
Log-likelihood = −723.8838		$X^2_{\text{LR}} = 1588.91$		$P(\chi^2(3) > 1588.91) = 0.0000$	

CI, Confidence interval; SE, Standard error.

5.3. Impact of Variability in Service Times

Our overbooking model for appointment scheduling, in order to focus specifically on the uncertainty caused by no-shows, ignores the variability in service times. Deterministic service times, however, have been considered in the literature before and have shown to be a good approximation: Green and Savin (2008) demonstrate the reliability of deterministic service times by comparing their model with a more realistic simulation-based appointment system. LaGanga and Lawrence (2007) also assume deterministic service times and demonstrate that their conclusions apply also when service times are uncertain. Finally, Gupta and Wang (2012) discuss the complexity of appointment scheduling problems, and they point out that models become intractable if multiple features are considered simultaneously.

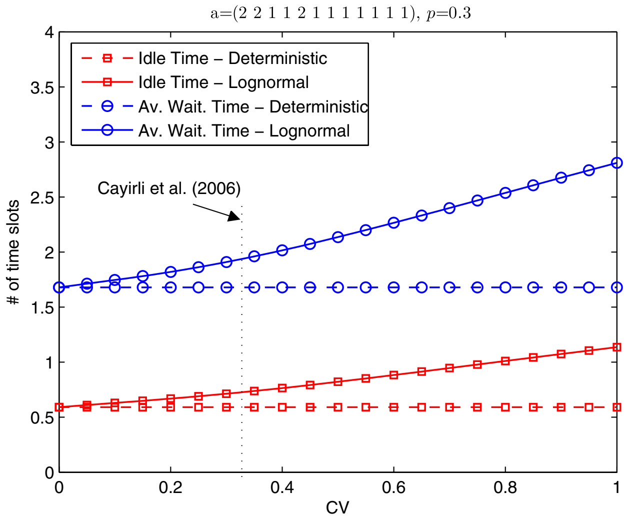
Empirical studies have shown that service times in certain medical care services have the lognormal distribution. Cayirli et al. (2006) analyze the data collected from a primary health care clinic in a New York metropolitan hospital that provides service to about 300,000 outpatients a year. They find that a lognormal distribution with a coefficient of variation (CV) of 0.325 is the best fit for service times of return customers.

We conduct simulation experiments to examine the impact of service times' variability on optimal schedules. In Figure 6, we compare the average patients' waiting time and physician's idle time in our model with the corresponding performance measures of a simulated clinic, where service times have a lognormal distribution with the same mean, for different coefficients of variation. In particular, for the simulation experiments, the service times have a mean of 1 time slot and standard deviation of CV time slots. As expected, the deterministic model is very accurate for low values of the CV, and as variability increases, the more our model underestimates patients' waiting times. Interestingly, we observe that the physician's idle time is increasing in the CV as well.

Increased patients' waiting times suggest that an optimal schedule should be less front loaded than the schedules provided by our deterministic model. On the other hand, increased physician's idle time suggests the exact opposite: an optimal schedule should be more front loaded than the schedules provided by our deterministic model. It is evident that an optimal schedule highly depends on the cost coefficients as well. Nevertheless, we believe that the effects from underestimating patients' waiting times and the physician's idle time cancel each other out to some degree, making the schedules of our deterministic model near optimal. Our next experiment was designed to test that belief.

In Table 3, we compare the optimal schedules when service times are lognormal, denoted as $\bar{a}_l^*$, with the optimal schedules of the deterministic model, denoted as $\bar{a}^*$. It turns out that indeed the deterministic model generates close to optimal schedules when the service times are log normal. For almost all the instances, the overbooking level is the same for $\bar{a}^*$ and $\bar{a}_l^*$. For small values of the waiting cost coefficient, the schedules $\bar{a}_l^*$ tend to be slightly more front loaded than $\bar{a}^*$, whereas for larger values of the waiting cost coefficient we observe that $\bar{a}_l^*$ tend to be slightly less front loaded than $\bar{a}^*$. Finally, for all the instances, the cost differences associated with schedules $\bar{a}^*$ and $\bar{a}_l^*$ are less than 5%. The effects from underestimating patients' waiting times and physician's idle time cancel each other out to some degree, making the schedules $\bar{a}^*$ near optimal when the service times are assumed to follow a lognormal distribution.

Figure 6 Deterministic vs. Lognormal Service Times



6. Sequential Scheduling

The theoretical results obtained in the previous sections were based on the assumption that all information regarding appointments for a given day are made available to the scheduler at the same time. In practice, this is often not the case. Usually, requests for appointments come in one at a time. Each time, the scheduler has to assign the customer to one of the open slots without having any knowledge regarding potential future requests for appointments on that day. In the scheduling literature, this type of environment is often referred to as an online scheduling environment, and the heuristics developed for online environments are often based on results obtained for offline scheduling.

In this section, we describe how the theoretical results obtained in the previous section for the offline

Table 3 Optimal Schedules: Deterministic vs. Lognormal Service Times

	$q = 0.2$												$q = 0.3$												$q = 0.4$											
$w = 0.01$																																				
$\bar{a}^*$	3	1	1	1	1	1	1	1	1	1	1	1	3	2	1	2	1	1	1	1	1	1	1	1	4	2	1	2	1	2	1	1	1	1	1	1
$\bar{a}_l^*$	3	1	1	1	1	1	1	1	1	1	1	1	3	2	1	2	1	1	1	1	1	1	1	1	5	2	1	2	1	1	1	1	1	1	1	1
$w = 0.05$																																				
$\bar{a}^*$	2	1	2	1	1	1	1	1	1	1	1	1	2	2	1	1	2	1	1	1	1	1	1	1	3	2	1	2	1	2	1	2	1	1	1	1
$\bar{a}_l^*$	2	2	1	1	1	1	1	1	1	1	1	1	3	1	1	2	1	1	1	1	1	1	1	1	4	1	2	1	2	1	1	1	1	1	1	1
$w = 0.10$																																				
$\bar{a}^*$	2	1	1	1	1	1	1	1	1	1	1	1	2	1	2	1	1	2	1	1	1	1	1	1	3	1	2	1	2	1	1	2	1	1	1	1
$\bar{a}_l^*$	2	1	1	1	1	1	1	1	1	1	1	1	2	2	1	1	2	1	1	1	1	1	1	1	3	2	1	2	1	2	1	1	1	1	1	1
$w = 0.15$																																				
$\bar{a}^*$	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	2	1	1	1	1	1	1	1	2	2	1	1	2	1	1	2	1	1	1	1
$\bar{a}_l^*$	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	1	1	1	1	1	1	1	1	3	1	2	1	1	2	1	1	1	1	1	1
$w = 0.20$																																				
$\bar{a}^*$	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	2	1	1	1	1	1	1	1	2	1	2	1	2	1	1	2	1	1	1	1
$\bar{a}_l^*$	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	2	1	1	1	1	1	1	1	3	1	1	2	1	1	2	1	1	1	1	1
$w = 0.25$																																				
$\bar{a}^*$	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1	2	1	2	1	2	1	1	2	1	1	1	1
$\bar{a}_l^*$	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	2	2	1	1	1	1	2	1	1	1	1	1
$w = 0.30$																																				
$\bar{a}^*$	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	1	1	2	1	1	1	1	1
$\bar{a}_l^*$	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	2	1	2	1	1	1	2	1	1	1	1	1
$w = 0.40$																																				
$\bar{a}^*$	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	1	1	1	2	1	1	1	1
$\bar{a}_l^*$	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1
$w = 0.50$																																				
$\bar{a}^*$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1
$\bar{a}_l^*$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1
$w = 0.60$																																				
$\bar{a}^*$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1
$\bar{a}_l^*$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1
$w = 0.70$																																				
$\bar{a}^*$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$\bar{a}_l^*$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

version of the appointment scheduling problem can be used in the design of a heuristic procedure for the online version of the problem.

6.1. Heuristic Procedure

Assume that the scheduler (or the scheduling system) has aggregate data regarding the total potential client pool that may call in with requests for appointments. That is, the scheduler has beliefs, based on past statistics, about the no-show probability q_j as well as about the weight w_j of each customer and where these values fit within the distribution of the entire client pool. Based on these aggregate data, the scheduler can also derive the distribution of the index values $z_j = w_j q_j/(1 - q_j) \sim Z$. Let F_Z be the cumulative distribution function of Z.

Our heuristic procedure for this online appointment scheduling problem consists of two phases. Phase I establishes a general framework for the online scheduling of any given day in the future. The design of the general framework to be used for the construction of any schedule depends on the aggregate data of the client pool as well as on the service provider's perception of what the costs are of an idle time slot, an overtime slot, and other environmental factors, depending on the practice. Phase II of the online heuristic establishes a procedure that, given the framework established in Phase I, creates a schedule gradually over time as calls come in with requests for appointments. In Phase II it is assumed that whenever any given customer calls for an appointment, the scheduler knows from the database the particular parameter values of the customer calling, that is, her w_j, q_j, and z_j, and where these values fit within the distribution of the entire client pool. Gupta and Wang (2012) refer to Phase I as the *clinic profile setup problem* and to Phase II as the *appointment booking problem*.

Phase I of the heuristic deals with the design of the structural framework of the schedule and consists of three steps. First, determine a target number of customers to overbook $y = m - n$, taking into account the aggregate data of the entire customer pool and other exogenous factors, depending on the special characteristics of the practice. A good approximation for y is given in the previous section. The number of overbookings could also be derived empirically; however, it is clear that certain structural monotonicity results should hold. For example, the higher the value of the average weight of the potential customer pool, the lower the value of y. LaGanga and Lawrence (2007, 2012) provide guidelines about the use of overbooking to improve patients' access and the clinic's performance.

Once y is determined, the scheduler has to decide which slots will be the targets for overbooking. One way is to approximate the non-homogeneous customer pool by considering a homogenous customers pool with the same average no-show probability and the same average weight and considering the optimal schedule structure that can be obtained for the homogeneous customer problem. From the numerical results presented in the previous section, it appears that the scheduler can partition the total day into y segments following a front loaded pattern: segments at the beginning of the schedule are shorter than the those towards the end. The first slot of each segment is a target slot for overbooking.

With the potential overbooking slots determined, it is thus determined what the horizontal segments of the schedule are. Knowing the distribution of the index values and taking into account that in an optimal schedule every horizontal segment should be scheduled according to the *SWPS* rule, an appropriate range of customers' priority index values z_j can be determined for each slot in each horizontal segment. If one horizontal segment consists of k slots, then an appropriate set of index values for the ith slot of that segment is $\{z : (i-1)/k < F_Z(z) \leq i/k\}$. The same structural framework is used for each day as a starting point for the schedules to be generated.

Summarizing, Phase I consists of the following three steps:

Step 1: Determining the target number of customers to overbook.
Step 2: Determining the specific time slots for overbooking.
Step 3: Determining an appropriate set of index values for each slot.

Phase II of the heuristic is the *Schedule Generation* phase, which can be described as follows: The goal is to generate, without any prior knowledge concerning the set of customers still about to call, a final schedule that more or less satisfies the optimal properties that have been derived for the offline schedules. That is, each segment should end up being scheduled according to the *SWPS* rule. The scheduler assigns the first n customers who call to time slots that are appropriate for their index values, as determined in Phase I. If a customer calls and all the time slots that would have been appropriate for her index value have already been taken, schedule her in an adjacent time slot, without violating the priority rules. If still no such slot is available, schedule this customer to the slot that violates the priority rules the least. After the first n calls have been assigned, one to each slot, late calls are assigned to the overbooking slots determined in Phase I. The customers in each overbooking slot are prioritized according to the *LW* priority rule. The scheduler does not offer more than $n + y$ appointments for the same day.

Summarizing, Phase II consists of the following two steps:

Step 1: Assignment of the first n customers following the structure of Phase I.
Step 2: Assignment of late calls to overbooking slots, as determined in Phase I.

6.2. Example

We are applying the proposed heuristic procedure to the following setting. A working day consists of $n = 12$ appointment slots; the average weight and average no-show probability of the entire client pool are $\bar{w} = 0.2$ and $\bar{q} = 0.3$, respectively. The idle time cost coefficient c_I is normalized to 1, and we consider $c_O = 1.5$. The desired overbooking level is $y = e^{\beta_1 n+\beta_2 \bar{w}+\beta_3 \bar{q}+\beta_4} = 1.626 \approx 2$. The goal is to sequentially schedule $14 = n + y$ requests for appointments (which come in one after the other) for a given working day, with no advance information regarding future calls (Step 1 of Phase I). The weight of an incoming request for an appointment is a random variable W that assumes a low value w_L with probability 1/2 and a high value w_H with probability 1/2. The probability of no-show of an incoming request for an appointment is a random variable Q, which takes a low value q_L with probability 1/2 and a high value q_H with probability 1/2. We assume that W and Q are independent. There are four types of customers. Type-*ij* customer corresponds to one with weight w_i and no-show probability q_j, $i,j \in \{L,H\}$. Let $Z = W(1-Q)/Q$, and $z_{ij} = w_i(1-q_j)/q_j$, for $i, j \in \{L,H\}$. Note that $z_{LH} \leq z_{HH}, z_{LL} \leq z_{HL}$, and the random variable Z takes each one of these values with probability 1/4. The ordering of z_{HH} and z_{LL} depends on the parameters w_L, w_H, q_L, q_H. Figure 7a demonstrates the optimal schedule for the homogeneous customers problem with $w = 0.2$ and $q = 0.3$ and that is the scheduling framework for Phase I (Step 2). The schedule consists of two Vertical–Horizontal segments. The first n requests are assigned to different time slots, with no overbooking (Step 1 of Phase II). Figures 7b and 7c demonstrate the desired customer types for each slot (Step 3 of Phase I). We intend to fill the four horizontal segments according to the *SWPS* sequencing rule.

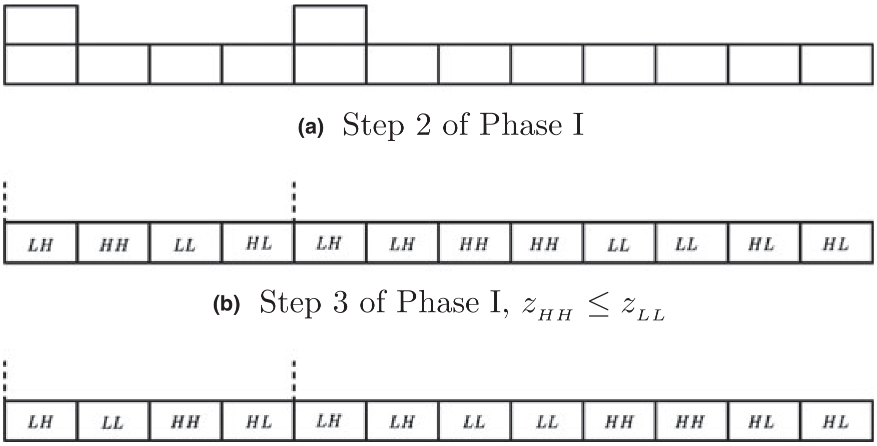
It may be the case that we are not able to schedule all the first 12 requests for appointment in agreement with Figures 7b or 7c. If no proper slot is available, then we make an assignment to an adjacent time slot, without violating the priority rules. If no such slot is available, we make an assignment to the slot that violates the sequencing rule the least. We finalize the schedule by assigning the last two calls to the slots with overbooking. The 13th request for an appointment is assigned to slot 1, the 14th to slot 5 (Step 2 of Phase II). At the overbooking slots, the customer with the higher weight has priority.

We aim to evaluate the performance of the proposed heuristic and the impact of the *SWPS* sequencing rule. For this purpose, we generate the following "base" schedule for a comparative analysis: assign the first n requests randomly to the n available slots with no overbooking (ignoring the *SWPS* rule), and assign the last y requests in the same way as in the proposed heuristic.

Note that the base schedules generated in our experimental analysis still benefit from the framework established in Phase 1 of our heuristic. That is, the overbooking still will occur in the same time slots as in the heuristic-generated schedules. So the base schedules generated in our experiments should actually perform better than an arbitrary schedule. Moreover, note that the random assignment of the first n requests is equivalent to letting the customers choose an available slot according to their preferences, while the latter are uniformly distributed over the available slots.

We simulate 100,000 different samples of 14 customers independently drawn from Z. Each sample corresponds to 14 consecutive requests for an appointment. The average weight and average no-show probability of the entire client pool are $\bar{w} = 0.2$ and $\bar{q} = 0.3$, respectively. For each sample, we compare the cost of a base schedule with the cost of a schedule generated by the heuristic described above. We repeat the experiment for different levels of variability within the client pool. We denote the range of the weights by $\Delta_w = w_H - w_L$ and the range of the no-show probabilities by $\Delta_q = q_H - q_L$. In Table 4, we summarize the average percent decrease in cost for different values of Δ_w and Δ_q.

Figure 7 Last Two Steps of Phase I



(a) Step 2 of Phase I

(b) Step 3 of Phase I, $z_{HH} \le z_{LL}$

(c) Step 3 of Phase I, $z_{LL} \le z_{HH}$

Table 4 Performance of the Proposed Heuristic

	% decrease in cost											
Δ_w \ Δ_q	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55
0.00	0.00	1.34	2.81	4.16	5.35	6.16	8.11	9.40	10.03	10.99	14.52	12.72
0.05	0.86	1.35	2.64	4.08	4.56	5.33	5.19	8.00	8.66	9.10	12.25	12.48
0.10	1.42	2.27	2.02	3.19	4.14	4.41	6.44	8.22	5.27	9.49	11.66	15.09
0.15	2.01	2.37	3.20	3.75	3.84	6.39	6.04	7.66	9.29	7.22	11.91	10.79
0.20	2.62	3.88	3.53	4.35	7.45	5.92	8.41	5.99	6.98	11.05	8.08	15.38
0.25	3.40	4.19	4.39	5.34	4.99	6.72	8.38	9.76	10.80	11.71	10.67	13.22
0.30	4.66	4.51	5.61	6.97	6.65	7.53	7.04	6.50	9.22	11.55	13.79	12.02
0.35	4.95	6.42	6.51	6.87	6.00	4.86	7.55	9.13	8.81	9.62	13.71	14.87

There is an indisputable reduction in costs when we implement the proposed heuristic, compared to the costs incurred by the base schedules, up to 15%. It is evident that the *SWPS* sequencing rule has a significant impact on the costs associated with a schedule and should be taken under consideration. Moreover, our proposed heuristic performs better as the variability in the no-show probabilities increases, whereas the variability in the weights has a smaller impact on the performance. We perform the Wilcoxon signed-rank test in order to compare the repeated measurements of the paired samples associated with the costs of the proposed and base schedules. The difference in the performance is statistically significant, for all Δ_w and Δ_q, with all the p-values being less than 0.001.

7. Conclusions

We study an overbooking model for scheduling the arrivals of patients who have different no-show probabilities and different weights. We explore the trade-offs between the benefits of efficient physician utilization and the costs of patients' waiting time.

This study considers static as well as sequential appointment scheduling and provides guidelines for the use of overbooking to compensate for no-shows. We demonstrate that the no-show rate and patients' heterogeneity have a significant impact on the optimal schedule and should be taken under consideration. Structural properties as well as a new sequencing rule are presented for the optimal offline schedules. A heuristic solution is presented for the online scheduling problem, where requests for appointments come in gradually over time and the scheduler has to fit each patient into a growing schedule for a given day.

Our overbooking model for appointment scheduling, in order to focus specifically on the uncertainty caused by no-shows, ignores the variability in service times. As demonstrated in section 4, the model with deterministic service times yields a near optimal solution to the problem where service times are lognormal. We therefore believe that the priority rule presented in this study will also prove to be useful when service times are random as well. Some other well-known priority rules that had first been established for deterministic processing times have turned out to perform very well in a stochastic environment. Consider, for example, the well-known *Weighted Shortest Expected Processing Time first* (*WSEPT*) rule. Smith (1956) had first established this rule for a deterministic environment; later on, the *WSEPT* rule turned out to be very popular in stochastic environments as well (Rothkopf 1966).

Various research directions appear to be of interest. A complete characterization of the optimal schedule in the offline environment remains an open problem. The more practical problem in the online environment requires more attention as well. The heuristic solution presented here may be made more adaptive. Phase I establishes a framework with designated overbooking slots and with target numbers of overbookings for each one of those slots. However, when calls come in and assignments are being made, the decision maker obtains gradually more information concerning the distribution of the no-show probabilities and the distribution of the weights of the customers already scheduled. With this additional knowledge, the decision maker may want to dynamically change the number of customers to overbook as well as the positions of the overbooked slots.

Acknowledgments

The authors gratefully acknowledge the constructive comments and helpful remarks from Prof. Sergei Savin, the senior editor, and the review team.

References

Begen, M. A., M. Queyranne. 2011. Appointment scheduling with discrete random durations. *Math. Oper. Res.* **36**(2): 240–257.

Cayirli, T., E. Veral. 2003. Outpatient scheduling in health care: A review of the literature. *Prod. Oper. Manag.* **2**(4): 519–549.

Cayirli, T., E. Veral, H. Rosen. 2006. Designing appointment scheduling systems for ambulatory care services. *Health Care Manag. Sci.* **9**(1): 47–58.

Cayirli, T., K. K. Yang, S. A. Quek. 2012. A universal appointment rule in the presence of no-shows and walk-ins. *Prod. Oper. Manag.* **21**(4): 682–697.

Defife, J. A., C. Z. Conklin, J. M. Smith, J. Poole. 2010. Psychotherapy appointment no-shows: Rates and reasons. *Psychotherapy Theor. Res. Pract. Train.* **47**(3): 413–417.

Dreiher, J., M. Froimovici, Y. Bibi, D. A. Vardy, A. Cicurel, A. D. Cohen. 2008. Nonattendance in Obstetrics and Gynecology of patients. *Gynecol. Obstet. Invest.* **66**(1): 40–43.

Glowacka, K. J., R. M. Henry, J. H. May. 2009. A hybrid data mining/simulation approach for modeling outpatient no-shows in clinic scheduling. *J. Oper. Res. Soc.* **60**: 1056–1068.

Green, L., S. Savin. 2008. Reducing delays for medical appointments: A queueing model. *Oper. Res.* **56**(6): 1526–1538.

Greene, W. 2008. *Econometric Analysis*, 6th edn. Prentice Hall, Englewood Cliffs, NJ.

Gupta, D., B. Denton. 2008. Appointment scheduling in health care: Challenges and opportunities. *IIE Trans.* **40**(9): 800–819.

Gupta, D., W. Y. Wang. 2012. Patient appointments in ambulatory care. R. Hall, ed. *Handbook of Healthcare System Scheduling*. Springer, New York, NY, 65–104.

Hall, R. (ed.). 2012. *Handbook of Healthcare System Scheduling*. Springer, New York, NY.

Hassin, R., S. Mendel. 2008. Scheduling arrivals to queues: A single server model with no-shows. *Manage. Sci.* **54**(3): 565–572.

Kaandorp, G. C., G. Koole. 2007. Optimal outpatient appointment scheduling. *Health Care Manag. Sci.* **10**(3): 217–229.

Klassen, K., R. Yoogalingam. 2009. Improving performance in outpatient appointment services with a simulation optimization approach. *Prod. Oper. Manag.* **18**(4): 447–458.

LaGanga, L. R., S. R. Lawrence. 2007. Clinic overbooking to improve patient access and provider productivity. *Decis. Sci.* **38**(2): 251–276.

LaGanga, L. R., S. R. Lawrence. 2012. Appointment overbooking in health care clinics to improve patient service and clinic performance. *Prod. Oper. Manag.* **21**(5): 874–888.

Liu, N., S. Ziya, V. G. Kulkarni. 2010. Dynamic scheduling of outpatient appointments under patient no-shows and cancelations. *Manuf. Serv. Oper. Manag.* **12**(2): 347–364.

Muthuraman, K., M. A. Lawley. 2008. A stochastic overbooking model for outpatient clinical scheduling with no-shows. *IIE Trans.* **40**(9): 820–837.

Robinson, L. W., R. R. Chen. 2010. A comparison of traditional and open-access policies for appointment scheduling. *Manuf. Serv. Oper. Manag.* **12**(2): 330–346.

Robinson, L. W., R. R. Chen. 2011. Estimating the implied value of the customer's waiting time. *Manuf. Serv. Oper. Manag.* **13**(1): 53–57.

Rothkopf, M. 1966. Scheduling with random service times. *Manage. Sci.* **12**(9): 703–713.

Rust, C. T., N. H. Gallups, W. S. Clark, D. S. Jones, W. D. Wilcox. 1995. Patient appointment failures in pediatric resident continuity clinics. *Arch. Pediat. Adolescent Med.* **149**(6): 693–695.

Smith, W. E. 1956. Various optimizers for single-stage production. *Naval Res. Logist. Quart.* **3**(1–2): 59–66.

Zeng, B., A. Turkcan, J. Lin, M. Lawley. 2010. Clinic scheduling models with overbooking for patients with heterogeneous no-show probabilities. *Ann. Oper. Res.* **178**: 121–144.

Supporting Information

Additional Supporting Information may be found in the online version of this article:

Appendix S1: Proofs