June 2013

WORKING PAPER SERIES

2013-MAN-01

Evaluation of appointment scheduling rules: a multi-performance measures approach

Stefan CREEMERS
IESEG School of Management (LEM-CNRS)

Pieter COLEN
Research Center for Operations Management, KU Leuven

Marc LAMBRECHT
Research Center for Operations Management, KU Leuven

IESEG School of Management
Lille Catholic University
3, rue de la Digue
F-59000 Lille
www.ieseg.fr
Tel: 33(0)3 20 54 58 92
Fax: 33(0)3 20 57 48 55

Evaluation of appointment scheduling rules: a multi-performance measures approach

Stefan Creemers[*], Pieter Colen[†], Marc Lambrecht[†]

Abstract

Appointment scheduling rules are used to determine when a customer is to receive service. Many appointment scheduling rules exist and are being used in practice (e.g., in healthcare and legal services). Which appointment scheduling rule is best, however, is still an open question. In order to answer this question, we develop an analytical model that allows to assess the performance (in terms of customer waiting time, server idle time and server overtime) of appointment scheduling rules in a wide variety of settings. More specifically, the model takes into account: (1) customer unpunctuality, (2) no-shows, (3) service interruptions and (4) delay of the service process. In addition, we allow the use of general distributions to capture system processes. We adopt an efficient algorithm (with respect to computational and memory requirements) to assess the performance of 314 scheduling rules and use data envelopment analysis to compare results.

1 Introduction

Professionals in healthcare and other services face the problem of allocating time windows to customers. This allocation can be done by means of appointment scheduling rules (ASR). ASR determine when a customer is to

[*]IESEG School of Management (LEM-CNRS), Rue de la Digue 3, 59000 Lille, France
s.creemers@ieseg.fr

[†]Research Center for Operations Management, Department of Decision Sciences and Information Management, KU Leuven, Naamsestraat 69, 3000 Leuven, Belgium
firstname.lastname@kuleuven.be

receive service during a service session. Although the literature on ASR is mainly focussed on healthcare, the research topic is generic and applicable in many industries: attorneys, faculty receiving students, tax accountants, consultants, barbers, automobile service centers, trailers at receiving bays and many others.

Conducting scheduled appointments on time is becoming ever more important across service industries. Timeliness of appointments is a key concern both for patients seeking treatment [Grote et al., 2007] and for customers who wait for field service [Apte et al., 2007]. The customer waiting time consequently is a relevant performance measure. A second important objective of appointment scheduling has to do with the efficiency of the service. For private companies, the impetus to efficiency comes naturally. But also healthcare systems are under pressure to use their capacity effectively and efficiently. Doctors' (or more general servers') idle time and overtime are hereby important performance measures. The objective of this article is to identify ASR that simultaneously minimize customer waiting time, server idle time and overtime. This has to be done in an environment where both demand and supply characteristics are highly uncertain and subject to many sources of variability.

ASR determine the planned (scheduled) arrival rate of customers during a service session. The actual arrival time may of course be different from the planned arrival time. We therefore assign each customer a probability of being too late or too early. In addition, we fix for each customer a probability of not showing up. Because of the no-show problem (i.e., customers not showing up for their appointment), the actual number of customer arrivals is unknown, even if the number of customers per session is fixed and predetermined. The performance of ASR is not only influenced by the arrival rate and service rate characteristics. Other types of outages during the service session are also important. We therefore allow for delays at the start of a service session due to late arrival of the doctor or due to setup activities at the start of a session. We also allow preemptive and non-preemptive interrupts during the service session. All these extensions allow us to model real-life appointment systems and identify ASR that have a robust performance across different settings.

We develop an analytical model that uses an efficient (in terms of computational and memory requirements) algorithm to assess the performance of ASR. The validity and accuracy of the model are supported by a simulation study. We use the model to assess the performance of a set of 314 ASR in an

elaborate computational experiment. To compare the performance of these ASR (in terms of waiting time, idle time and overtime), we apply a data envelopment analysis (DEA).

The contribution of this article is threefold: (1) we develop a new analytical model to assess the performance of an ASR in a general setting, (2) we perform an elaborate computational experiment to analyze the performance of a large number of ASR and (3) we use DEA to identify the best ASR based on multiple performance measures.

This article is organized as follows. Section 2 provides a description of the problem setting. The literature on appointment systems is discussed in Section 3. Section 4 defines the basic processes that govern the appointment system and Section 5 presents the basic model. The design and the results of the computational experiment are discussed in Section 6. Section 7 concludes.

2 Problem description

ASR are used to schedule the servicing a given number of customers during a service session. Complexity is introduced in the form of so-called "environmental variables". An extensive overview of such environmental variables is provided in Cayirli and Veral [2003]. We take the following environmental variables into account:

- Customers are allowed to arrive early, late or may even fail to show up.
- Each customers has a unique arrival process characterized by (1) a probability to show up, (2) probabilities to arrive early or late and (3) distributions to model the amount of time a customer arrives early or late.
- The start of a service session may be delayed due to the absence or lateness of staff, the setup of equipment, etc.
- The service process of a customer may be interrupted (e.g., a doctor who is called away for an emergency). We allow for both preemptive interrupts and non-preemptive interrupts.

Despite the availability of models that incorporate customer unpunctuality and customer no-show, our model is the first that allows for an individual

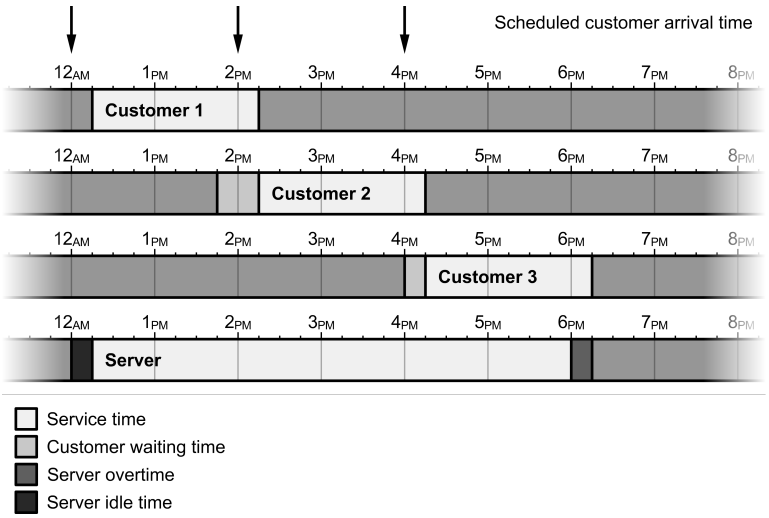


Figure 1: Dynamics of the appointment system

characterization of the arrival process. In addition, computational performance and model accuracy (and hence practical applicability) of our model significantly exceed the capabilities of comparable models in the literature on ASR.

The model has the following properties:

- Only one customer can be served at the same time (i.e., customers are served by a single server).
- Customers have i.i.d. service time distributions.
- All customers that arrive during the service session are served.
- Service is provided even if only a single customer is to be served.
- Customers that arrive early (i.e., prior to their scheduled arrival time) receive service if the server is idle. Note that this implies the possibility of overtaking other customers.

We use an example to illustrate the dynamics of an ASR. Figure 1 provides visual support. Suppose we have a service session in which 3 customers need to be scheduled. The service time requirement of a customer amounts to

exactly 2 hours, whereas the service session itself has a total duration of 6 hours. The service session starts at 12AM. Assume we schedule customers using a common variation of the well-known Bailey-Welch ASR [Welch and Bailey, 1952]. More specifically, we schedule the first customer to arrive at the start of the service session. The other customers are scheduled to arrive at 2 PM and at 4 PM respectively (i.e., the time in between two successive scheduled arrival times equals the mean service time requirement of a customer). In the example, the first customer arrives 15 minutes late, resulting in 15 minutes of server idle time (regular operating costs such as staff wages and equipment costs are still incurred). Because service starts immediately upon entry of the first customer, no customer waiting takes place. The second customer on the other hand arrives 15 minutes early and has to wait for 30 minutes prior to receiving service. The third customer arrives on time and is served after a waiting time of 15 minutes. As such, the average waiting time of a customer amounts to 15 minutes. The service session itself finishes 15 minutes late, resulting in 15 minutes of server overtime (additional costs such as penalties or staff compensation might be incurred).

The performance measures of interest are: (1) the expected waiting time of a customer, (2) the expected amount of time that the server resides in an idle state and (3) the expected amount of server overtime. Our model can provide these performance measures for any given schedule of customers (i.e., the outcome of any given ASR or scheduling procedure). In this article, however, we limit ourselves to the analysis of a set of 314 ASR.

3 Literature Review

Appointment systems (AS) have been studied extensively over the past 50 years. They arise in many contexts. In transportation, AS have been used to schedule the arrival of cargo ships and trucks at ports [Sabria and Daganzo, 1989, Giuliano and O'Brien, 2007, Namboothiri and Erera, 2008], to schedule railway operations [Wendler, 2007, Lawley et al., 2008] and to allocate airport slots [Madas and Zografos, 2006, 2008]. AS have also been adopted in telecommunication networks to schedule data transmissions [Rose and Yates, 1995, J. et al., 2006]. In manufacturing settings, AS have been used to schedule deliveries in just-in-time inventory systems [Wang, 1993, C.J. et al., 1993], to support lot-sizing decisions [Dellaert and Melo, 1998] and to schedule job release times [Tardif and Spearman, 1997, Yan and Lai,

2007, Biskup et al., 2008]. The bulk of the AS literature, however, deals with the scheduling of patients in a healthcare context. Excellent overviews of relevant literature may be found with Mondschein and Weintraub [2003], Cayirli and Veral [2003].

Nearly all of the literature on AS deals with the scheduling of customers during a single service session. Studies observing AS ranging over multiple service session are rather scarce. In Rohleder and Klassen [2002], a "rolling horizon" concept is used to schedule customers over two service sessions (before lunch and after lunch). In Vanden Bosch and Dietz [2002], customers are scheduled over several days using a heuristic approach. The computational complexity involved, limits applicability of their model to settings in which only a small number of customers can be scheduled. Creemers and Lambrecht [2009, 2010] analyze appointment-driven systems and observe the queueing behavior of a customer from the making of an appointment until the start of the service session in which the customer will receive service.

With respect to environmental variables, it is known that no-shows have a dire impact on the performance of an AS [Ho and Lau, 1992, Green, 2008, Gupta and Denton, 2008]. As such, all but a few studies incorporate the possibility of customer no-shows. Robinson and Chen [2010] and Liu et al. [2010] advocate the use of open-access AS in order to mitigate the impact of no-shows. Next to no-shows, some research also considers the occurrence of walk-ins (i.e., unscheduled customers) [Vissers, 1979, Fetter and Thompson, 1966, Klassen and Rohleder, 1996, Rohleder and Klassen, 2000, Swisher et al., 2001, Rohleder and Klassen, 2002]. The modeling of customer unpunctuality is less prevalent. Relevant literature includes Mercer [1960], Blanco White and Pike [1964], Fetter and Thompson [1966], Mercer [1973], Vissers [1979], Sabria and Daganzo [1989], Wang [1993]. Most of these models only allow for the late arrival of customers. In addition, all studies assume customer unpunctuality to be independent from the scheduled arrival times. Staff lateness (such that service cannot commence at the start of a service session) is considered in Blanco White and Pike [1964], Fetter and Thompson [1966], Vissers [1979], Babes and Sarma [1991], Liu and Liu [1998a,b]. Server interruptions are modeled in Rising et al. [1973], Lehaney et al. [1999]. Both simulation models, however, assume interrupts only to occur in between the service process of two subsequent customers (i.e., they assume non-preemtive interrupts).

Most AS literature assumes that customers are scheduled for arrival at discrete moments in time only. Individual ASR assume a single customer

to be scheduled at each of the discrete appointment times. Often, the time intervals between two such discrete appointment times are assumed to be fixed. Such studies may be found with Bailey [1952], Welch [1964], Fetter and Thompson [1966], Klassen and Rohleder [1996], Rohleder and Klassen [2000]. When allowing for multiple initial appointments (i.e., as to minimize the server idle time at the beginning of a service session) individual ASR with fixed intervals are observed in Bailey [1952], Jansson [1966], Blanco White and Pike [1964], Ho and Lau [1992], Klassen and Rohleder [1996], Ho and Lau [1999]. Block ASR allow the scheduling of multiple customers at each of the discrete appointment times (i.e., during each of the "blocks"). In Blanco White and Pike [1964], Soriano [1966], fixed block sizes (i.e., the number of appointments made at each of the discrete appointment times) as well as fixed block lengths (i.e., the time interval in between two successive discrete appointment times) are assumed. Variable block sizes and fixed intervals have been studied in Rising et al. [1973], Fries and Marathe [1981], C.J. et al. [1993], Liu and Liu [1998a,b]. Fixed block sizes and variable intervals are analyzed in Pegden and Rosenshine [1990], Wang [1997], Vanden Bosch and Dietz [2002]. Variable block sizes and variable intervals have not yet been studied.

Only a limited number of studies allow customers to have distinct service requirements. Most of these studies do not only optimize the scheduling of customers, but also the sequence of customers to be served [Weiss, 1990, Klassen and Rohleder, 1996, Rohleder and Klassen, 2000, Vanden Bosch and Dietz, 2002, 2001, Cardoen et al., 2009].

Optimization of customer appointment times usually occurs over some subset of: (1) customer waiting time, (2) server idle time and (3) server overtime. Most of the research observes either server idle time or server overtime. Surprisingly few studies assess the trade-off between all three performance measures. Well established multidimensional performance techniques, however, exist. DEA, for instance, provides a means to perform a multidimensional performance analysis based on mathematical optimization (see Cook and Seiford [2009] for an overview of the DEA literature). Fries and Marathe [1981], Kaandorp and Koole [2007] take all three performance measures into account, however, they do not use an objective technique. In order to deal with multiple performance measures, Ho and Lau [1992] adopt a frontier approach that can be considered as a simplification of a DEA [Cook and Seiford, 2009].

Ho and Lau [1992, 1999] examine 50 scheduling rules under various envi-

ronmental factors (such as the probability of no-shows, the number of patients per session, etc.). In this article, we extend the work of Ho and Lau by (1) examining more scheduling rules, (2) allowing more realistic operating environments (3) using analytical methods to obtain performance measures and (4) applying DEA to compare the performance of different ASR. Other articles related to our work are Chakraborty et al. [2010], Jouini and Benjaafar [2010] and Lian et al. [2010].

4 Definitions

In this section we classify the different ASR considered in our study. In addition, we define the basic processes that govern the AS and introduce a discretization procedure that allows us to obtain the discrete distributions of customer service and arrival times. These discrete distributions are used in the Discrete Time Markov Chain (DTMC) that is used to model the AS.

4.1 Classification of appointment scheduling rules

Most ASR may be classified in terms of:

- A_i, the scheduled arrival time of customer i,
- μ^{-1}, the mean service time of a customer,
- σ_i, the standard deviation of the service time requirement of customer i,
- N, the number of customers that require scheduling.

We implement a set of 314 ASR and use an analytical model to perform an extensive computational experiment in which the performance of these rules is assessed with respect to three performance measures in a wide variety of settings. The adopted set of ASR is an extension of the 50 ASR selected in Ho and Lau [1992, 1999]. These ASR are common in practice or have been shown to yield good and robust results.

The ASR may be summarized as variations of (1) the individual ASR, (2) the block ASR and (3) early-lateness ASR (hereafter referred to as the EL ASR).

The individual ASR schedules the arrival times of customers as follows:

$$\begin{array}{lcll} A_i & = & ia\mu^{-1} & \forall i : i < l, \\ A_i & = & A_{i-1} + \mu^{-1} + h\sigma_i & \forall i : l \leq i < N, \end{array} \tag{1}$$

where a is a multiplier to delay the start of the first arriving customers, l denotes the number of customers scheduled for arrival at the start of a service session and h is a multiplier used to adjust the impact of σ_i. We implement 91 variants of the individual ASR by allowing parameters a, l and h to vary over set $\{0, 0.3, 0.5\}$, set $\mathbf{L} = \{1, 2, 3, 4, 5\}$ and set $\mathbf{H} = \{0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3\}$ respectively.

The block ASR may be summarized as follows:

$$\begin{array}{lcll} A_i & = & 0 & \forall i : i < b, \\ A_{nb} & = & A_{(n-1)b} + b\mu^{-1} + h\sqrt{b\sigma_{nb}} & \forall n : 1 \leq n < \frac{N}{b}, \\ A_{nb+i} & = & A_{nb} & \forall i : 1 \leq i < b, \end{array} \tag{2}$$

where b denotes the block size (i.e., the number of customers assigned to arrive at a single time instance). Varying parameters b and h over set $(\mathbf{L}\backslash\{1\})$ and set $\mathbf{H}$ respectively, we obtain 28 ASR.

The EL ASR speed up and/or slow down the pace of scheduled arrivals using correction factors r_1 and r_2. The computation of scheduled arrival times is performed in two steps. First, all scheduled arrival times are initialized using the individual ASR where $(l = 1)$ and $(h = 0)$. Next, a correction is applied to speed up and/or slow down the pace of scheduled customer arrivals.

$$\begin{array}{l} \text{Initialization:} \\ \begin{array}{lcll} A_0 & = & 0, & \\ A_i & = & A_{i-1} + \mu^{-1} & \forall i : 1 \leq i < N. \end{array} \\ \text{Correction:} \\ \begin{array}{lcll} A_i & = & A_i - r_1(z-i)h\sigma_i & \forall i : 1 \leq i \leq z, \\ A_i & = & A_i - r_2(z-i)h\sigma_i & \forall i : z < i < N, \end{array} \end{array} \tag{3}$$

where r_1 and r_2 are correction factors used to speed up or slow down the succession of scheduled arrivals and z is any multiple of 5 smaller than N. Parameter r_1 controls the arrival pace of the first z customers; the arrival pace of these customers increases as r_1 increases. Conversely, parameter r_2 controls the arrival pace of those customers that are scheduled to arrive after customer z. When varying parameter h over set $(\mathbf{H} \setminus \{0\})$ and parameters r_1 and r_2 over the set $\{0, 1, 2\}$ (where $(r_1 + r_2) > 0$), we obtain 39 times $\left\lfloor \frac{N-1}{5} \right\rfloor$ ASR.

A summary of the ASR may be found in Table 1.

Table 1: Summary of the different appointment scheduling rules

Rule	$A_i = ia\mu^{-1}, \quad \forall i : i < l,$
	$A_i = A_{i-1} + \mu^{-1} + h\sigma_i, \quad \forall i : l,$
Rule no.	$1-7, 8-14, 15-21, 22-28, 29-35,$
Conditions	$l = 1, 2, 3, 4, 5 \wedge a = 0 \wedge h \in \mathbf{H},$
Rule no.	$36-42, 43-49, 50-56, 57-63,$
Conditions	$l = 2, 3, 4, 5 \wedge a = 0.3 \wedge h \in \mathbf{H},$
Rule no.	$64-70, 71-77, 78-84, 85-91,$
Conditions	$l = 2, 3, 4, 5 \wedge a = 0.5 \wedge h \in \mathbf{H},$
Rule	$A_i = 0, \quad \forall i : i < b,$
	$A_{nb} = A_{(n-1)b} + b\mu^{-1} + h\sqrt{b\sigma_{nb}}, \quad \forall n : 1 \leq n < \frac{N}{b},$
	$A_{nb+i} = A_{nb}, \quad \forall i : 1 \leq i < b,$
Rule no.	$92-98, 99-105, 106-112, 113-119,$
Conditions	$b = 2, 3, 4, 5 \wedge h \in \mathbf{H},$
Rule	initialize$A_0 = 0, \quad A_i = A_{i-1} + \mu^{-1}, \quad \forall i : 1 \leq i < N,$ then
	$A_i = A_i - r_1(z-i)h\sigma_i, \quad \forall i : 1 \leq i \leq z,$
	$A_i = A_i - r_2(z-i)h\sigma_i, \quad \forall i : z < i < N,$
Rule no.	$120-125, 159-164, 198-203, 237-242, 276-281,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 0 \wedge r_2 = 1 \wedge h \in (\mathbf{H} \setminus \{0\}),$
Rule no.	$126-128, 165-167, 204-206, 243-245, 282-284,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 0 \wedge r_2 = 2 \wedge h \in \{0.2, 0.25, 0.3\},$
Rule no.	$129-134, 168-173, 207-212, 246-251, 285-290,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 1 \wedge r_2 = 0 \wedge h \in (\mathbf{H} \setminus \{0\}),$
Rule no.	$135-140, 174-179, 213-218, 252-257, 291-296,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 1 \wedge r_2 = 1 \wedge h \in (\mathbf{H} \setminus \{0\}),$
Rule no.	$141-146, 180-185, 219-224, 258-263, 297-302,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 1 \wedge r_2 = 2 \wedge h \in (\mathbf{H} \setminus \{0\}),$
Rule no.	$147-149, 186-188, 225-227, 264-266, 303-305,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 2 \wedge r_2 = 0 \wedge h \in \{0.2, 0.25, 0.3\},$
Rule no.	$150-155, 189-194, 228-233, 267-272, 306-311,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 2 \wedge r_2 = 1 \wedge h \in (\mathbf{H} \setminus \{0\}),$
Rule no.	$156-158, 195-197, 234-236, 273-275, 312-314,$
Conditions	$z = 5, 10, 15, 20, 25 \wedge r_1 = 2 \wedge r_2 = 2 \wedge h \in \{0.2, 0.25, 0.3\}.$

4.2 Basic processes

Because of notational requirements introduced in later sections, we will sometimes use the superscript $^{(2)}$ to identify some of the basic processes. For each customer i, define:

- A_i^*, the effective arrival time,
- E_i, the earliest possible arrival instance,
- L_i, the latest possible arrival instance,
- $P[A_i^* < A_i]$, the probability of arriving early (i.e., prior to the scheduled arrival time A_i),
- $P[A_i^* > A_i]$, the probability of arriving late,
- $P[A_i^* = A_i]$, the probability of arriving on time,
- $P[\delta_i^{(2)} = 1]$, the probability of customer i not showing up (conversely, event $\left(\delta_i^{(2)} = 0\right)$ corresponds to the showing up of customer i),
- $f_i^{(E)}$, the density function of the amount of time customer i arrives early ($F_i^{(E)}$ denotes the cumulative distribution function),
- $f_i^{(L)}$, the density function of the amount of time customer i arrives late ($F_i^{(L)}$ denotes the cumulative distribution function).

The parameters of the service process of a customer may be defined as follows:

- S, the maximum service time requirement of a customer,
- $f^{(2)}$, the density function of the service time requirement of a customer ($F^{(2)}$ denotes the cumulative distribution function),
- S^*, the realized service time requirement of a customer.

Let $\mathbf{n}$ denote the set of system parameters and environmental variable settings that characterize an AS. For a given set $\mathbf{n}$ and a given schedule of customer arrivals during a service session, we obtain the following performance measures:

- $\mathcal{O}_{\mathbf{n}}$, the expected amount of overtime performed (with O being defined as the available time capacity after which overtime is performed),
- $\mathcal{I}_{\mathbf{n}}$, the expected amount of time the server resides in an idle state,
- $\mathcal{V}_{\mathbf{n}}$, the total expected customer waiting time (i.e., the expected sum of the waiting times of all customers scheduled to receive service during the service session).
- $\mathcal{W}_{\mathbf{n}}$, the expected customer waiting time.

Note that we assume the server to be idle if service of all customers is completed early (because staff wages, equipment costs, etc. are incurred until the end of the service session).

4.3 Discretization

We model the AS as a DTMC. Let Δ denote the unit time interval over which transitions are observed (e.g., we observe the state of the system every 5 minutes). During a time interval of length Δ, various events may take place (the completion of service of a customer, the arrival of one or more customers, etc.). State transitions (i.e., from a state at time instance $x\Delta$ towards a state at time instance $(x+1)\Delta$), where x is defined as $x \in \{0, 1, \ldots, \mathcal{X}\}$ and $\mathcal{X}\Delta$ is the last possible time instance at which service of all customers completes) need to take these unobserved events into account.

With respect to the service process, let $P\left[\mathcal{S}^{(2)} = x\right]$ denote the probability of finishing service during time interval $[x\Delta, (x+1)\Delta)$ (where $\mathcal{S}^{(2)}$ identifies the time interval in which service completes and equals $\left\lfloor \frac{S^*}{\Delta} \right\rfloor$). $P\left[\mathcal{S}^{(2)} = x\right]$ is computed as follows:

$$\begin{aligned} P\left[\mathcal{S}^{(2)} = x\right] &= \int_{x\Delta}^{(x+1)\Delta} f^{(2)}(t)dt \quad \forall x : x < \left\lfloor \tfrac{S}{\Delta} \right\rfloor, \\ P\left[\mathcal{S}^{(2)} = \left\lfloor \tfrac{S}{\Delta} \right\rfloor\right] &= \int_{\left\lfloor \frac{S}{\Delta} \right\rfloor \Delta}^{S} f^{(2)}(t)dt. \end{aligned} \tag{4}$$

Note that the maximum number of service phases equals $\left(Y^{(2)} = \left(\left\lfloor \frac{S}{\Delta} \right\rfloor + 1\right)\right)$. The discretization of the service process is illustrated in figure 2. The probability of completing service during a time interval $[x\Delta, (x+1)\Delta)$, given that

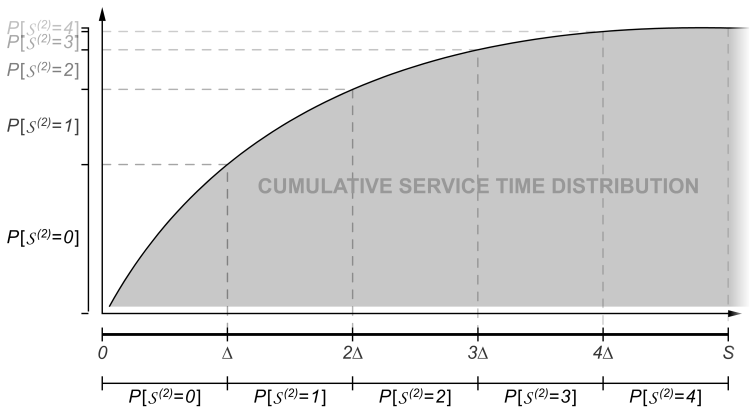


Figure 2: Discretization of the service time requirement distribution

service did not finish prior to time instance $x\Delta$, is defined as:

$$P\left[\mathcal{S}^{(2)} = x | \mathcal{S}^{(2)} > (x-1)\right] = \frac{P\left[\mathcal{S}^{(2)} = x\right]}{\sum_{n=x}^{\lfloor \frac{S}{\Delta} \rfloor} P\left[\mathcal{S}^{(2)} = n\right]} \quad \forall x : x \leq \left\lfloor \frac{S}{\Delta} \right\rfloor. \tag{5}$$

As such, the probability of finishing service during a time interval $[x\Delta, (x+1)\Delta)$ is weighted using the remaining probability mass at a time instance $x\Delta$. The weighted probability of not finishing service during a time interval $[x\Delta, (x+1)\Delta)$ is:

$$P\left[\mathcal{S}^{(2)} > x | \mathcal{S}^{(2)} > (x-1)\right] = 1 - P\left[\mathcal{S}^{(2)} = x | \mathcal{S}^{(2)} > (x-1)\right]. \tag{6}$$

For notational convenience let $P_\omega[\mathcal{S}^{(2)} = x]$ and $P_\omega[\mathcal{S}^{(2)} > x]$ be the equivalent of $P\left[\mathcal{S}^{(2)} = x | \mathcal{S}^{(2)} > (x-1)\right]$ and $P\left[\mathcal{S}^{(2)} > x | \mathcal{S}^{(2)} > (x-1)\right]$ respectively. Note that for $(x = 0)$, $P_\omega[\mathcal{S}^{(2)} = x]$ equals $P[\mathcal{S}^{(2)} = x]$ and $P_\omega[\mathcal{S}^{(2)} > x]$ equals $\left(1 - P[\mathcal{S}^{(2)} = x]\right)$.

With respect to the arrival process, $P\left[\mathcal{A}_i^* = x\right]$ denotes the probability of arrival of customer i during time interval $[x\Delta, (x+1)\Delta)$ (where $\mathcal{A}_i^*$ identifies the time interval in which customer i arrives and equals $\left\lfloor \frac{A_i^*}{\Delta} \right\rfloor$). The equations

that determine probability $P\left[\mathcal{A}_i^* = x\right]$ are presented below:

$$P[\mathcal{A}_i^* = x] = \begin{cases} P[A_i^* < A_i] + P[A_i^* = A_i] + P[A_i^* > A_i] = 1 & x = \left\lfloor \frac{E_i}{\Delta} \right\rfloor \wedge x = \left\lfloor \frac{A_i}{\Delta} \right\rfloor \wedge x = \left\lfloor \frac{L_i}{\Delta} \right\rfloor, \\ P[A_i^* < A_i] + P[A_i^* = A_i] & x = \left\lfloor \frac{E_i}{\Delta} \right\rfloor \wedge x = \left\lfloor \frac{A_i}{\Delta} \right\rfloor \wedge x < \left\lfloor \frac{L_i}{\Delta} \right\rfloor, \\ P[A_i^* = A_i] + P[A_i^* > A_i] & x > \left\lfloor \frac{E_i}{\Delta} \right\rfloor \wedge x = \left\lfloor \frac{A_i}{\Delta} \right\rfloor \wedge x = \left\lfloor \frac{L_i}{\Delta} \right\rfloor, \\ P[A_i^* = A_i] & x > \left\lfloor \frac{E_i}{\Delta} \right\rfloor \wedge x = \left\lfloor \frac{A_i}{\Delta} \right\rfloor \wedge x < \left\lfloor \frac{L_i}{\Delta} \right\rfloor, \\ P[A_i^* < A_i] \left(F_i^{(E)}(\infty) - F_i^{(E)} \left(\left(\left\lfloor \frac{A_i}{\Delta} \right\rfloor - \gamma_i^{(E)} \right) \Delta \right) \right) & x = \left\lfloor \frac{E_i}{\Delta} \right\rfloor \wedge x < \left\lfloor \frac{A_i}{\Delta} \right\rfloor, \\ P[A_i^* < A_i] \left(F_i^{(E)} \left(\left(\left\lfloor \frac{A_i}{\Delta} \right\rfloor - x - 2 \right) \Delta \right) - F_i^{(E)} \left(\left(\left\lfloor \frac{A_i}{\Delta} \right\rfloor - x - 1 \right) \Delta \right) \right) & x > \left\lfloor \frac{E_i}{\Delta} \right\rfloor \wedge x < \left\lfloor \frac{A_i}{\Delta} \right\rfloor, \\ P[A_i^* > A_i] \left(F_i^{(L)}(\infty) - F_i^{(L)} \left(\left(\left\lfloor \frac{L_i}{\Delta} \right\rfloor - \gamma_i \right) \Delta \right) \right) & x > \left\lfloor \frac{A_i}{\Delta} \right\rfloor \wedge x = \left\lfloor \frac{L_i}{\Delta} \right\rfloor, \\ P[A_i^* > A_i] \left(F_i^{(L)}(((x+1) - \gamma_i)\Delta) - F_i^{(L)}((x - \gamma_i)\Delta) \right) & x > \left\lfloor \frac{A_i}{\Delta} \right\rfloor \wedge x < \left\lfloor \frac{L_i}{\Delta} \right\rfloor. \end{cases} \tag{7}$$

Where: (1) γ_i indicates the end of the time interval in which the arrival of a customer i is scheduled to take place and (2) $\gamma_i^{(E)}$ indicates the end of the first time interval in which the customer is allowed to arrive. γ_i is defined as follows ($\gamma_i^{(E)}$ is defined analogously):

$$\gamma_i = \left\lfloor \frac{A_i}{\Delta} \right\rfloor + 1. \tag{8}$$

The maximum number of arrival phases equals $\left(Y^{(A)} = \left(\left\lfloor \frac{L_i}{\Delta} \right\rfloor - \left\lfloor \frac{E_i}{\Delta} \right\rfloor + 1 \right)\right)$. The probability of a customer i arriving during a time interval $[x\Delta, (x+1)\Delta)$, given that customer i did not arrive prior to time instance $x\Delta$, is given by:

$$P\left[\mathcal{A}_i^* = x | \mathcal{A}_i^* > (x-1)\right] = \frac{P\left[\mathcal{A}_i^* = x\right]}{\sum_{n=x}^{\left\lfloor \frac{L_i}{\Delta} \right\rfloor} P\left[\mathcal{A}_i^* = n\right]} \; \forall x : \left\lfloor \frac{E_i}{\Delta} \right\rfloor \leq x \leq \left\lfloor \frac{L_i}{\Delta} \right\rfloor. \tag{9}$$

The corresponding weighted probability of a customer not arriving during a time interval $[x\Delta, (x+1)\Delta)$ is:

$$P\left[\mathcal{A}_i^* > x | \mathcal{A}_i^* > (x-1)\right] = 1 - P\left[\mathcal{A}_i^* = x | \mathcal{A}_i^* > (x-1)\right]. \tag{10}$$

For notational convenience let $P_\omega[\mathcal{A}_i^* = x]$ and $P_\omega[\mathcal{A}_i^* > x]$ be the equivalent of $P\left[\mathcal{A}_i^* = x | \mathcal{A}_i^* > (x-1)\right]$ and $P\left[\mathcal{A}_i^* > x | \mathcal{A}_i^* > (x-1)\right]$ respectively. Note that for $(x = 0)$, $P_\omega[\mathcal{A}_i^* = 0]$ equals $P[\mathcal{A}_i^* = 0]$ and $P_\omega[\mathcal{A}_i^* > 0]$ equals $(1 - P[\mathcal{A}_i^* = 0])$.

5 Model

In this section we discuss the DTMC that is used to model the AS and that allows us to obtain the performance measures. To efficiently compute these performance measures we use an algorithm that is also introduced here.

5.1 Discrete Time Markov Chain

In order to illustrate the state transitions, define: (1) $\mathbf{N}$, the set of all customers that require scheduling and (2) $\mathbf{T}_x$, the set of customers allowed to arrive during the time interval $[x\Delta, (x+1)\Delta)$. Using the earliest and latest arrival time instances of a customer i, membership of $\mathbf{T}_x$ may easily be determined. The set of customers that have become eligible to arrive at a time instance $x\Delta$ is defined as $(\mathbf{E}_x = (\mathbf{T}_x \setminus \mathbf{T}_{x-1}))$ (with $(\mathbf{E}_0 \equiv \mathbf{T}_0)$). In addition, define the following state-dependent sets:

- $\mathbf{S}$, the set of customers that are eligible to arrive but that have not arrived yet (i.e., $\mathbf{S}$ is the subset of customers in $\mathbf{T}_x$ that did not yet arrive),
- $\mathbf{U}$, the set of customers that arrives (including no-shows),
- $\mathbf{V}$, the set of arriving customers that do not show up.

Note that $\mathbf{V} \subseteq \mathbf{U} \subseteq \mathbf{S} \subseteq \mathbf{T}_x \subseteq \mathbf{N}$ at any time instance $x\Delta$.

The AS may be modeled as a DTMC of four dimensions:

- $x\Delta$, the time instance at which the system is observed,
- $Q : Q \in \{0, 1, 2, \ldots\}$, the number of waiting customers in queue at time instance $x\Delta$,
- $y : y \in \left\{-2, 0, \ldots, Y^{(2)}\right\}$, the phase of the service process at time instance $x\Delta$ (where $(y = -2)$ indicates the completion of service of all customers, $(y = -1)$ indicates server idleness and $(y \geq 0)$ indicates that a service process is ongoing),
- $\mathbf{S}$, the set of customers that are eligible to arrive at time instance $x\Delta$ but that have not arrived yet.

Because $\mathbf{S} \subseteq \mathbf{T}_x$ at any time instance $x\Delta$, the size of the statespace depends heavily on the cardinality of set $\mathbf{T}_x$ (i.e., the size of the statespace is mainly determined by the number of customers that is allowed to arrive in parallel during a given time interval). The statespace may be divided into two sets of states: (1) transient states which are visited only once and (2) absorbing states which indicate the service completion of all customers at a given time instance (more specifically, each time instance $x\Delta$ is associated with a single

absorbing state that masses all probability to complete the service process of all customers at time instance $x\Delta$). We represent the statespace using quadruples $(x, Q, y, \mathbf{S})$. In addition, let $\pi\left[x, Q, y, \mathbf{S}\right]$ denote the probability of visiting state $(x, Q, y, \mathbf{S})$.

A state transition (from a state at time instance $x\Delta$ towards a state at time instance $(x+1)\Delta$) may result in one (or multiple) events occurring. A summary of these events is presented below:

- The arrival of a set of customers $\mathbf{U}$. The set of customers that do not arrive, is defined as $(\mathbf{U}^{\mathsf{c}} = (\mathbf{S} \setminus \mathbf{U}))$. The event in which no customers arrive is associated with sets $(\mathbf{U} = \emptyset)$ and $(\mathbf{U}^{\mathsf{c}} = \mathbf{S})$, whereas the event in which all customers arrive is associated with sets $(\mathbf{U} = \mathbf{S})$ and $(\mathbf{U}^{\mathsf{c}} = \emptyset)$.

- The not showing up of a set of arriving customers $\mathbf{V}$. The set of arriving customers that do show up, is defined as $(\mathbf{V}^{\mathsf{c}} = (\mathbf{U} \setminus \mathbf{V}))$.

- A set of customers that become eligible to arrive (i.e., $\mathbf{E}_{x+1}$).

- The completion of the service process of a customer. If the service process of a customer does not complete, it advances a phase. Note that only a single customer is allowed to complete service during a time interval of length Δ (whereas multiple arrivals are allowed to take place during the same interval).

The probability of arrival of a set of customers $\mathbf{U}$ at a state $(x, Q, y, \mathbf{S})$ is defined as $P\left[\mathbf{U}|x, \mathbf{S}\right]$. The equations determining probability $P\left[\mathbf{U}|x, \mathbf{S}\right]$ are given below:

$$P\left[\mathbf{U}|x, \mathbf{S}\right] = \begin{cases} 1 & \text{for} \quad \mathbf{U} = \emptyset \wedge \mathbf{U}^{\mathsf{c}} = \emptyset, \\ \prod\limits_{i \in \mathbf{S}} P_{\omega}[\mathcal{A}_i^* = x] & \text{for} \quad \mathbf{U}^{\mathsf{c}} = \emptyset, \\ \prod\limits_{n \in \mathbf{S}} P_{\omega}[\mathcal{A}_n^* > x] & \text{for} \quad \mathbf{U} = \emptyset, \\ \prod\limits_{i \in \mathbf{U}} P_{\omega}[\mathcal{A}_i^* = x] \prod\limits_{n \in \mathbf{U}^{\mathsf{c}}} P_{\omega}[\mathcal{A}_n^* > x] & \text{for} \quad \mathbf{U} \neq \emptyset \wedge \mathbf{U}^{\mathsf{c}} \neq \emptyset. \end{cases} \tag{11}$$

Analogously, the probability of having a set of customers $\mathbf{V}$ not showing up, when a set of customers $\mathbf{U}$ is supposed to arrive, is defined as $P\left[\mathbf{V}|\mathbf{U}\right]$.

Probabilities $P[\mathbf{V}|\mathbf{U}]$ are computed as follows:

$$P[\mathbf{V}|\mathbf{U}] = \begin{cases} 1 & \text{for} \quad \mathbf{V} = \emptyset \wedge \mathbf{V}^{\mathbf{c}} = \emptyset, \\ \prod\limits_{i \in \mathbf{U}} P\left[\delta_i^{(2)} = 1\right] & \text{for} \quad \mathbf{V}^{\mathbf{c}} = \emptyset, \\ \prod\limits_{n \in \mathbf{U}} P\left[\delta_n^{(2)} = 0\right] & \text{for} \quad \mathbf{V} = \emptyset, \\ \prod\limits_{i \in \mathbf{V}} P\left[\delta_i^{(2)} = 1\right] \prod\limits_{n \in \mathbf{V}^{\mathbf{c}}} P\left[\delta_n^{(2)} = 0\right] & \text{for} \quad \mathbf{V} \neq \emptyset \wedge \mathbf{V}^{\mathbf{c}} \neq \emptyset. \end{cases} \tag{12}$$

Seven transitions are possible at a time instance $x\Delta$:

- service is ongoing and does not finish during $[x\Delta, (x+1)\Delta)$,
- service is ongoing, finishes and at least one customer is present in the queue at time instance $(x+1)\Delta$,
- service is ongoing, finishes and although no customers are left in the queue at time instance $(x+1)\Delta$, there are still customers that have to arrive,
- service is ongoing, finishes and all customers have arrived or have failed to show up (i.e., an absorbing state has been entered; service has finished at time instance $x\Delta$).
- the server is idle and at least one customer arrives during $[x\Delta, (x+1)\Delta)$,
- the server is idle, no customer arrives during $[x\Delta, (x+1)\Delta)$ and some customers have yet to arrive,
- the server is idle, no more customers are present in the queue and all customers have arrived (i.e., an absorbing state has been entered; service has finished at time instance $x\Delta$).

A summary of all state transitions (and their probabilities) is presented in table 2.

5.2 Performance measures

The transition probabilities may be used to compute $\pi[x, Q, y, \mathbf{S}]$, the probability of visiting a state $(x, Q, y, \mathbf{S})$. Using the probabilities to visit each of these states, the performance measures may easily be obtained. More specifically, a state $(x, Q, y, \mathbf{S})$ (with corresponding probability $\pi[x, Q, y, \mathbf{S}]$) is associated with:

Table 2: Summary of all state transitions when departing from state $(x, Q, y, \mathbf{S})$

Arrival state:	$((x+1), \Delta Q, (y+1), \mathbf{S}^{\diamond})$,
Conditions:	$0 \leq y < Y^{(2)}$,
Transition probability:	$P_{\omega}[\mathcal{S}^{(2)} > y] P[\mathbf{U}\|x, \mathbf{S}] P[\mathbf{V}\|\mathbf{U}]$,
Arrival state:	$((x+1), (\Delta Q - 1), 0, \mathbf{S}^{\diamond})$,
Conditions:	$0 \leq y \leq Y^{(2)} \wedge \Delta Q > 0$,
Transition probability:	$P_{\omega}[\mathcal{S}^{(2)} = y] P[\mathbf{U}\|x, \mathbf{S}] P[\mathbf{V}\|\mathbf{U}]$,
Arrival state:	$((x+1), 0, -1, \mathbf{S}^{\diamond})$,
Conditions:	$0 \leq y \leq Y^{(2)} \wedge \Delta Q = 0$,
Transition probability:	$P_{\omega}[\mathcal{S}^{(2)} = y] P[\mathbf{U}\|x, \mathbf{S}] P[\mathbf{V}\|\mathbf{U}]$,
Arrival state:	$((x+1), 0, -2, \emptyset)$,
Conditions:	$0 \leq y \leq Y^{(2)} \wedge \Delta Q = 0 \wedge \mathbf{V} = \mathbf{U} = \mathbf{S} = \mathbf{E}$,
Transition probability:	$P_{\omega}[\mathcal{S}^{(2)} = y] P[\mathbf{U}\|x, \mathbf{S}] P[\mathbf{V}\|\mathbf{U}]$,
Arrival state:	$((x+1), (\Delta Q - 1), 0, \mathbf{S}^{\diamond})$,
Conditions:	$y = -1 \wedge \Delta Q > 0$,
Transition probability:	$P[\mathbf{U}\|x, \mathbf{S}] P[\mathbf{V}\|\mathbf{U}]$,
Arrival state:	$((x+1), 0, -1, \mathbf{S}^{\diamond})$,
Conditions:	$y = -1 \wedge \Delta Q = 0 \wedge \mathbf{E} \neq \emptyset$,
Transition probability:	$P[\mathbf{U}\|x, \mathbf{S}] P[\mathbf{V}\|\mathbf{U}]$,
Arrival state:	$((x+1), 0, -2, \emptyset)$,
Conditions:	$y = -1 \wedge \Delta Q = 0 \wedge \mathbf{V} = \mathbf{U} = \mathbf{S} = \mathbf{E}$,
Transition probability:	$P[\mathbf{U}\|x, \mathbf{S}] P[\mathbf{V}\|\mathbf{U}]$,
where:	$\Delta Q = Q + \|\mathbf{U}\| - \|\mathbf{V}\|$,
	$\mathbf{S}^{\diamond} = ((\mathbf{S} \setminus \mathbf{U}) \cup \mathbf{E}_{x+1})$,
	$\mathbf{E} = \left((\mathbf{S} \setminus \mathbf{U}) \cup \left(\bigcup_{n>x} \mathbf{E}_n \right) \right)$.

- a total customer waiting time of $Q\Delta$ time units (i.e., Q customers are waiting during time interval $[x\Delta, (x+1)\Delta)$),
- a server idle time of Δ time units if $(y = -1)$,
- a server idle time of $(O - x\Delta)$ time units if: (1) $(x\Delta < O)$ or (2) $(y = -2)$ (i.e., $(x, Q, y, \mathbf{S})$ is an absorbing state),
- a server overtime of $(x\Delta - O)$ time units if: (1) $(x\Delta > O)$ or (2) $(y = -2)$.

General performance measures may be obtained as the weighted sum of the performance measures corresponding to each of the states (where the probabilities of visiting a state serve as weights). More formally, for a given set $\mathbf{n}$ and a given schedule of customer arrivals, the expected amount of overtime performed is given by:

$$\mathcal{O}_{\mathbf{n}} = \sum_{x > \lfloor \frac{O}{\Delta} \rfloor}^{\mathcal{X}} \pi[x, 0, -2, \emptyset] \, (x\Delta - O) \, . \tag{13}$$

With respect to the expected server idle time, we obtain the following result:

$$\mathcal{I}_i = \left(\sum_{x=0}^{\mathcal{X}} \sum_{\mathbf{S} \subseteq \mathbf{T}_x} \pi[x, 0, -1, \mathbf{S}] \Delta \right) + \left(\sum_{x=0}^{\lfloor \frac{O}{\Delta} \rfloor - 1} \pi[x, 0, -2, \emptyset] \, (O - x\Delta) \right) . \tag{14}$$

The total expected customer waiting time may be expressed as:

$$\mathcal{V}_{\mathbf{n}} = \sum_{x=0}^{\mathcal{X}} \sum_{Q=1}^{N} \sum_{y=0}^{Y^{(2)}} \sum_{\mathbf{S} \subseteq \mathbf{T}_x} \pi \left[x, Q, y, \mathbf{S} \right] Q\Delta. \tag{15}$$

Conversely, the expected customer waiting time is given by:

$$\mathcal{W}_{\mathbf{n}} = \frac{\mathcal{V}_{\mathbf{n}}}{\sum_{i=0}^{N} P[\delta_i^{(2)} = 0]}. \tag{16}$$

Where $\left(\sum_{i=0}^{N} P[\delta_i^{(2)} = 0] \right)$ denotes the expected number of customers to show up.

5.3 Algorithm and implementation

The algorithm consists of two main steps: (1) initialization and selection of the ASR and (2) iterative computation of probabilities $\pi[x, Q, y, \mathbf{S}]$ and the assessment of performance measures. During the initialization, the ASR is selected. The selected rule determines the arrival process. The service process does not depend on the ASR. The iterative procedure uses probabilities $\pi[x, Q, y, \mathbf{S}]$ (associated with a time instance $x\Delta$) to compute probabilities $\pi[(x+1), Q, y, \mathbf{S}]$ (associated with a time instance $(x+1)\Delta$). Performance measures are computed simultaneously. After computation of all probabilities $\pi[(x+1), Q, y, \mathbf{S}]$, probabilities $\pi[x, Q, y, \mathbf{S}]$ are no longer needed. As such, the memory occupied by these latter probabilities may be freed. The iterations continue until all probability mass is gathered in the absorbing states. Next, performance measures corresponding to the selected ASR are stored. The process is repeated until all adopted ASR have been assessed. A general outline of the algorithm is presented in algorithm 1.

Algorithm 1 Algorithm for computing performance measures

```
for all x do
    Compute P_ω[S^(2) = x] and P_ω[S^(2) > x]
end for
for all Appointment scheduling rules do
    Compute P_ω[A_i^* = x] and P_ω[A_i^* > x]
    Set x = 0
    for all Q, y, S do
        Compute π[x, Q, y, S]
        Update performance measures
    end for
    while x < X do
        for all Q, y, S do
            Compute π[(x + 1), Q, y, S] using π[x, Q, y, S]
            Update performance measures
        end for
        Free memory used by states (x, Q, y, S)
        Increment x
    end while
    Store performance measures
end for
```

The algorithm is implemented in Visual C++. The main inputs of the application are: (1) the size of the unit time interval, (2) the number of customers that require scheduling, (3) the parameters of the service process and (4) the parameters of the arrival process of each of the customers.

5.4 Model extensions

In this section, we discuss three model extensions: (1) the delayed start of a service session, (2) interruptions that take place during the service process of a customer (i.e., preemptive interrupts) and (3) interruptions that take place in between the service process of two subsequent customers (i.e., non-preemptive interrupts, the delayed start of service itself). In order to take these extensions into account, we allow for an additional Markov chain dimension that captures the type of service process currently in progress. As such the resulting DTMC holds five dimensions. Its statespace may be represented by quintuples $(x, Q, y, w, \mathbf{S})$, where w indicates the type of service currently in progress. By convention we have:

- $(w = -1)$ if the server is idle,
- $(w = 0)$ if the service process cannot start because the start of the service session is delayed,
- $(w = 1)$ if the ongoing service process is subject to non-preemptive interrupts,
- $(w = 2)$ if a regular service process is ongoing (i.e., as defined in the previous sections),
- $(w = 3)$ if the ongoing service process is subject to preemptive interrupts.

With the exception of $(w = -1)$, these service outages are modeled as "special" customers, each associated with a unique service process characterization. More specifically, each type of service has unique parameters: $f^{(w)}$, $P[\mathcal{S}^{(w)} = x]$, $P[\mathcal{S}^{(w)} = x|\mathcal{S}^{(w)} > (x-1)]$, $P[\mathcal{S}^{(w)} > x|\mathcal{S}^{(w)} > (x-1)]$, $Y^{(w)}$ and $P[\delta^{(w)} = 1]$ (note that index i is discarded for the non-regular types of service processes). The type of the ongoing service process is decided at: (1) the start of a service session (for the delayed start of a services session), (2) the start of a service process (for the delayed and the regular start of a service process) and (3) the end of a service process (for a service process subject to preemptive interrupts). A detailed discussion of how to implement these extensions is given in Creemers [2009a].

6 Computational experiment

The validity and accuracy of the model has been verified by means of an elaborate simulation study (no ref: blind review) From their results, we conclude that a unit time interval ($\Delta = 5$) provides a sufficient level of accuracy while maintaining computational performance. As such, in the upcoming experiment we let ($\Delta = 5$). In what follows, we discuss the design and the results of the computational experiment.

6.1 Experimental design

We consider 243 operating environments that are generated by all combinations of:

- $N \in \{10, 20, 30\}$,
- squared coefficient of variation of service times in $\{0.2, 0.5, 1.0\}$ (the mean service requirement equals 300 time units),
- squared coefficient of variation of early and late arrival times in $\{0.5, 1.0\}$,
- probability of early arrival in $\{0, 0.1\}$,
- probability of late arrival in $\{0, 0.1\}$,
- probability of no-show in $\{0, 0.1, 0.2\}$.

Let $\mathbf{P} = \{\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_{243}\}$ denote the set of operating environments. In addition, define ($O = N\mu^{-1}$) as the time capacity after which overtime is performed. We evaluate 158, 236 and 314 ASR for each value of N respectively (resulting in a total of 57,348 instances analyzed). The performance measures of an ASR $r : r \in \{1, 2, \ldots, 314\}$ are:

$$\mathcal{O}_r = \sum_{i=1}^{243} \mathcal{O}_{\mathbf{n}_i,r}, \tag{17}$$

$$\mathcal{I}_r = \sum_{i=1}^{243} \mathcal{I}_{\mathbf{n}_i,r}, \tag{18}$$

$$\mathcal{W}_r = \sum_{i=1}^{243} \mathcal{W}_{\mathbf{n}_i,r}, \tag{19}$$

where $\mathcal{O}_{\mathbf{n}_i,r}$, $\mathcal{I}_{\mathbf{n}_i,r}$ and $\mathcal{W}_{\mathbf{n}_i,r}$ denote the expected server overtime, the expected server idle time and the expected customer waiting time when ASR

r is used to schedule the arrival of customers at an AS that operates in environment $\mathbf{n}_i$.

While the implementation of an ASR might yield good results in terms of a single performance measure (e.g., server idle time), its impact on the results of another performance measure (e.g., customer waiting time) can be detrimental. Hence, the need to consider multiple performance measures when evaluating ASR. To conduct a multidimensional performance evaluation, we use a composite indicator (CI) :

$$CI_r = v_{o,r}\mathcal{O}_r + v_{i,r}\mathcal{I}_r + v_{w,r}\mathcal{W}_r, \tag{20}$$

where CI_r is the weighted sum of the performance of an ASR r and $v_{(\cdot),r}$ is the weight allocated to performance measure $(\cdot)$.

An ASR r performs well if the score over all performance measures is low (i.e., the lower the value of the CI, the better the ASR performs). Although practical and intuitive, CI have several drawbacks, among which the need to normalize the performance measures and the inherent difficulty of determining appropriate weights [Cherchye et al., 2008]. Cherchye et al. [2008] have demonstrated the applicability of DEA to objectively set weights. In order to avoid the subjective fixing of weights, we use DEA to identify the optimal set of weights for each ASR individually, under the restriction that no ASR can have a CI larger than one for any of the selected weight sets.

The resulting CI are conservative (i.e., allow high weights to be set on strong performance measures and low weights on measures for which performance is bad). This can be a welcome feature as best-practice ASR can be identified. Nevertheless, zero weights are often allocated, which is problematic as every performance measure included is by definition relevant. Extensive research has been conducted in order to identify methods that allow to avoid zero weights while maintaining the statistical properties of the DEA method (e.g., Allen and Thanassoulis [2004], Portela and Thanassoulis [2006] and Cooper et al. [2007]). We opt to avoid zero weights by only allowing weight sets that are fully defined [Cooper et al., 2007, Olesen and Petersen, 1996]. A fully defined set of weights does not contain any zero values and can be found as a set of weights for which m (where m equals the number of performance measures minus one) linearly independent ASR have a CI value that equals unity [Olesen and Petersen, 1996]. After finding the set of independent ASR with CI scores equal to one (let $\mathbf{C}$ denote this set) by use of a super-efficiency model, we solve the following model for each ASR:

$$\min v_{o,r}\mathcal{O}_r + v_{i,r}\mathcal{I}_r + v_{w,r}\mathcal{W}_r \tag{21}$$

subject to

$$
\begin{aligned}
(v_{o,r}\mathcal{O}_r) + (v_{i,r}\mathcal{I}_r) + (v_{w,r}\mathcal{W}_r) &= 1 - d_r \qquad \forall r \in \mathbf{C} && (22)\\
Mb_r - d_r &\geq 0 \qquad \forall r \in \mathbf{C} && (23)\\
\sum_{r\in\mathbf{C}} b_r &= |\mathbf{C}| - 2 && (24)\\
v_{o,r} &\geq 0 && (25)\\
v_{i,r} &\geq 0 && (26)\\
v_{w,r} &\geq 0 && (27)\\
d_r &\geq 0 \qquad \forall d_r : r \in \mathbf{C} && (28)\\
b_r &\in \{0,1\} \qquad \forall b_r : r \in \mathbf{C}, && (29)
\end{aligned}
$$

where M is a large value, b_r is a binary variable that equals zero only if d_r equals zero and constraint 24 ensures that the set of weights is fully defined. Model 21 minimizes the value of the CI by selecting one of the fully defined weight sets. Consequently, the selected weights do not contain any zero values. Model 21 is solved for each of the ASR, resulting in 314 CI values and 314 sets of weights. The model yields high CI values for ASR that are less attractive. In order to make the CI more intuitive, we use the inverse value. As such, higher CI values indicate a better performing ASR, with the best-performing ASR obtaining a value of one.

Although the objectivity of a DEA-based performance evaluation is a merit, decision makers can have good reasons to value some performance measures more than others. Different possibilities exist to incorporate such valuation into the DEA [Cook and Seiford, 2009]. We incorporate the valuation of different performance measures by adding constraints:

$$\frac{v_{(\cdot),r}}{v_{(\cdot),r}} \leq 1, \tag{30}$$

where $(\cdot)$ is a performance measure. Constraint $\left(\frac{v_{o,r}}{v_{i,r}} \leq 1\right)$ for example, imposes the restriction that the expected server idle time is considered to be more important than the expected server overtime.

Some ASR will perform strongly across a wide range of possible weight sets, while others may have a CI value that depends heavily on the choice of a particular set of weights. To measure the sensitivity of the CI value to the selected set of weights, we calculate the maverick index [Doyle and Green, 1994]. If the maverick index is high, the CI score of the ASR is sensitive to the choice of weights. A low maverick index indicates a robust performance across different sets of weights.

6.2 Experimental results

In this section, we discuss the performance of all ASR over all environments. Next, we select nine ASR to assess the impact of environmental variables and to discuss the influence of subjective valuation of the different performance measures. Although our method is suitable to select the best ASR for any given setting and preference structure, we focus on general insights with respect to the three types of ASR (i.e., individual ASR, block ASR and EL ASR).

Table 3 provides an overview of the best-performing ASR across all environments and over all performance measures. The table also indicates (1) the value of the composite indicator, (2) type of ASR and (3) the maverick index. It is striking that the six best-performing ASR are all individual ASR with the common characteristics of zero delay for the first arrival, three or more initial customers and a very small (or even no) adjustment for service time variance. This observation is encouraging for practitioners as these rules are simple to implement. At the downside, the maverick index indicates that the CI values of these simple ASR are quite sensitive to the weight selection. The lower sensitivity to the weight selection can be a reason to opt for an EL ASR, which are firmly established in the top 15 as from rank seven. Interestingly, the best-performing EL ASR incorporate substantial adjustments of the appointments to compensate for the variance in service times. This is remarkable as EL ASR already incorporate an adjustment mechanism to avoid customer waiting time in the form of the earlier or postponed appointments. Block ASR are the least attractive type of ASR. The best-performing block ASR (ASR 93, 92, 94) have a block size of two with a small (or even zero) adjustment for service time variance (they are ranked at positions 72, 84 and 89, respectively). The dominance within the block ASR of rules with a block size of two (i.e., customers arrive in groups of two) confirms the conclusions of earlier research [Blanco White and Pike, 1964, Ho and Lau, 1992]. Lastly, the simple Bailey-Welch rule performs rather well both in terms of the efficiency score and weight sensitivity (respectively 99.81% and 0.0959).

Based on both prior research and the results discussed above, we select nine ASR for further discussion. Table 4 presents the selected ASR and their characteristics. We select five individual ASR of whom only ASR 7 incorporates an adjustment of the arrival times to compensate for the variance in service times. The main difference in the selected individual ASR is the number of customers that receive service at the start of the service session.

Table 3: Ranking of ASR based on average efficiency across environments

Rank	ASR	CI (%)	Type	Maverick
1	15	99.9980	IND	0.1956
2	22	99.9980	IND	0.3139
3	30	99.9980	IND	0.3891
4	23	99.9980	IND	0.2843
5	29	99.9980	IND	0.4125
6	16	99.9980	EL	0.1636
7	273	99.9980	EL	0.0667
8	162	99.9980	EL	0.1719
9	256	99.9980	EL	0.0703
10	298	99.9980	EL	0.0640
11	31	99.9980	IND	0.3640
12	7	99.9980	IND	0.3409
13	321	99.9980	EL	0.0637
14	9	99.9980	IND	0.0769
15	209	99.9980	EL	0.0953
...	...	...	...	...
23	8	99.8136	IND	0.0959

The two selected block ASR both have a block size of two and a very small (or zero) adjustment of the arrival times. With respect to the EL ASR, we select two rules that postpone the arrival rate of customers after the arrival of the first ten customers (the arrival rate of the first ten customers is not corrected). Contrary to the other selected ASR, the adjustment for service time variance is substantial for the selected EL ASR. In what follows, we first illustrate the effect of the different environmental parameters. Next, we analyze the impact of subjective valuation of the different performance measures.

6.2.1 Impact of environmental variables

In this section, we examine the impact on the performance of the ASR of (1) the number of customers during a service session, (2) the SCV of the service times, (3) the SCV of the early- and late arrival distributions and (4) the probability of no-shows.

From figure 3(a), it is clear that an increase in the number of customers that are to be served results in an increase of customer waiting time. The effect of N on the idle time of the server is similar although less outspoken and not always in the same proportion (see figure 3(b)). The effect of N on server overtime is nearly identical to its effect on server idle time. As could be expected, the EL ASR (ASR 161 and 162) have strong performance in terms

ASR	Characteristics
7	Individual ASR, one customer at session start, maximum ($h = 0.3$) adjustment for service time standard deviation
8	Individual ASR, two customers at session start, no adjustment for service time standard deviation
15	Individual ASR, three customers at session start, no adjustment for service time standard deviation
22	Individual ASR, four customers at session start, no adjustment for service time standard deviation
29	Individual ASR, five customers at session start, no adjustment for service time standard deviation
92	Block ASR, block size of two customers, no adjustment for service time standard deviation
93	Block ASR, block size of two customers, small ($h = 0.05$) adjustment for service time standard deviation
161	EL ASR, postpone arrivals after first 10 customers ($z = 10$), strong ($h = 0.25$) adjustment for service time standard deviation
162	EL ASR, postpone arrivals after first 10 customers ($z = 10$), maximum ($h = 0.3$) adjustment for service time standard deviation

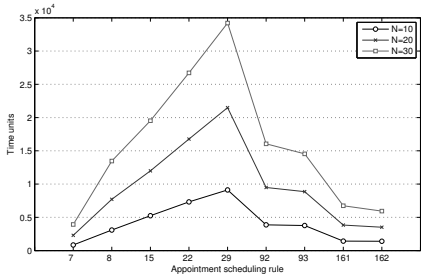
Table 4: Overview of the selected ASR

of waiting time. Both individual and block ASR are characterized by large customer waiting times. The only exception is ASR 7. However, the server idle and overtime explodes under this ASR. In general, the evaluation based on server-oriented measures (i.e., server idle time and server overtime) is fully opposite compared to the evaluation based on the customer waiting time: the individual ASR and the block ASR have low idle time (and overtime) for all levels of N. Moreover, we note that the impact of the number of customers can be significant, in some cases even larger than proportional to the increase of N which seems to favor smaller service sessions.

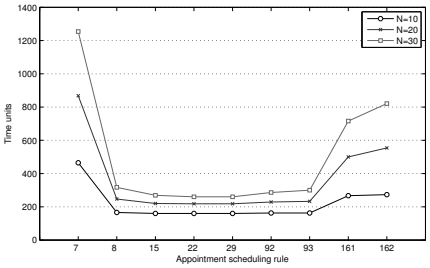
Figures 3(c) and 3(d) show the impact of the SCV of the service times on the performance of the selected ASR. An increase in service time variability has an unfavorable impact on each of the performance measures. Once more, the impact on server idle time and customer waiting time is very similar. Based on customer waiting time, the EL rules (and the individual ASR 7) are preferred. However, we observe that exactly these ASR are most influenced by an increase of the service time variability. If the server-oriented measures are key, the most suitable ASR are individual ASR 8, 15, 22 and 29.

The probability of a customer arriving too early or too late and the SCV of the deviation time only have minor influences on the performance of the ASR.

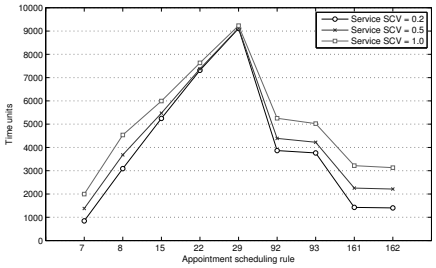
Customers not showing up cause the total waiting time for the customers


x 10^4
N=10
N=20
N=30
Time units
Appointment scheduling rule
7
8
15
22
29
92
93
161
162

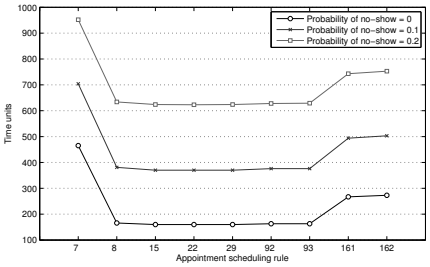
(a)


N=10
N=20
N=30
Time units
Appointment scheduling rule

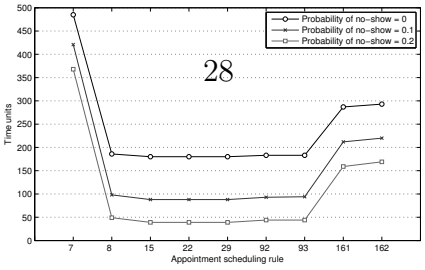
(b)


Service SCV = 0.2
Service SCV = 0.5
Service SCV = 1.0
Time units
Appointment scheduling rule

(c)


Probability of no−show = 0
Probability of no−show = 0.1
Probability of no−show = 0.2
Time units
Appointment scheduling rule

(d)


Probability of no−show = 0
Probability of no−show = 0.1
Probability of no−show = 0.2
Time units
Appointment scheduling rule

(e)

to decrease (figure 3(f)). Moreover, no-shows imply additional server idle time (figure 3(e)), while the server overtime decreases. We bring the readers attention to the fact that while the server idle time increases more or less proportional to the no-show probability (figure 3(e)), the impact on the amount of overtime decreases as the probability increases. This shows how detrimental no-shows can be. Moveover, the impact of customers not showing up is substantial whatever ASR is used, indicating the importance of avoiding no-shows. This can explain the popularity of "open-access" systems, because these systems minimize the problem of no-shows, by having only recently booked appointments. The drawback is a higher variability in the workload. This trade-off is studied by Robinson and Chen [2010], who find that open-access systems perform strongly in a wide range of situations, especially with high no-show probabilities (i.e., larger than 5%).

Based on the results discussed above, we observe that the impact of environmental variables differs depending on the selected ASR. In general, we can state that the probability of no-shows has the largest impact, the number of customers in a session and the service time variance have a significant impact. These conclusions are in line with those of Ho and Lau [1992]. Contrary, the impact of customer punctuality is limited. Moreover, from the figures above it is clear that EL rules (and ASR 7) are to be preferred with respect to customer waiting time, while individual and block ASR (8, 15, 22, 29, 161 and 162) excel in minimizing server idle and server overtime. Strikingly, is the relative balanced performance of the simple Bailey-Welch rule with two initial arrivals. The strong performance of this ASR was also noted in Ho and Lau [1992].

6.2.2 Impact of subjective valuation of performance measures

At this point, we have established which ASR work well across environments and we have determined the impact of the different environmental parameters. However, the decision makers (e.g., hospital management) may want to stress one (or more) of the performance measures. We already discussed how to incorporate such subjective valuation into the DEA model and will now discuss how the suitability of the ASR changes when either the customer waiting time or the server idle time is pivotal for scheduling purposes.

Table 5 provides the 15 best-performing ASR across all environments when the weight assigned to the average waiting time is higher than the weight of the idle time and the weight of the overtime. When comparing

tables 3 and 5, it is clear that the EL ASR dominate the performance ranking when customer waiting time is the most important criteria. The top 4 ranked ASR, are all EL ASR. Interestingly, these are exactly the same ASR that are the best-performing EL ASR without any weight restrictions (ranked 7^{th} to 10^{th} in table 3). In sharp contrast, the top 6 individual ASR in table 3 tumble down in the ranking. The strong performance of the EL rules over all environments reinforces our earlier observation that these rules perform strongly with respect to waiting time.

Surprisingly, we notice the strong performance of individual ASR with four initial customers (whereas individual ASR with a single initial customer are dominant when customer waiting time, server idle time and server overtime are considered to be equally important). However, the four initial customers are not scheduled to arrive all exactly at the start of the session but will gradually come in during the treatment of the first customer. The customer waiting time is further limited by high levels of adjustment for the service time variance. Whereas Ho and Lau [1999] conclude that an individual ASR with four initial customers leads to very high waiting times, we have shown that more advanced individual ASR in which initial customers gradually arrive during the start of the session can significantly reduce the customer waiting time. Similar to our results, the results of Ho and Lau [1999] show that EL rules dominate if customer waiting time is the most important evaluation criteria. As was expected, block ASR perform badly. ASR 94 and ASR 93 are ranked 107^{th} and 108^{th}, respectively. Lastly, the Bailey-Welch ASR performs less well when customer waiting time is important.

Albeit the benefits of waiting time minimization, the expensive equipment (and people) used in AS may favor the use of ASR that minimize idle time. Consequently, we incorporated the additional constraint that the weight allocated to the idle time should be the highest (i.e., idle time is the most important performance criteria). With this constraint on the weight selection, the individual ASR return to the top rankings, pushing back the EL ASR (see table 5). All of the top six ranked individual ASR from table 3 reappear in the top 15 and make up the four best-ranked ASR when idle time is the most important evaluation criteria.

Even more consistency is found in the best-performing EL ASR: ASR 273, 162, 256, 298, 321 and 209 are part of the top 15 ranked ASR across all environments in the case without weight restrictions, as well as with waiting time or with idle time as the most important performance measure. This let us to belief that these ASR are especially suitable for a wide range of

Table 5: Ranking of ASR across environments when performance measures are not considered to be equally important

Waiting time is most important					Idle time is most important				
Rank	ASR	CI (%)	Type	Maverick	Rank	ASR	CI (%)	Type	Maverick
1	273	99.972	EL	0.0508	1	15	99.998	EL	0.1896
2	162	99.965	EL	0.1150	2	23	99.991	EL	0.2758
3	256	99.945	EL	0.0533	3	16	99.984	EL	0.1579
4	298	99.918	EL	0.0544	4	22	99.974	EL	0.3039
5	7	99.888	IND	0.2736	5	273	99.972	IND	0.0624
6	321	99.887	EL	0.0535	6	162	99.965	EL	0.1516
7	209	99.872	EL	0.0734	7	256	99.945	EL	0.0658
8	84	99.856	IND	0.1927	8	298	99.918	IND	0.0612
9	208	99.846	EL	0.0628	9	30	99.892	EL	0.3736
10	83	99.843	IND	0.1508	10	321	99.887	IND	0.0608
11	163	99.835	EL	0.1385	11	9	99.883	EL	0.0700
12	82	99.834	IND	0.1145	12	209	99.872	IND	0.0896
13	250	99.825	EL	0.0514	13	84	99.855	EL	0.2322
14	179	99.784	EL	0.0876	14	29	99.855	EL	0.3957
15	161	99.783	EL	0.0944	15	51	99.850	EL	0.0995
...	...	...	...	...	...	...	...	...	...
37	8	99.537	IND	0.0527	22	8	99.811	IND	0.0899

environments and operating conditions. In line with the previous observations, the block ASR perform badly (the best-ranked block ASR (ASR 93) is ranked 72^{th}).

7 Conclusion

Appointment scheduling rules (ASR) are used to determine the point in time at which a customer is to receive service during a service session. ASR are commonly applied in service and manufacturing industries (e.g., healthcare or after sales service).

We develop an analytical model that uses a Discrete Time Markov Chain and an efficient algorithm to assess the performance (in terms of customer waiting time, server idle time and server overtime) of ASR in a wide variety of settings. More specifically, the model takes into account the following environmental variables: (1) customer unpunctuality, (2) no-shows, (3) service interruptions and (4) delay of the service process. The validity and accuracy of the model are verified using a simulation study. We use the model to assess the performance of 314 ASR and use data envelopment analysis to compare results.

In general, we conclude that individual ASR have a superior performance

across all environments. EL rules, however, hardly underperform and often have a performance that depends less on the valuation of the different performance measures. In addition, EL ASR outperform individual ASR if customer waiting time is considered to be the most important performance measure. We are able to identify a group of six EL ASR that appear in the top 15 of best-performing ASR in the following value-judgement scenarios: (1) all performance measures are equally important, (2) customer waiting time is the most important measure and (3) server idle time is the most important measure. The performance of block ASR is dismal and the use of such rules should be avoided as better alternatives exist.

With respect to the environmental variables, we conclude that the probability of no-show has the most severe impact on ASR performance. The number of customers served, as well as the variability of the service itself, have a large impact on the performance of the scheduling rules. Customer unpunctuality, however, only has a minor impact.

References

R. Allen and E. Thanassoulis. Improving envelopment in data envelopment analysis. *European Journal of Operational Research*, 154(2):363–379, 2004.

A. Apte, U. M. Apte, and N. Venugopal. Focusing on customer time in field service: A normative approach. *Production and Operations Management*, 16(2):189–202, 2007.

M. Babes and G.V. Sarma. Out-patient queues at the Ibn-Rochd health centre. *The Journal of the Operational Research Society*, 42(10):845–855, 1991.

N.T. Bailey. A study of queues and appointment systems in hospital outpatient departments, with special reference to waiting-times. *Journal of the Royal Statistical Society, Series B*, 14(2):185–199, 1952.

D. Biskup, J. Herrmann, and J.N.D. Gupta. Scheduling identical parallel machines to minimize total tardiness. *International Journal of Production Economics*, 115:134–142, 2008.

M.J. Blanco White and M.C. Pike. appointment systems in out-patients'

clinics and the effect of patients' unpunctuality. *Medical Care*, 2(3):133–145, 1964.

B. Cardoen, E. Demeulemeester, and J. Belien. Sequencing surgical cases in a day-care environment: An exact branch-and-price approach. *Computers & Operations Research*, 36(9):2660–2669, 2009.

T. Cayirli and E. Veral. Outpatient scheduling in health care: a review of literature. *Production and Operations Management*, 12(4):519–549, 2003.

Santanu Chakraborty, Kumar Muthuraman, and Mark Lawley. Sequential clinical scheduling with patient no-shows and general service time distributions. *IIE Transactions*, 42(5):354–366, 2010.

L. Cherchye, W. Moesen, N. Rogge, T. Van Puyenbroeck, M. Saisana, A. Saltelli, R. Liska, and S. Tarantola. Creating composite indicators with dea and robustness analysis: the case of the technology achievement index. *Journal of the Operational Research Society*, 59(2):239–251, 2008.

Liao C.J., C.D. Pegden, and M. Rosenshine. Planning timely arrivals to a stochastic production or service system. *IIE Transactions*, 25(5):63–73, 1993.

W. D. Cook and L. M. Seiford. Data envelopment analysis (dea) - thirty years on. *European Journal of Operational Research*, 192(1):1–17, 2009.

W. W. Cooper, J. L. Ruiz, and I. Sirvent. Choosing weights from alternative optimal solutions of dual multiplier models in dea. *European Journal of Operational Research*, 180(1):443–458, 2007.

S. Creemers. *Appointment-driven queueing systems*. PhD thesis, Department of Decision Sciences & Information Management, K.U. Leuven, 2009a.

Stefan Creemers and Marc Lambrecht. An advanced queueing model to analyze appointment-driven service systems. *Computers & Operations Research*, 36(10):2773–2785, 2009.

Stefan Creemers and Marc Lambrecht. Queueing models for appointment-driven systems. *Annals of Operations Research*, 178(1):155–172, 2010.

N.P. Dellaert and M.T. Melo. Make-to-order policies for a stochastic lot-sizing problem using overtime. *International Journal of Production Economics*, 56-57:79–97, 1998.

J. Doyle and R. Green. Efficiency and cross-efficiency in dea - derivations, meanings and uses. *Journal of the Operational Research Society*, 45(5): 567–578, 1994.

R.B. Fetter and J.D. Thompson. Patients' waiting time and doctors' idle time in the outpatient setting. *Health Services Research*, 1(1):66–90, 1966.

B.E. Fries and V.P. Marathe. Determination of optimal variable-sized multiple-block appointment systems. *Operations Research*, 29(2):324–345, 1981.

G. Giuliano and T. O'Brien. Reducing port-related truck emissions: the terminal gate appointment system at the ports of Los Angeles and Long Beach. *Transportation Research Part D*, 12:460–473, 2007.

V.L. Green. Using operations research to reduce delays for healthcare. *Tutorials in Operations Research INFORMS*, DOI 10.1287/educ.1080.0049, 2008.

K.D. Grote, J.R. Newman, and S.S. Sutaria. A better hospital experience. *The McKinsey Quarterly*, November:1–11, 2007.

Diwakar Gupta and Brian Denton. Appointment scheduling in health care: Challenges and opportunities. *IIE Transactions*, 40(9):800–819, 2008.

C. J. Ho and H. S. Lau. Minimizing total-cost in scheduling outpatient appointments. *Management Science*, 38(12):1750–1764, 1992.

C. J. Ho and H. S. Lau. Evaluating the impact of operating conditions on the performance of appointment scheduling rules in service systems. *European Journal of Operational Research*, 112(3):542–553, 1999.

van Leeuwaarden J., D. Denteneer, and J. Resing. A discrete-time queueing model with periodically scheduled arrival and departure slots. *Performance Evaluation*, 63:278–294, 2006.

B. Jansson. Choosing a good appointment system - a study of queues of the type (D,M,1). *Operations Research*, 14(2):292–312, 1966.

O. Jouini and S. Benjaafar. Queueing systems with appointment-driven arrivals, non-punctual customers and no-shows. *Working Paper, University of Minnesota, College of Science & Engineering*, May:26, 2010.

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

K.J. Klassen and T.R. Rohleder. Scheduling outpatient appointments in a dynamic environment. *Journal of Operations Management*, 14(2):83–101, 1996.

M. Lawley, V. Parmeshwaran, J.P. Richard, A. Turkcan, M. Dalal, and D. Ramcharan. A time-space scheduling model for optimizing recurring bulk railcar deliveries. *Transportation Research Part B*, 42:438–454, 2008.

B. Lehaney, S.A. Clarke, and R.J. Paul. A case of an intervention in an outpatients department. *The Journal of the Operational Research Society*, 50(9):877–891, 1999.

J. Lian, K. Distefano, S. Shields, C. Heinichen, M. Giampietri, and L. Wang. Clinical appointment process, improving through schedule defragmentation. *IEEE Engineering in Medicine and Biology Magazine*, March/April: 127–134, 2010.

L. Liu and X. Liu. Block appointment systems for outpatient clinics with multiple doctors. *The Journal of the Operational Research Society*, 49(12): 1254–1259, 1998a.

L. Liu and X. Liu. Dynamic and static job allocation for multi-server systems. *IIE Transactions*, 30(9):845–854, 1998b.

Nan Liu, Serhan Ziya, and Vidyadhar G. Kulkarni. Dynamic scheduling of outpatient appointments under patient no-shows and cancellations. *M&SOM-Manufacturing & Service Operations Management*, 12(2):347–364, 2010.

M.A. Madas and K.G. Zografos. Airport slot allocation: from instruments to strategies. *Journal of Air Transport Management*, 12(2):53–62, 2006.

M.A. Madas and K.G. Zografos. Airport capcity vs. demand: mismatch or mismanagement? *Transportation Research Part A*, 42:203–226, 2008.

A. Mercer. A queueing problem in which the arrival times of the customers are scheduled. *Journal of the Royal Statistical Society, Series B (Methodological)*, 22(1):108–113, 1960.

A. Mercer. Queues with scheduled arrivals: a corretion, simplification and extension. *Journal of the Royal Statistical Society, Series B (Methodological)*, 35(1):104–116, 1973.

S. Mondschein and G.Y. Weintraub. Appointment policies in service operations: a critical analysis of the economic framework. *Production and Operations Management*, 12(2):266–286, 2003.

R. Namboothiri and A.L. Erera. Planning local container drayage operations given a port access appointment system. *Transportation Research Part E*, 44:185–202, 2008.

O. B. Olesen and N. C. Petersen. Indicators of ill-conditioned data sets and model misspecification in data envelopment analysis: An extended facet approach. *Management Science*, 42(2):205–219, 1996.

C.D. Pegden and M. Rosenshine. Scheduling arrivals to queues. *Computers and Operations Research*, 17(4):343–348, 1990.

Mcas Portela and E. Thanassoulis. Zero weights and non-zero slacks: Different solutions to the same problem. *Annals of Operations Research*, 145: 129–147, 2006.

E.J. Rising, R. Baron, and B. Averill. A systems analysis of a university-health-service outpatient clinic. *Operations Research*, 21(5):1030–1047, 1973.

Lawrence W. Robinson and Rachel R. Chen. A comparison of traditional and open-access policies for appointment scheduling. *M&Som-Manufacturing & Service Operations Management*, 12(2):330–346, 2010.

T.R. Rohleder and K.J. Klassen. Using client-variance information to improve dynamic appointment scheduling performance. *Journal of Operations Management*, 28(3):293–305, 2000.

T.R. Rohleder and K.J. Klassen. Rolling horizon appointment scheduling: a simulation study. *Health Care Management Science*, 5:201–209, 2002.

C. Rose and R. Yates. Scheduling arrivals to queues for minimum average blocking: the $s(n)/m/c/c$ system. *Computers and Operations Research*, 22(8):793–806, 1995.

F. Sabria and C.F. Daganzo. Approximate expressions for queueing systems with scheduled arrivals and established service order. *Transportation Science*, 23(3):159–165, 1989.

A. Soriano. Comparison of two scheduling systems. *Operations Research*, 14 (3):388–397, 1966.

J.R. Swisher, S.H. Jacobson, J.B. Jun, and O. Balci. Modeling and analyzing a physician clinic environment using discrete-event (visual) simulation. *Computers and Operations Research*, 28(2):105–125, 2001.

V. Tardif and M.L. Spearman. Diagnostic scheduling in finite-capacity production environments. *Computers and Industrial Engineering*, 32(4):867–878, 1997.

P.M. Vanden Bosch and D.C. Dietz. Scheduling and sequencing arrivals to an appointment system. *Journal of Service Research*, 4(1):15–25, 2001.

P.M. Vanden Bosch and D.C. Dietz. Minimizing expected waiting in a medical appointment system. *IIE Transactions*, 32:841–848, 2002.

J.M.H. Vissers. Selecting a suitable appointment system in an outpatient setting. *Medical Care*, 17(12):1207–1220, 1979.

P.P. Wang. Static and dynamic scheduling of customer arrivals to a single-server system. *Naval Research Logistics*, 40:345–360, 1993.

P.P. Wang. Optimally scheduling n customer arrival times for a single-server system. *Computers and Operations Research*, 24(8):703–716, 1997.

E.N. Weiss. Models for determining estimated start times and case orderings in hospital operating room. *IIE Transactions*, 22(2):143–150, 1990.

J. Welch and N. Bailey. Appointment systems in hospital outpatient departments. *The lancet*, 319:1105–1108, 1952.

J.D. Welch. Appointment systems in hospital outpatient departments. *Operations Research Quarterly*, 15(3):224–232, 1964.

E. Wendler. The scheduled waiting time on railway lines. *Transportation Research Part B*, 41:148–158, 2007.

S. Yan and W. Lai. An optimal scheduling model for ready mixed concrete supply with overtime considerations. *Automation in Construction*, 16:734–744, 2007.