MANAGEMENT SCIENCE
Vol. 00, No. 0, Xxxxx 0000, pp. 000–000
ISSN 0025-1909 | EISSN 1526-5501 | 00 | 0000 | 0001

INFORMS
DOI 10.1287/xxxx.0000.0000

Data-driven appointment-scheduling under uncertainty: The case of an infusion unit in a cancer center

Avishai Mandelbaum, Petar Momčilović, Nikolaos Trichakis
Technion – Israel Institute of Technology, University of Florida, Massachusetts Institute of Technology

Sarah Kadish, Ryan Leib, Craig A. Bunnell
Dana-Farber Cancer Institute

Service systems are often stochastic and pre-planned by appointments, yet implementations of their appointment systems are prevalently deterministic. At the planning stage of healthcare services, for example, customer punctuality and service durations are often assumed equal their means—and this gap, between planned and reality, motivated our research. Specifically, we consider appointment scheduling and sequencing under a time-varying number of servers, in a data-rich environment where service durations and punctuality are uncertain. Our data-driven approach, based on infinite-server queues, yields tractable and scalable solutions that accommodate hundreds of jobs and servers. We successfully test our approach against near-optimal algorithms (which exist for merely single-servers). This entails the development of a data-driven robust optimization approach with novel uncertainty sets. To test for practical performance, we leverage a unique dataset from a cancer center, that combines real-time locations, electronic health records and appointments log. Focusing on one of the center's infusion units (∼90 daily appointments, 25+ infusion chairs), we reduce cost (waiting+overtime) in the order of 15%-40% consistently, under a wide range of experimental setups.

Key words: Appointment scheduling; appointment book; data-driven decision making; scheduling under uncertainty

1. Introduction

Appointment scheduling is a ubiquitous operational process in service systems. In Healthcare Delivery Systems (HDS), for example, the majority of services operate under scheduled appointments, ranging from primary care exams to surgeries. Being essentially the process of matching supply and demand, appointment scheduling is typically a key performance driver: surplus of supply, *i.e.*, too "few" appointments over a period of time, leads to low server utilization; surplus of demand, *i.e.*, too "many" appointments, leads to customer delays. Such mismatches can have profound negative

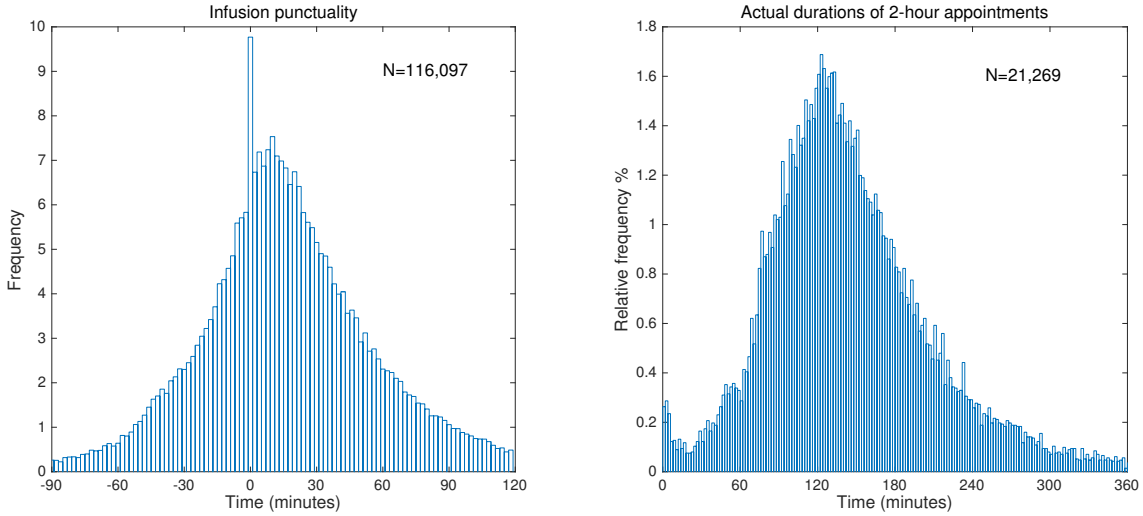
consequences in settings where server idling and customer waiting are both costly—for example, in HDS, where servers could correspond to specialized physicians or equipment, and customers to patients. From a methodological standpoint, scheduling (a given number of) appointments, the service durations of which are stochastic, is a notoriously difficult problem. The main reason is that its analysis requires tracking transient performance measures, which renders standard queueing theory inapplicable (Kong et al. 2013). Consequently, solution approaches have so far been developed only for specialized cases; notably for settings where service durations and punctuality are deterministic (Pinedo 2009, Santibáñez et al. 2012), or for settings where service durations are uncertain, but where only a single server is in operation (Denton and Gupta 2003).

The majority of HDS, however, follow modes of operation that lie beyond the specialized cases amenable to the existing scheduling algorithms. First, they tend to operate by *sharing multiple servers*. In particular, for cost-cutting and efficiency purposes, there is a recent consolidation trend in the healthcare industry, with large-scale HDS increasingly pooling or sharing their resources, instead of operating specialized and dedicated centers (Dafny et al. 2012, Bravo et al. 2017). For example, in oncology outpatient care, Massachusetts General Hospital (MGH) recently opened a new adult cancer center operating a sixty bed/chair infusion unit; similarly, Dana-Farber Cancer Institute (DFCI) built a new cutting-edge clinical and research facility (the Yawkey Center), where infusion chairs are shared among different disease centers. Second, state-of-the-art clinical treatment processes are highly variable and subject to *inherent uncertainty*, owing to myriads of physiological factors and/or emergence of complex and personalized clinical pathways. For instance, infusion appointments in oncology are contingent on the patient's evolving health status, resulting in their durations being variable and unpredictable (Barysauskas et al. 2016). (We elaborate on this in Section 3.)

In reality, there exists a significant discrepancy between the planned and actual operations, even at HDS that employ sophisticated scheduling systems and state-of-the-art practices (*e.g.*, DFCI). As a consequence, patients face lengthy waiting times, with their experience often being very different from what was scheduled. This mismatch can be traced back to the aforementioned assumptions of deterministic service times and punctuality—assumptions routinely made in appointment scheduling implementations, which are capable of handling large-scale operations that involve hundreds of servers and hundreds of customers. To this end, consider DFCI's infusion appointment operations: Figure 1 depicts histograms of appointment *punctuality* and *actual* duration of appointments; the latter were scheduled for two hours, but their actualization illustrates significant variability.

Figure 1 offers yet another perspective: it was compiled using high-resolution data from a *Real-Time Locating System* (RTLS) deployed at DFCI. Our RTLS tracks both patients (∼850 per day) and providers (∼300 per day) across DFCI's Ambulatory Cancer Center, in a continuous

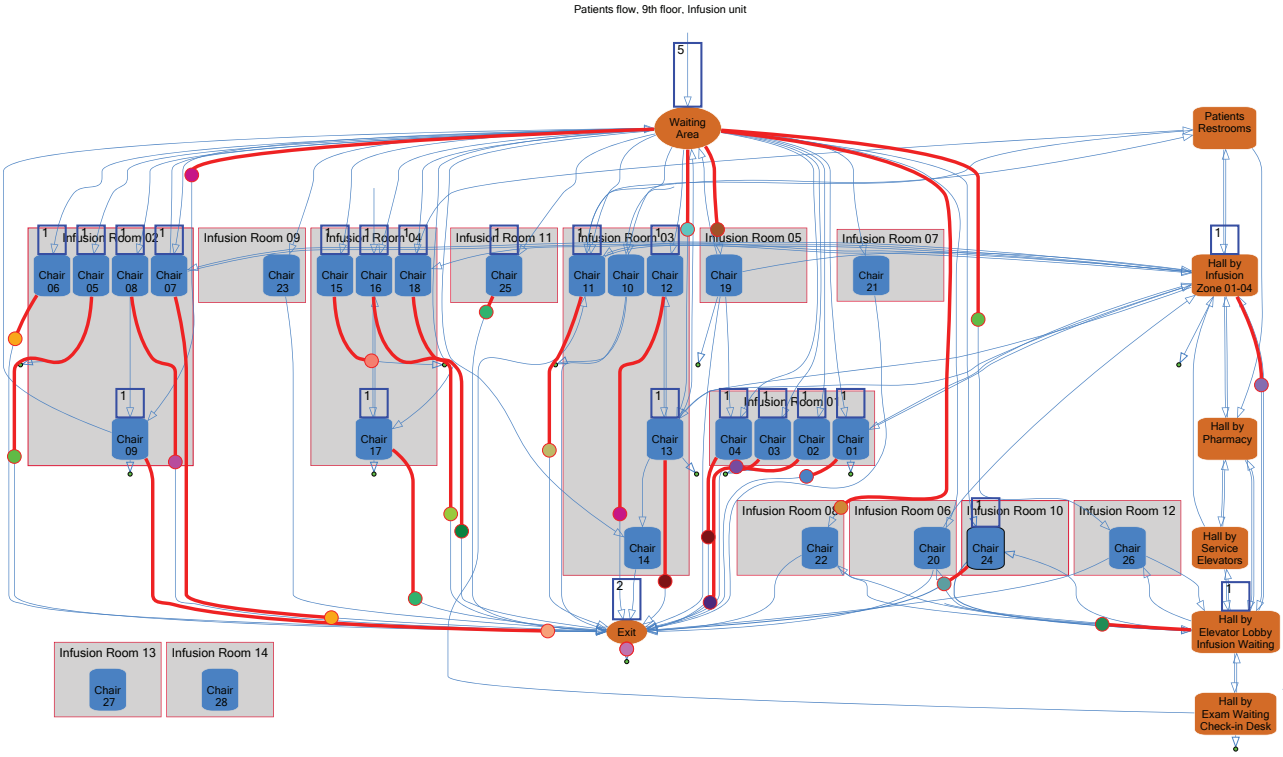
Figure 1 Histograms of punctuality of infusion treatments (left) and of actual durations of infusion treatments scheduled for two hours (right). The graphs are based on data for all DFCI business days between November 2013 and May 2015.



and fully-automated fashion. Figure 2 illustrates a snapshot of such patient location data collected at the infusion unit of the ninth floor, at 11:17 on XX-YYY-2014—we do not report the exact date due to privacy considerations. An animation of the entire day's data is accessible at youtu.be/e1qHeYg7hfw. RTLS enables a unique data-gathering process of *large-scale* operational data that are considerably less susceptible to observational bias. For more details on our RTLS data, see the discussion in Section 3. DFCI's RTLS implementation is aligned with the general trend of HDS increasingly collecting data at a massive scale. All these data can be harnessed to improve operational processes, and our work is one initial step in this direction.

In this paper, we deal with the so-called *offline appointment sequencing problem*, where one must decide on the appointment times for a given set, or number of jobs—which have random durations—so as to minimize a combination of the expected wait time of jobs and the overtime of servers. Equivalently, one needs to decide on the sequence in which jobs will be scheduled and the time allowances between successive jobs. Problems in which the sequence of jobs is given and the decision only revolves around the time allowances between successive jobs are referred to as *appointment scheduling* problems. From a practical standpoint, the problem we study here arises, for example, at service systems that manage appointment requests in two steps: in the first step that occurs several weeks or days before service, each incoming appointment request is booked for a particular day/time-window in the future; in the second step, only a few days before service, an exact time is set for all appointments that were booked for a particular day in the first step. This two-step approach is well accepted both in practice (Mak et al. 2015) and in the literature, where the second step precisely corresponds to the (offline) appointment sequencing problem.

Figure 2 Snapshot of our RTLS data, namely patient location at DFCI's Yawkey Center ninth floor infusion unit at 11:17 on XX-YYY-2014. An animation of the entire day is available at youtu.be/e1qHeYg7hfw. Nodes represent specific locations, *e.g.*, infusion chairs and hallways. Patients represented by dots traversing edges are obtaining service at the originating node and reach the destination node upon completion—for example, the patient traversing the edge emanating from Infusion Chair 25 in Room 11 is currently undergoing infusion treatment in that chair.



In the present work, we focus on the offline appointment sequencing problem for relatively large service systems. Our systems consist of a (potentially time-varying) large number of servers, who cater to a large number of customers; and the systems are stochastic in the sense that both service-durations and (customer) punctuality are subject to significant uncertainty. Our motivating example is infusion services at DFCI, where a disease center consists of 10's of chairs (servers) that accommodate over 50 patients (customers) per day. Methodologically, as we argued already, the appointment sequencing problem is notoriously difficult (Kong et al. 2013). It becomes even more challenging in multi-server settings in which jobs are served by shared servers, as opposed to, for example, being pre-assigned to one particular server; this is because, in general, sharing of servers leads to complex queueing dynamics that cannot be expressed in closed form (*e.g.*, à la Lindley recursion for a single server).

Contributions

We develop a novel, data-driven approach to solve the appointment sequencing problem; we accommodate a time-varying number of *shared* servers, in a tractable and scalable fashion, for problem

Table 1 Applicability of existing approaches in the literature (SO (Denton and Gupta 2003) and DR (Kong et al. 2013, Mak et al. 2015)), our DDR approach and our IS approach in terms of number of servers, and the ability to capture sequencing or punctuality.

	Number of shared servers			Allows for	
	single	few ($e.g.$ $2-15$)	many ($e.g.$ ≥ 15)	Sequencing	Punctuality
Existing approaches: SO, DR	✓	✗	✗	✓	✗
DDR	✓	✓	✓	✗	✗
IS	✗	✗	✓	✓	✓

instances with hundreds of jobs and hundreds of servers. Our approach relies on an infinite-server relaxation, which accounts for uncertainty in service times and punctuality while usefully (*e.g.*, Figures 4 and 6) modeling the complex queueing dynamics of shared-server environment; we refer to it as an *Infinite-Server* (IS) sequencing approach (Section 5), and to its underlying model as the IS-model or simply IS (Section 5.1). Furthermore, we utilize a CLT approximation for computational purposes. These relaxations render our approach applicable to problems involving a large enough number of servers.

We perform an extensive benchmarking of our IS approach in three steps, and find that it yields very strong performance with significant reductions in average wait time and overtime cost. These reductions could be as large as 40%-60% in large-scale HDS such as DFCI, when compared against appointment sequencing implementations usually deployed in such systems. We elaborate on the three steps next.

First, in Section 6 we benchmark IS against certain near-optimal solution approaches established in the literature, specifically Stochastic Optimization (SO) (Denton and Gupta 2003) and Distributionally Robust (DR) approaches (Kong et al. 2013, Mak et al. 2015). As it happens, these state-of-the-art approaches are applicable only to single-server environments: Table 1 provides a schematic illustration of their scope and limitations. To enable a comparison, nevertheless, we develop another solution approach, dubbed *Data-Driven Robust* (DDR) that, loosely speaking, lies "in-between": DDR leverages the tractability of robust optimization and enables one to explicitly model the queueing dynamics for multiple, shared servers (see Table 1), while also leveraging the wealth of operational data by relying on novel uncertainty sets. In our numerical studies, we then compare DDR with near-optimal solution approaches in a single-server setting and find that it performs equally well (if not better). This supports comparison of IS with DDR in a multi-server environment, which reveals that IS performs equally well as (if not better than) DDR.

Second, in Section 7.1 we benchmark IS against appointment sequencing algorithms that treat service times and punctuality as deterministic (*e.g.*, equal to their mean values), which is the *modus operandi* of virtually all implementations in practice (Berg and Denton 2012); we refer to such algorithms as *means-based sequencing*. We conduct our analysis using DFCI data: as already

mentioned, this is a unique real-world data set, of unprecedented resolution, which draws data from both RTLS and EHR systems. To the best of our knowledge, ours is the first work to use RTLS log data (at such scale) to extract operational data. Using DFCI data allows one to evaluate the benefits that our work could yield for offline appointment sequencing in a realistic setting. Compared to means-based sequencing, we find that IS provides considerable wait time and overtime cost reductions, in the order of 30% consistently, under a wide range of experimental setups. Our approach is entirely data-driven, with data playing a central role in every step: calibration, validation, testing, and benchmarking.

Third, in Section 7.2 we benchmark IS with means-based sequencing in the context of optimizing DFCI's infusion appointment process, which deviates from but closely resembles offline sequencing. In particular, at DFCI, templates determine the times-of-day that appointments are scheduled for, based on patient's type, namely disease and infusion type. Templates are computed offline based on the anticipated number of requests and patient types—the actual request patterns usually differ, although not by too much. We use our IS and a means-based approach, which mimics DFCI practice, to produce templates. Then, we use our unique data to simulate 34 real historical days, in which stochastic appointment requests arrive sequentially in a random order and are scheduled based on the IS- or the means-based-computed template. In comparison with means-based, the IS-computed template yields a cost reduction of approximately 55%. This significant improvement of IS over means-based sequencing, in our realistic experiments, illustrates IS robustness, its potential to be used in practice, and that it could also serve as a useful building block in the design of online appointment algorithms.

Managerial Insights

Does variability "average out" in multi-server appointment systems? In particular, one might hypothesize that in such systems, where a large number of jobs are processed simultaneously by shared servers, delays from jobs that run over their expected service time would be offset by jobs that finish earlier, to the extent that stochasticity would cancel out. Such a hypothesis could rationalize the use of means-based scheduling and sequencing algorithms in practice. Our work, however, refutes this hypothesis, informing HDS managers of the significant gains—overtime plus waiting time reductions in the order of 30%—that appointment processes, which do account for uncertainty, can bear over means-based algorithms in multi-server environments.

RTLS implementations have been very recent and it is not yet well understood how to make the best of such systems and their data (neither in an offline nor an online fashion). RTLS systems avert many observational biases that plague other data-gathering processes (see Section 3). Hence another insight that our work affords is that it showcases a way to utilize RTLS data to improve operational procedures, such as appointment scheduling and sequencing.

More broadly, RTLS could better the efficient use of fixed resources (*e.g.*, exam room space), which is essential to enable clinical growth within existing facilities. Indeed, adding or expanding a facility requires significant financial resources and time, likely years, to implement. RTLS could support methods that optimize schedules, reduce patient wait time, and distribute clinical volume to avoid unsustainable volume peaks. This would lead to a more efficient use of fixed resources and extend the time horizon before demand consistently exceeds available capacity and thus new facilities are required.

2. Brief Literature Review

The literature on appointment scheduling and sequencing is vast and has dealt with multiple facets of the problem. Excellent surveys are provided by Cayirli and Veral (2003), Gupta and Denton (2008), Cardoen et al. (2010) and Ahmadi-Javid et al. (2017). Outpatient Procedure Center sequencing is reviewed in Berg and Denton (2012), where it was remarked that, in practice, schedules are commonly based on mean procedure times. Trading of customer waiting vs. server idleness, in the context of appointment systems, is a classical problem that can be traced back to Bailey (1952), Jackson (1964), White and Pike (1964). Recent works on appointment sequencing (e.g., see Zacharias and Pinedo (2014), Zacharias and Armony (2017), Zacharias and Pinedo (2017), including Santibáñez et al. (2012), Gocgun and Puterman (2014), Dunn et al. (2017) that focus on cancer care) consider models with various features (e.g., deadlines) that capture uncertainty of future appointment requests (stochastic arrivals, urgent/elective procedures, cancellations). However, these studies do not address uncertainty in punctuality and treatment durations, *i.e.*, deterministic service times and perfect punctuality is assumed. Below, we only discuss work that is directly related to ours.

Work on appointment sequencing and scheduling under uncertainty has almost exclusively dealt with the case where a single server processes all jobs. Another prevalent assumption is that jobs arrive precisely at their scheduled times (perfect punctuality). For the single-server scheduling problem, Denton and Gupta (2003) employ a stochastic linear programming formulation and develop a variant of the standard L-shaped algorithm (*e.g.*, Chapter 5 in Birge and Louveaux (1997)) to obtain optimal solutions. Kaandorp and Koole (2007) show that, under independent and exponentially distributed job durations, the objective is L# convex. Using this property, they develop a local search algorithm guaranteed to converge to the optimal schedule. Begen and Queyranne (2011) are the first to show that, under a general joint discrete probability distribution, the single-server scheduling problem is solvable in polynomial time, although their results are of a theoretical nature and no numerical solutions are presented.

The single-server sequencing problem appears to be intractable (Mak et al. 2015). Several complexity results have been derived (*e.g.*, Mancilla and Storer (2012)), including recent work by

Kong et al. (2016), which proves the sequencing problem to be NP-hard even under fixed time allowances. Not surprisingly hence, no exact solution methods are known. A popular heuristic is to sequence jobs in increasing order of variance (OV), which was also found to produce "good" sequences numerically. Optimality of OV has only been shown for two jobs by Weiss (1990) and Denton et al. (2007), assuming independent job durations. The recent work by Kong et al. (2016), however, argues that OV is very unlikely to be optimal for a general number of jobs.

Due to lack of data, required to fit credible service time probability distributions, a recent stream of papers proposes the use of robust optimization (*e.g.*, see Ben-Tal et al. (2009)) to minimize the worst-case waiting and overtime costs. Kong et al. (2013) deal with the scheduling problem and show that, under a service time uncertainty model that uses the complete covariance matrix, it can be solved to optimality as a convex optimization problem. Using a different model of uncertainty that relies on knowledge of only marginal moments, instead of the complete covariance matrix, Mak et al. (2015) provide tractable conic programs for the robust appointment sequencing problem. We re-emphasize that all the above pertains to *single*-server models.

To our knowledge, when jobs are served by shared multiple servers, and are subject to random durations and punctuality, no optimal solution methods exist. In particular, by relying on Lindley's recursion to express wait times of jobs, none of the aforementioned approaches extend to the case where jobs are served by shared servers. In addition, a direct application of stochastic programming techniques does not scale to practical problem sizes (Castaing et al. 2016).

A handful of recent papers deal with multi-server settings and study how to *pre-assign* each job to a particular server, which subsequently operates separately and serves only its pre-assigned jobs. In these settings, a job cannot be served by an arbitrary available server, *i.e.*, there is no resource pooling. To deal with this job-to-server pre-assignment problem, chance-constrained optimization (Deng and Shen 2016, Deng et al. 2017) or load-balancing heuristics are employed (Mak et al. 2014). Our work differs by considering shared-server operations. We are not aware of any work that provides a tractable (approximate) solution approach for the multi-server scheduling or sequencing problems, in which a (potentially time-varying) number of servers are being shared to serve jobs with uncertain punctualities and service times.

Finally, we mention a line of data-driven research (Kim et al. 2015, 2017) where the data originates from an appointment-driven system. Their approach is descriptive in that they leverage data for developing models of system characteristics (patients arrivals), while their appointment system is fixed. Our approach, on the other hand, is prescriptive in that our models seek to improve the driving appointments-system itself. To be specific, Kim et al. (2015, 2017) use data from an appointment system in a Korean outpatient endocrinology clinic, in order to model patients arrivals. Their models capture variability relative to the underlying appointments which, for example, is

due to changes by patients, doctors or inherent in the system itself. To put our research in this perspective, we propose a data-driven model for the occupancy process arising from an appointment scheduling/sequencing system, and our model supports improvements of the latter; to recapture the contributions, our data-source is very large, models are analytically-tractable and likely to be robust, prescriptions are well-validated and they yield significant reductions in patients delay and doctors overtime.

3. DFCI Infusion Operations and Data

In this section, we briefly describe the operational practices at DFCI, which are very similar to those employed in peer leading cancer centers, for example, Massachusetts General Hospital (Rieb 2015). Inevitably, due to space limitations, we omit many details, since processes at large-scale cancer centers are complex. Rather, we focus on the infusion operations and the unique operational RTLS+EHR data gathering processes at DFCI.

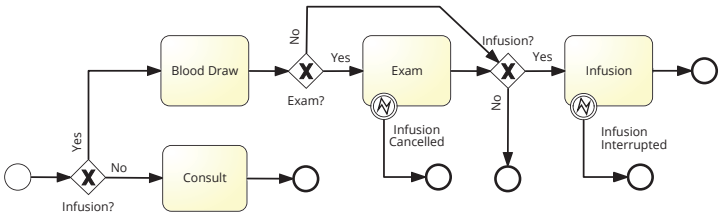
DFCI cares for approximately 1,000 cancer outpatients on a daily basis. Their majority receive care at the Yawkey Center, a state-of-art 14-story building with more than 100 exam rooms and 150 infusion spaces, which are spread over eight of the floors. The center is organized into multiple "disease centers" that specialize in specific cancer types. Depending on their size, several disease centers can be co-located on a single floor and share exam rooms and infusions spaces.

The focus of our work is on DFCI's infusion operations. Patient flow is schematically shown in Figure 3. Infusion patients visit the center according to an appointment schedule, either by having a same day exam scheduled prior to their infusion (linked appointments), or not (unlinked appointments). We elaborate on DFCI's appointment system in Section 7.2. Currently, for appointment setting, infusion durations are determined by historical averages, for patients who followed the same treatment protocol in the past. Actual appointment durations, however, differ from scheduled ones significantly—see, for example, Figure 1. This is because, for most patients, infusion protocols are adjusted according to their updated clinical conditions, which are determined by their blood draw results and physician exam. These same-day changes in treatment plans, *e.g.*, addition, change, or removal of procedures, make actual infusion durations highly uncertain and variable. Other sources of variability include the heterogeneity of patients' clinical condition and clinical trials, with the latter typically requiring non-standardized treatments. Consequently, the infusion process is usually the bottleneck in DFCI's operations.

Real-time Locating System: A Data Goldmine

A key feature of the Yawkey Center is that it has an RTLS system installed. This system consists of a network of more than 900 sensors on the ceilings of eight clinical floors, two non-clinical floors and the parking garage. The sensors use infrared to track badges worn by patients, providers and

Figure 3 Scheme of the main process at DFCI.



administrators. Badge locations are recorded every three seconds. On average, approximately 850 daily patient visits are recorded by the RTLS system. RTLS compliance in the infusion units is excellent, facilitated by nurses confirming that patients are properly badged prior to treatment.

A unique feature of the RTLS data-gathering process is that it is fully automated and requires no effort by users to record activities. This automation alleviates an important deficiency of classical solely-EHR-based implementations, which typically require some manual user input. Indeed, manual data collection often suffers from observational bias because, during busy periods, manual tasks, including data logging, are frequently overlooked or not performed on-time.

As a part of our work, RTLS data from the eight clinical floors were "synchronized" with appointment book data, as well as data from the in-house pharmacy. Having access to all these logs, enables us to accurately determine not only locations of patients, but also the activities in which they are engaged. An animation of patients, undergoing infusion activities on XX-YYY-2014 at DFCI, is accessible at youtu.be/e1qHeYg7hfw (see also Figure 2). The resulting data set provides a representation of infusion operations at DFCI that is of unprecedented fidelity and accuracy, and this motivates our data-driven approach.

4. Appointment Scheduling and Sequencing Problems

There are n customers to be scheduled for service during a time interval $[0,\bar{T}]$. For exposition purposes, we consider time to be continuous; our solution approach can be readily employed both for continuous and discrete time. The service facility has c_t available servers at time $t \in [0,\bar{T}]$. Regular business hours of the facility are $[0,T]$, $T \le \bar{T}$; any work during the time interval $(T,\bar{T}]$ counts as overtime, which incurs cost of γ per server per unit of time.

The planner needs to choose appointment times a_i, for each customer $i = 1,\ldots,n$. The quantities $a_1,\ldots,a_n$ are deterministic. Appointments must be scheduled during regular business hours, that is $a_i \le T$. (Appointment times can also be subject to some extra constraints, *e.g.*, $a_i \le T_i$, where T_i is a deadline for customer i. Such simple linear constraint can readily be embedded in our algorithms.) The service time (duration) and punctuality of the ith customer are $D_i \ge 0$ and P_i, respectively. Customer i arrives to the system at time $t = a_i + P_i$; $P_i > 0$ implies that the customer is late,

and conversely for $P_i < 0$. The random variables D_i and P_i are described by distribution functions F_i and G_i, respectively; these distributions are known to the planner. Distribution F_i quantifies the cancellation probability for customer i via its mass at the origin—cancelled appointments (no-shows) can be thought of as appointments with zero service requirements. It is assumed that D_i and P_i are mutually independent (this assumption is not crucial and can be relaxed), as well as independent across customers. In general, the distributions F_i and G_i can vary with a_i. That is, appointment time-of-day can impact punctuality, service duration or no-shows. Let S_i be the service start time of customer i. The corresponding server is busy during the service interval $[S_i, S_i + D_i)$. Customers are served by any available idle server in the order of their arrival. Customer i enters service at S_i only if this does not lead to a capacity violation during the service interval. That is, a customer that enters service stays in service until service is completed—a decrease in capacity never forces a customer out of service.*

Customer i experiences waiting $W_i = S_i - a_i - P_i$ if showing up for the appointment, with probability $1 - F_i(0)$, and $W_i = 0$ otherwise. Similarly, there is overtime work $O_i = (D_i - (T - S_i)^+)^+$ if customer i shows up for the appointment, and $O_i = 0$ otherwise. Note that O_i is the amount of work for ith customer that occurs beyond the end of business hours T; the sum of all O_i's is thus the total overtime, the amount of work required beyond time T. The appointment *sequencing* problem is to select the appointment times $a_1, \ldots, a_n$, so as to minimize the expected cost

$$\mathbb{E}\left[\sum_{i=1}^{n}(W_i + \gamma O_i)\right]. \tag{1}$$

Given a permutation π of $(1, \ldots, n)$, the appointment *scheduling* problem is to select appointment times $a_1, \ldots, a_n$, in order to minimize (1) subject to $a_{\pi_1} \le a_{\pi_2} \le \cdots \le a_{\pi_n}$. That is, the scheduling problem is easier than the sequencing problem, since the relative order of appointments in the former is fixed, and the planner needs to choose only the spacing between them.

Modeling Choices We conclude the presentation of the scheduling and sequencing problems by discussing some of the modeling choices. First, as we remarked in Section 2, a penalty term for server idleness is often included in the objective of appointment problems like ours. Some papers omit it, however, noticing that problems including such a penalty can be reduced, under mild conditions, to problems minimizing objectives as in (1), with parameter γ appropriately adjusted (Kong et al. 2013). Herein, we omit the cost-term for server idleness to better align our study (a) with DFCI infusion service practices wherein servers correspond to nurse-staffed chairs for which costs have

* Such an assumption is common in analyses of time-varying queueing systems, *e.g.*, see Liu and Whitt (2011). Moreover, it implies that service times are realized upon arrival. At DFCI infusion units, this occurs when patient's clinical condition is assessed upon arrival, and the corresponding treatment plan is adjusted accordingly.

already been sunk, and (b) with the papers by Kong et al. (2013) and Mak et al. (2015) that serve as important benchmarks in our numerical experiments. Nevertheless, it is straightforward to extend our model to include a server idleness penalty term; we remark on this extension in Section 5.3.

Second, waiting time is calculated from the customer's arrival to the system. Alternatively, it could be calculated from the customer's scheduled appointment time. Both these definitions have been well studied in the literature; see Klassen and Yoogalingam (2014) for references and a thorough discussion of the relative merits of each approach. In short, the former (latter) is more relevant for systems wherein congestion in the waiting area has some negative (no) impact. Accordingly, DFCI measures waiting time from patient's arrival, and this is the definition we adopt.

Finally, service for each customer beyond T contributes to the calculation of overtime in the expected cost to be minimized. Alternatively, overtime could be calculated as the amount of time that makespan exceeds T by, *i.e.*, as $(\max_i\{S_i + D_i\} - T)^+$. If the per hour overtime cost is fixed and independent of the number of customers being served beyond regular hours, then the latter definition would be a better fit. If the per hour overtime cost is variable and scales with the number of customers, the definition we use in our model would be a better fit. There are certainly service systems that would fit better under one case, and service systems that would fit better under the other. For cancer centers, because of strict staff-to-patient ratios, overtime costs tend to be variable and the higher the number of patients being served, the higher these costs usually are.

5. Our Infinite-Server (IS) Approach

We now present our approach to solve the appointment scheduling and sequencing problems. As remarked above, even the scheduling version is a very hard problem. For "single-server" ($c_t = 1$ for all t) scheduling, although no efficient, general-purpose solution methods exist, there is a plethora of well-performing heuristics (surveyed in Section 2). Unfortunately, none of these approaches extend to multiple-server scheduling and sequencing problems where uncertainty is fully taken into account. Motivated by the challenges facing DFCI, our goal is to propose a solution approach that accommodates multiple-server environments, while utilizing the available wealth of operational data. As already noted, the multiple-server version of the problem is challenging due to the absence of closed-form, tractable expressions for its underlying queueing dynamics. Consequently, even the mapping of a *given* schedule to actual occupancy, or delays for that matter, remains intractable due to uncertainty of service requirements and punctuality.

We first introduce a model that approximates occupancy in a multiple-server system, given a fixed appointment schedule. Next we perform extensive validation — using real data from DFCI — that demonstrate the credibility of our model's approximations for practical situations. Then, we

Table 2 **A sample of the infusion-unit appointment book at DFCI; we report neither precise date nor patient id due to privacy considerations.**

date	time	patient_id	duration (min)	link_flag	floor_id	disease_center
AA-BBB-2014	15:00	xxxxx021	60	unlinked	9	breast oncology
AA-BBB-2014	12:30	xxxxx247	120	unlinked	9	breast oncology
AA-BBB-2014	10:30	xxxxx083	180	linked	9	genitourinary oncology
AA-BBB-2014	12:30	xxxxx602	60	linked	9	breast oncology
AA-BBB-2014	12:00	xxxxx740	120	linked	9	genitourinary oncology
AA-BBB-2014	07:00	xxxxx741	60	unlinked	9	breast oncology
⋮	⋮	⋮	⋮	⋮	⋮	⋮

embed this model in an optimization routine that allows us to produce well-performing schedules. We detail these steps in the following sections.

5.1. Infinite-Server Model

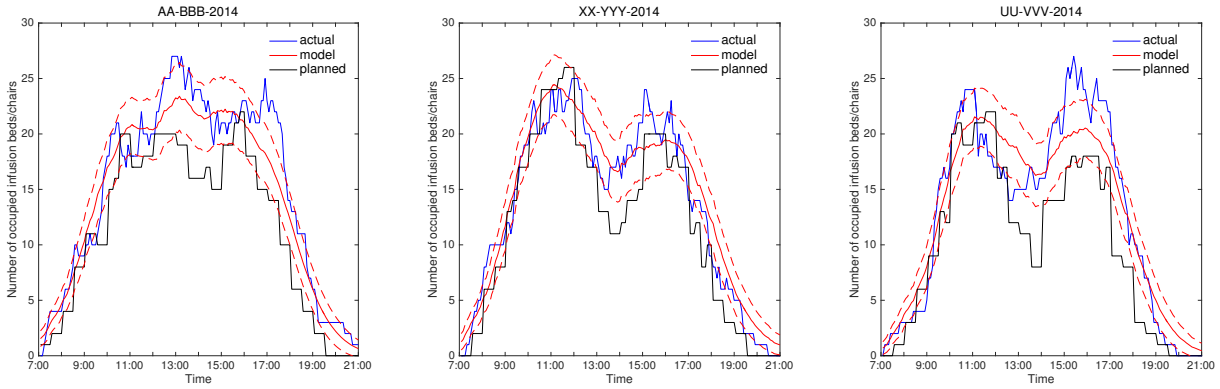
In this subsection, we introduce a model for approximating occupancy given a particular schedule. To provide some motivation first, consider Table 2: it depicts a snapshot of the appointment book for the ninth floor infusion unit at DFCI, at the beginning of the day AA-BBB-2014. How "good" is this schedule? To answer this question, at the very least one must usefully predict the resulting occupancy (given distributional information on punctualities and service durations).

In Figure 4, the black "brick-shaped" line illustrates the occupancy as planned in the appointment book on AA-BB-2014, plus two other days. This deterministic description of planned occupancy is based on nominal service durations and perfect punctuality (as in Table 2). On the other hand, because of uncertainty in treatment durations and punctuality, actual occupancy can be viewed as a stochastic process, for which only a single realization is observable for a particular day (schedule). The blue lines illustrate the actual, realized occupancy we recovered using RTLS data from these days. Clearly the (deterministic) planned occupancy does not describe the corresponding actual process adequately. We hence develop a model for approximating occupancy based on an Infinite-Server (IS) model. Using this model, we obtain an occupancy approximation that is itself a stochastic process much like actual occupancy: the red lines illustrate its mean (solid) plus/minus a standard deviation (dashed). A key observation is that the IS model provides a very close fit to observed occupancies, well within confidence intervals. We perform an extensive model validation in Section 5.2 after we formally present the IS model next.

Given a set of n customers and their appointment times $a_1, \ldots, a_n$, our approximation is obtained by eliminating the constraint on the number of available servers. In particular, customer i arrives at time $t = a_i + P_i$ and leaves the infinite-server system just before $t = a_i + P_i + D_i$. For notational convenience, we define $\tilde{F}_i(x) := 1_{\{x \geq 0\}}(1 - F_i(x))$. Let $Z_i := \{Z_i(t), t \in \mathbb{R}\}$, where

$$Z_i(t) := 1_{\{a_i + P_i \leq t < a_i + P_i + D_i\}}$$

Figure 4 Examples of actual (solid blue) and planned (dotted black) infusion bed/chair occupancy for three days at DFCI. For the corresponding models, the means (solid red) and plus/minus one standard deviation from the mean (dashed red) are shown.



is the indicator of customer i being present in the system at time t. Then, it follows that

$$\mathbb{E}Z_i(t) = \mathbb{E}\tilde{F}_i(t - a_i - P_i) =: \Omega_i(t) \quad \text{and} \quad \mathrm{Var}(Z_i(t)) = \Omega_i(t) - \Omega_i^2(t).$$

We use $Z := \{Z(t), t \geq 0\}$ to denote the total number of customers in the infinite-server system:

$$Z(t) = \sum_{i=1}^{n} Z_i(t).$$

Assuming independence of customers, it follows that

$$\mathrm{Var}(Z(t)) =: \sigma^2(t) = \sum_{i=1}^{n} \Omega_i(t)\,(1 - \Omega_i(t))\,. \tag{2}$$

Let $X := \{X(t), t \in \mathbb{R}\}$ be the occupancy process under a fixed capacity profile (c_t) and a given schedule $(\{a_i, i = 1, \ldots, n\})$. The corresponding infinite-server process Z is expected to approximate the actual occupancy X well for situations where capacity violations do not occur frequently. In general, however, Z serves as a lower bound only: $X(t) \geq Z(t)$, for all t. Nonetheless, as we shall discover, using the infinite-server process proves overall useful for scheduling and sequencing purposes.

Before using real DFCI data to validate our IS model approximations, we present an illustrative example with synthetic data. This example is our first demonstration that the IS model provides a credible approximation of the actual occupancy process for our purposes.

Example 1 (Homogeneous customers). Consider a system serving $n = 100$ homogeneous customers. In particular, for all customers $i = 1, \ldots, n$, the service distribution is exponential with rate $\beta = 1$, $\tilde{F}_i(x) = 1_{\{x \geq 0\}} e^{-\beta x}$, and the punctuality distribution G_i is Laplace, defined by the density $\lambda e^{-\lambda|x|}/2$, with $\lambda = 10$. Capacity c_t is constant throughout (equal to 20, 22, 25, or ∞).

We consider two possible service schedules, A and B illustrated in the upper right portion of Figure 5. Schedule A can be thought of as the result of a "means-based" scheduling process, akin to the deterministic planning we alluded to in the beginning of this section: service durations and punctuality are assumed to equal their expected values (unit and zero, respectively). Specifically, Schedule A entails five customer batches. Each batch has 20 customers and arrives every one unit of time (as long as service is expected to last). Schedule B has an initial batch of 25 customers arriving at $t=0.05$, after which the 75 customer scheduled times are uniformly spread out between $t=0.3$ and $t=4$.

Using our IS model, one has

$$\Omega_i(t+a_i)=\begin{cases}\frac{\lambda}{2(\lambda+\beta)}e^{\lambda t}, & t<0,\\ \frac{\lambda}{2(\lambda+\beta)}e^{-\beta t}+\frac{\lambda}{2(\lambda-\beta)}(e^{-\beta t}-e^{-\lambda t}), & t\geq 0.\end{cases}$$

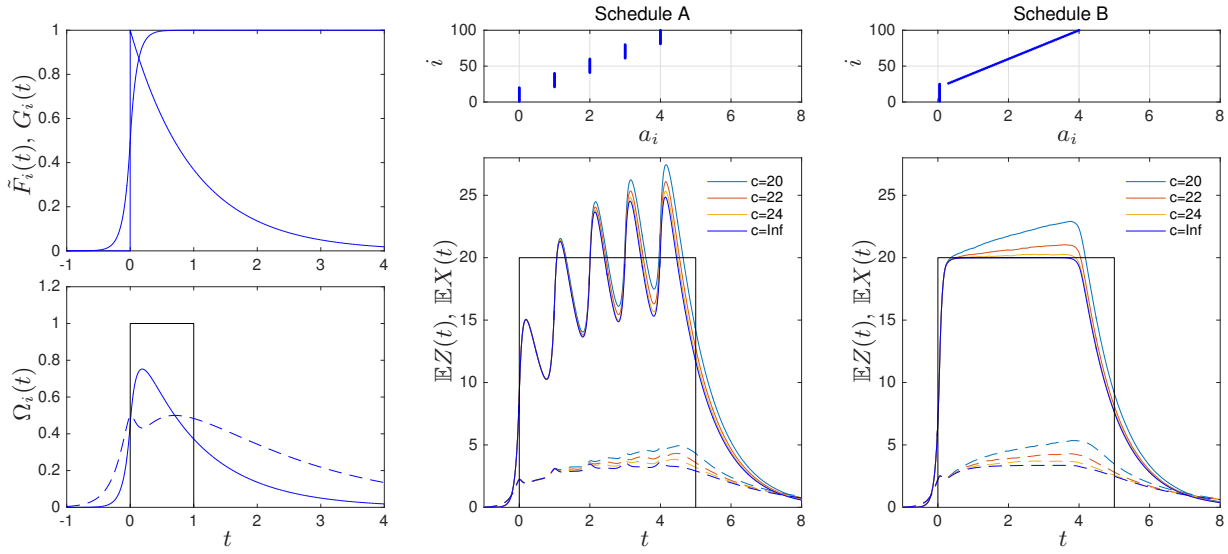
In the left column of Figure 5, $\tilde{F}_i$ and G_i are plotted on the upper plot. The lower plot depicts Ω_i (blue line) and $(\Omega_i-\Omega_i^2)^{1/2}$ (dashed line), *i.e.*, the standard deviation of Z_i, for $a_i=0$. On the lower plot, the corresponding Ω_i under perfect punctuality ($P_i=0$) and deterministic service ($\mathbb{E}D_i=1$) is plotted as well (black line).

In the center column, for Schedule A (shown on the top), we plot $\mathbb{E}X(\cdot)$ under three different capacity levels (c_t is constant equal to 20, 22 and 24) and $\mathbb{E}Z(\cdot)$ ($c_t=\infty$); the corresponding standard deviations are shown with dashed lines. The lower plot also depicts the IS process under the assumption of unit deterministic service times and perfect punctuality (black line), whereby no capacity violation occurs for the considered capacity levels. In the right column, we depict the associated quantities for Schedule B.

The example above provides some intuition behind the first managerial insight we outlined in the Introduction. Specifically, it illustrates that the "shape" of the demand curve affects performance significantly. Because the resulting shape depends on full distributional information regardless of the number of jobs/servers, variability remains a key consideration and does not "average out" in multi-server systems.

Furthermore, the example also suggests that the higher the capacity level, the better the IS model approximates the finite-capacity system (as seen in Figure 5). Interestingly, even for $c_t=20$, the IS process still provides a viable approximation of the occupancy process under both schedules. In particular, the IS model readily reveals that Schedule A would lead to a less efficient demand-capacity match, as compared to Schedule B. Most important, the IS approximation accurately reflects a "shape" of the demand curve that matches capacity more closely under Schedule B rather than under Schedule A, as seen in Figure 5. As we shall see in our numerical experimants, even if the approximation of the exact occupancy level is off, approximating the demand shape would prove to be sufficient for scheduling/sequencing purposes.

Figure 5 Illustration for Example 1.

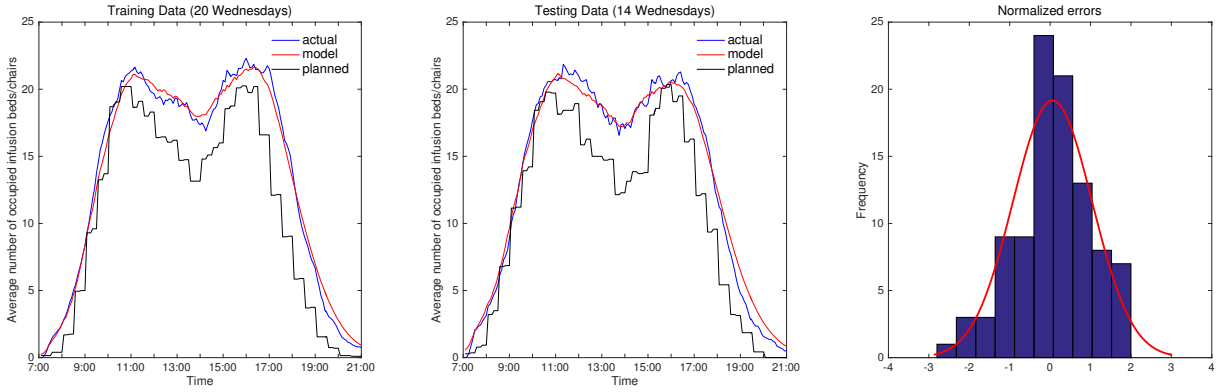


5.2. Model Validation

Before using our infinite-server model to develop a sequencing algorithm, we validate it within the DFCI operating environment. This is done with real data, and the analysis yields our first important insight: despite the complexity of DFCI operations, our basic IS model provides a very good approximation of the intricate underlying dynamics. In particular, actual occupancy at DFCI infusion units (blue lines in Figure 4) is driven by myriad factors and is the outcome of interdependent processes that are intractable to analyze. Yet, we find that the IS model provides approximation bands that accurately characterize occupancy: as discussed above, Figure 4 depicts these bands (red) for a particular DFCI floor (infusion unit) on three specific days, with actual occupancy (blue) falling well within them. Furthermore, we find the high fidelity of the IS proximation to be consistent. Specifically, Figure 6 depicts the actual and IS-model occupancies averaged across multiple days, alongside the corresponding standardized errors; we observe a very close fit. We detail our studies in the rest of the present subsection.

Model focus and granularity We apply our model to a single floor (infusion unit) at DFCI separately, rather than to all floors simultaneously, since (i) operation of different floors is semi-autonomous, (ii) a disease center is located on a single floor, and (iii) statistical properties of disease centers (patients undergoing treatment, operations) can differ significantly. Moreover, we consider different days of the week separately, as provider schedules differ from one day of the week to another, resulting in different daily patient loads and populations (*e.g.*, Fridays are not as "busy" as other days of the week). Given a floor and a day of the week, we estimate bed/chair time distributions and punctuality distributions. In particular, we estimate "service" distributions

Figure 6 The average number of occupied infusion beds/chairs at DFCI for the training (left) and the testing (center) sets based on RTLS data ("actual," solid blue) and appointment book data ("planned", dotted black); the average of model means ("model," solid red) is shown as well. A histogram of normalized errors is shown along with the corresponding normal density (right).



conditional on (i) the disease center, (ii) the scheduled infusion duration, (iii) the type of the appointment (linked/unlinked), and (iv) the time of the day (morning vs. afternoon); these parameters are extracted from the appointment books of DFCI. As far as punctualities are concerned, we estimate conditional punctuality distributions based on (i) the disease center, (ii) the time of the day (scheduled infusion start times), since average delays change during a day, and (iii) the type of the appointments (linked/unlinked). Hence, the function Ω_i for the ith patient depends on (i) the scheduled treatment (infusion) start time (via the a_i and the punctuality/service distribution that corresponds to a_i), (ii) the scheduled treatment duration (via the corresponding distribution that corresponds to the treatment duration), and (iii) the type of the appointment. Given these estimated distribution functions, an infinite-server model is constructed for a given day (schedule). The model mean and variance can be computed from Ω_i's based on the previous subsection.

"Modeling" Wednesdays on the ninth floor We illustrate the described procedure by an analysis of Wednesdays on the ninth floor. A time period containing 37 Wednesdays in 2014 was considered (from February 19 to October 29). During this time period, the infusion unit was shared between two disease centers (breast oncology and genitourinary oncology). Three days were excluded from our analysis (March 12, June 11 and August 20) due to RTLS system interruptions on those particular days. The remaining 34 days were divided into the training (20 Wednesdays, from February 19 to July 16) and testing (14 Wednesdays, from July 23 to October 29) sets. All schedules for the considered days are different. The number of patients varies from 62 to 101 a day (the average is 83.3, while the standard deviation is 7.7). The number of scheduled (actual) infusion hours ranges from 122.0 (153.4) to 188.5 (223.8) a day (the average is 153.6 (190.3), while the standard deviation is 17.2 (17.8)).

The training set was used to estimate conditional punctuality and service distributions. All analysis was conducted on a five-minute scale, *i.e.*, the unit of time is five minutes. Punctuality distributions were estimated based on values of appointment times. In particular, based on a statistical analysis, eight time intervals were considered: [00:00-09:00), [09:00-10:00), [10:00-11:00), [11:00-12:00), [11:00-12:00), [12:00-14:00), [14:00-15:00), [15:00-16:00), and [16:00-24:00). Similarly, service distributions were estimated based on scheduled appointment durations being in one of eight intervals (in minutes): [5,25), [25,45), [45,75), [75,105), [150,210), [210,270), [270,330), [330,420), [420,540), [540,660), and [660,1000). For service distributions, dependence on appointment times was not found to be significant, unlike for punctuality distributions, and was therefore not modeled. The estimated distributions were used to construct infinite-server models for all considered days.

Validation results In Figure 6, we plot the number of occupied infusion beds/chairs averaged across the training (left) and the testing (center) sets. In both graphs, the "actual" and "planned" curves are plotted based on RTLS data and data derived from the appointment book, respectively. We note that an infinite-server model for a given day (conditional on its schedule, *i.e.*, appointment book) is a random object. Thus, in Figure 6, the red lines corresponding to the model represent the average (across days) of model means. In Figure 4, we illustrate the relationship between the actual/planned number of occupied infusion beds/chairs and the model for three particular days. For the model, we show the means (the solid red lines) as well as the levels corresponding to plus/minus one standard deviation from the means (dashed red lines). *The actual number of occupied beds/chairs can be viewed as a single realization of a random process conditional on the schedule.* Given a day in a testing set and a time of the day, we define a normalized error as a difference between the actual number of occupied beds/chairs and the model mean, scaled by the standard deviation of the model. A histogram of normalized errors for the testing set and specific times of the day (10:00, 11:00, 12:00, 13:00, 14:00, 15:00, and 16:00) is shown in Figure 6 (right); a corresponding standard normal density is plotted as well. Note that the distribution of normalized errors would be standard normal if the occupancy process were indeed an infinite-server process with a large traffic intensity. In this case, the sample average and standard deviation would be equal to 0.06 and 0.98, respectively. These numbers demonstrate the very strong fit the IS model provides, as we discussed at the beginning of this section.

5.3. Infinite-Server Sequencing Heuristics

We now use the infinite-server model to propose a heuristic for the appointment scheduling and sequencing problems. For a function $r:\mathbb{R}\to[0,\infty)$, we define a cost of a sample path of Z as

$$Q(Z):=\int_{-\infty}^{\infty} r(Z(t)-c_t)\,\mathrm{d}t+\tilde{\gamma}\int_{T}^{\infty}\min\{Z(t),c_t\}\,\mathrm{d}t,$$

for some $\tilde{\gamma} > 0$; the two terms approximate costs of waiting time and overtime. (As a side note, penalizing server idleness would similarly involve here adding the term $\int_0^T (c_t - Z(t))^+ \,\mathrm{d}t$ to the sample path cost.)

Define $\tilde{Z} = \{\tilde{Z}(t), t \in \mathbb{R}\}$ as a CLT-based approximation of Z:

$$\tilde{Z}(t) := \mathbb{E}Z(t) + \xi(t)\,\sigma(t), \tag{3}$$

where $\xi(t)$ is a standard normal random variable and $\sigma(t)$ is given by (2). Consequently, $Q(\tilde{Z})$ is an approximation of $Q(Z)$, which gives rise to the following problem:

$$\min_{\{a_i \in [0,T]\}_i} \mathbb{E}Q(\tilde{Z}). \tag{4}$$

For example, if $r(x) = x^+$, one can readily obtain that

$$\mathbb{E}Q(\tilde{Z}) = \int_{-\infty}^{T} \Psi(t)\,\mathrm{d}t + (1-\tilde{\gamma}) \int_{T}^{\infty} \Psi(t)\,\mathrm{d}t + \tilde{\gamma} \int_{T}^{\infty} \mathbb{E}\tilde{Z}(t)\,\mathrm{d}t, \tag{5}$$

where

$$\Psi(t) := \sigma(t)\,(\varphi(\psi(t)) + \psi(t)\,\Phi(\psi(t))) \quad \text{and} \quad \psi(t) := \frac{\mathbb{E}Z(t) - c_t}{\sigma(t)};$$

here we use the common notation for the standard normal density and distribution function (φ and Φ, respectively). In the case when G_i does not vary with a_i, (5) can be used to obtain explicit expressions for partial derivatives of $\mathbb{E}Q(\tilde{Z})$ with respect to appointment times a_i.

The advantage of using approximation (3) is that $\tilde{Z}(t)$ is fully characterized by its mean and standard deviation only. It is expected that the approximation is relevant whenever the central limit theorem is applicable. In general, solving (4) to optimality is hard, since the objective function is not necessarily convex. However, one can efficiently obtain local optimal solutions for large problem sizes. Candidate starting points can be obtained using the OV heuristic, or efficient means-based sequencing algorithms that ignore uncertainty, or random sampling. By approximately solving the problem for various values of $\tilde{\gamma}$, one can obtain schedules close to the efficient frontier of wait time versus overtime costs, and eventually select the one that trades off γ units of wait time with one unit of overtime.

The non-convexity of the IS heuristic objective (4) is a limitation of the IS approach, since one must resort to local-search algorithms. One can overcome this limitation by formulating a well-behaved objective that leads to a simpler (or standard) optimization framework. However, such an alternative comes at the expense of neglecting key (stochastic) features of the underlying system. Our design choice is to incorporate essential stochastic aspects of the system (along with the corresponding distributional data) in the model by sacrificing convexity.

6. Performance Evaluation: Comparison with Existing Approaches

We benchmark our IS solution approach, which accommodates multiple servers, against the state-of-the-art solution approaches in the literature that deal with single-server instances. To this end, we develop a novel Data-Driven Robust (DDR) solution approach that "bridges" the two: it deals with any capacity vector c_t, albeit for the appointment scheduling problem where customers' order is fixed (see Table 1). Existing approaches in the literature assume perfect punctuality, that is, $P_i = 0$, for all i; we henceforth also make this assumption in the present section.

Numerical studies on the scheduling problem reveal that the performance of our IS approach is at least as good as (if not superior to) the DDR approach for the multiple-server problem, while the performance of the latter is at least as good as (if not superior to) the state-of-the-art heuristics in the literature for the single-server problem. To provide further evidence on the quality of DDR as a benchmark, we compare DDR with existing means-based scheduling approaches for the multiple-server problem as well, and find that it provides a cost reduction of around 10%.

6.1. Data-Driven Robust (DDR) Scheduling

We develop a robust optimization approach to solve the appointment scheduling problem under perfect punctuality. It is applicable to any capacity specification c_t and considers discrete time; for the single-server case, it can also be readily adapted to continuous time. Our approach differs from classical approaches in robust optimization by being grounded in data—a point elaborated upon in the uncertainty set description below.

At a high-level, the robust optimization approach employs two stages. In the first one, we select a schedule $a_1 \le a_2 \le \cdots \le a_n$; it minimizes the cost incurred in the second stage, in which an adversary draws the service durations from an uncertainty set so as to maximize the incurred cost.

Uncertainty set of service durations Uncertainty sets in classical robust optimization rely on limited information about the uncertain variables, which in our case are the service times. In particular, they are usually calibrated using only the support of the uncertain variables, for example, or their means, or some of their higher-order moments, etc. Consequently, they tend to perform well in settings where there is a scarcity of data.

In our setting, on the other hand, a wealth of data is available. Thus, instead of extracting only partial support and/or moment related information, we propose a novel construction of uncertainty sets that leverage all the available data *explicitly*, specifically by "sampling" from the available empirical distributions. (Hence, we refer to our approach as *data-driven*.) In particular, consider n independent uniformly distributed random variables, U_i, $i = 1, \ldots, n$. Samples from these random variables can be readily used to obtain samples of the service time durations of the n customer appointments:

$$D_i = F_i^{\leftarrow}(U_i), \quad i = 1, \ldots, n;$$

here, $F_i^{\leftarrow}$ is the (left continuous) inverse of F_i: $F_i^{\leftarrow}(x) = \inf\{s : F_i(s) \geq x\}$. Since the U_i's are independent and identically distributed, we propose to use CLT-style constraints to form the uncertainty sets that they lie in. Such constraints were introduced and found to perform well in recent papers in the robust optimization literature; see, *e.g.*, Bandi and Bertsimas (2012).

To formalize the construction of our proposed uncertainty set, we need to "discretize" our random variables. In particular, consider J point values of the ith service duration's inverse distribution function:

$$\delta_{ij} := F_i^{\leftarrow}\left(\frac{j}{J+1}\right), \quad i = 1, \ldots, n, \quad j = 1, \ldots, J.$$

Let the associated uniform random variable U_i take the corresponding discrete values $j/(J+1)$, $j = 1, \ldots, J$. We can then express the service durations as

$$d_i = \sum_{j=1}^{J} \delta_{ij} u_{ij}, \quad i = 1, \ldots, n, \tag{6}$$

where u_{ij} is the indicator variable

$$u_{ij} = \begin{cases} 1 & \text{if} \quad U_i = \frac{j}{J+1}, \\ 0 & \text{otherwise.} \end{cases}$$

Viewed from a different angle, the variables u can be interpreted as *assignment* variables. That is, the service durations have a range of possible values δ that they can take. The variables u assign to the service durations one of their possible values.

The indicator variables lie in the uncertainty set

$$\mathcal{U} := \left\{ u \in \{0,1\}^{n \times J} : \sum_{j=1}^{J} u_{ij} = 1, \forall i \in \{1, \ldots, n\}, \left| \frac{\sum_{i=1}^{n} \sum_{j=1}^{J} \frac{j}{J+1} u_{ij} - \frac{n}{2}}{\sqrt{\frac{n}{12}}} \right| \leq \Gamma \right\},$$

where Γ is a *conservatism* parameter. The first constraint in the definition of $\mathcal{U}$ is an assignment constraint: precisely one of the u_{ij}'s is equal to one for each i. The second constraint ensures that the normalized sum of the i.i.d. variables U_i is not larger than Γ or smaller than $-\Gamma$. For example, if $\Gamma = 1$, the sum is within one standard deviation from its mean. Higher values of Γ mean that the sum of service durations can deviate even more from its mean, allowing nature a larger set of possible durations to pick from so as to maximize costs. The resulting uncertainty set that service durations lie in is then

$$\mathcal{D} := \left\{ d \in \mathbb{R}^n : d_i = \sum_{j=1}^{J} \delta_{ij} u_{ij}, \forall i \in \{1, \ldots, n\}, u \in \mathcal{U} \right\}.$$

Data-driven robust appointment scheduling We can now formulate the DDR problem as the following two-stage optimization problem:

$$\text{minimize} \quad \max\{w(a,d) : d \in \mathcal{D}\} \tag{7a}$$

$$\text{subject to} \quad a_1 \le a_2 \le \cdots \le a_n \le T \tag{7b}$$

$$a \in \mathbb{Z}^n, \tag{7c}$$

where $w(a,d)$ is the incurred cost under schedule a and service durations d. In Appendix A we provide a tractable reformulation of (7) and develop a Benders decomposition approach that provably solves the DDR problem to optimality.

6.2. Comparison of DDR with Existing Approaches

We conduct numerical experiments to assess the performance of our data-driven robust approach. We find that DDR performs at least as good as the state-of-the-art solution approaches in the literature for the single-server problem. Furthermore, we find that it significantly outperforms means-based scheduling approaches for the multiple-server problem.

Single-server scheduling First, we compare DDR with existing appointment scheduling algorithms that deal with single-server instances under uncertainty. We use an experimental setup akin to Kong et al. (2013). In particular, we consider different cases, where in each case we vary the number of customers, service distributions, utilization and overtime cost rate. For concrete parameter choices, the reader is referred to Appendix B.1.

We compare our DDR approach with Stochastic Optimization (SO) and a Distributionally-Robust (DR) approach. SO minimizes expected cost having access to full distributional information. It is used as a benchmark, since it is known to produce near optimal solutions for the small problem sizes we consider here (Denton and Gupta 2003). Table 3 provides an overview of our results. Not surprisingly, SO produces lower costs than our approach; however, our approach yielded costs that were only 2% higher on average across all cases, and at the most 3.7%. We provide a more detailed comparison in Appendix B.1.

The DR approach we use follows Kong et al. (2013) and Mak et al. (2015), by minimizing worst-case cost, over all distributions that have the same mean and variance as the true service distributions. That is, the DR approach leverages only partial information about service times. It is used as a benchmark, since it has been shown to perform very well for realistic problem sizes that include more than 40 customers (still only with a single server). In comparison, our DDR approach yielded costs that were 2.5% lower on average across all cases, occasionally 4.6% lower. Note that our

Table 3 Summary of mean percentage differences (in %) between DDR and SO/DR solutions for the single-server problem, and between DDR and MB solutions for the multiple-server problem. The summary is across service distributions (lognormal/gamma) and utilization levels (0.85/1.00/1.15).

	single-server		multi-server
	SO	DR	MB
Min. gap	+0.8	−0.1	−2.6
Avg. gap	+2.0	−2.5	−10.6
Max. gap	+3.7	−4.6	−20.4

approach yields superior performance by utilizing more distributional information (as we remarked above).[†] For additional comparison details, see Appendix B.1.

Multiple-server scheduling Second, we compare DDR with standard Means-Based (MB) scheduling for multiple-server problems. The experimental setup we use is akin to the one we used previously for the single-server problem. In particular, we consider the problem of scheduling $n = 40$ customers to 15 servers ($c_t = 15$, $t = 1, \ldots, \bar{T}$, where $\bar{T} = 90$). The per-unit overtime cost is $\gamma = 2$, and overtime contributes to the overall cost after $T = 50$. All customers have perfect punctuality and the same distribution type for their service requirements: lognormal or gamma (as in Kong et al. (2013)). However, the mean and the coefficient of variation for each customer are sampled from a uniform distribution. Specifically, the mean service requirement d_i of customer i is uniform on $[2\bar{d}/3, 4\bar{d}/3]$, where $\bar{d}$ is determined by a desired mean utilization $n\bar{d}/\sum_{t=1}^{T} c_t$; the coefficients of variation are uniform on $[2/3, 4/3]$. For each distribution type (lognormal or gamma) and three utilization levels (0.85, 1.00 and 1.15; this corresponds to $\bar{d} \approx 0.32T$, $0.38T$ and $0.43T$), we generate (sample means and coefficients of variation) 20 cases: these are solved by the two proposed algorithms, MB and DDR (the DDR's parameter Γ was calibrated for each group of 20 cases). The two solutions (for each case) are then fed into a simulator to obtain estimates of the mean of the total cost (based on 10^6 samples).

Table 3 provides an overview of our results. Because it leverages full distributional information of customer requirements, as opposed to simply means, DDR provides a significant cost reduction over MB scheduling that is as high as 20.4%, and averages at 10.6% across our experiments. We provide a more detailed comparison in Appendix B.2, along with details about the specific implementation of MB scheduling we used.

[†] Note, however, that this advantage comes at the cost of having to solve integer optimization problems (as opposed to second-order cone problems in the DR approach). On the positive side, our approach can also deal with multiple-server problems.

6.3. Comparison of IS with DDR

Having established that the DDR approach exhibits near-optimal performance for single-server scheduling problems and that it significantly outperforms means-based scheduling for multiple-server problems, we now benchmark our IS approach against it for multiple-server scheduling problems. We use the same experimental setup as we used previously in the comparison of DDR with MB scheduling. For the IS approach, we use $r(x) = x^+$, a choice we retain for all subsequent experiments, and calibrate the parameter $\tilde{\gamma}$ for each group of 20 cases. In a similar fashion as before, the IS and DDR solutions (for each case) are fed into a simulator to obtain estimates of the mean of their total cost.

In order to assess robustness, we also simulate the performance of the IS and DDR models in situations where the distributions are mis-specified, that is the "actual" distributions do not match the ones assumed by the models. In particular, for each of the 120 cases we consider, if both models assumed the lognormal (gamma) distribution to produce schedules, we simulate the costs of the produced schedules under a gamma (lognormal) distribution, with the same first two moments for each customer as the models assumed.

Our numerical results are summarized in Figure 7 and Table 4. Note that the IS approach delivers a uniformly superior performance to DDR across all our experimental setups. When the underlying distributions are accurately specified, *i.e.*, the simulator uses the same distribution as the models assumed, the cost reduction that IS achieves over DDR is 3.6%, on average. Interestingly, IS holds on to its advantage, even under distributional mis-specifications, *i.e.*, when the models assume some distribution type but the "actual" one that the simulator uses is different, lowering cost by 2.8%, on average. In summary, our experiments suggest that the IS approach is uniformly slightly superior to the DDR approach, with the latter found to be near-optimal for the single-server problem setting.

From a computation standpoint, we measured the required computation time of both approaches in our experiments and found IS to be one to two orders of magnitude faster. Specifically, DDR required approximately two hours, on average, to solve each instance, whereas IS required approximately two minutes. For larger problem sizes (*e.g.*, $n = 50$ or higher), we found DDR unable to solve them within four hours—Kong et al. (2013) report that their approach can solve instances up to around similar sizes. In comparison, we found IS able to solve scheduling problem instances with $n = 100$ in less than thirty minutes, on average. These experiments illustrate the ability of IS to handle practice-relevant problem sizes. For details, see Appendix B.2.2.

Furthermore, recall that, unlike DDR, the IS approach is also capable of handling sequencing and uncertainty in punctualities. In particular, we considered generalizations of DDR that allowed for punctuality and sequencing, but found them unable (due to computational burden) to solve for the experiments we considered in this section. In contrast, IS was able to solve them, when accounting

Figure 7 **Mean costs of DDR (dash-dot blue) and IS (solid red) appointment scheduling solutions for different distribution (actual/assumed) pairs and mean utilization levels (0.85, 1.00 and 1.15). The number of servers and customers are 15 and 40, respectively.**

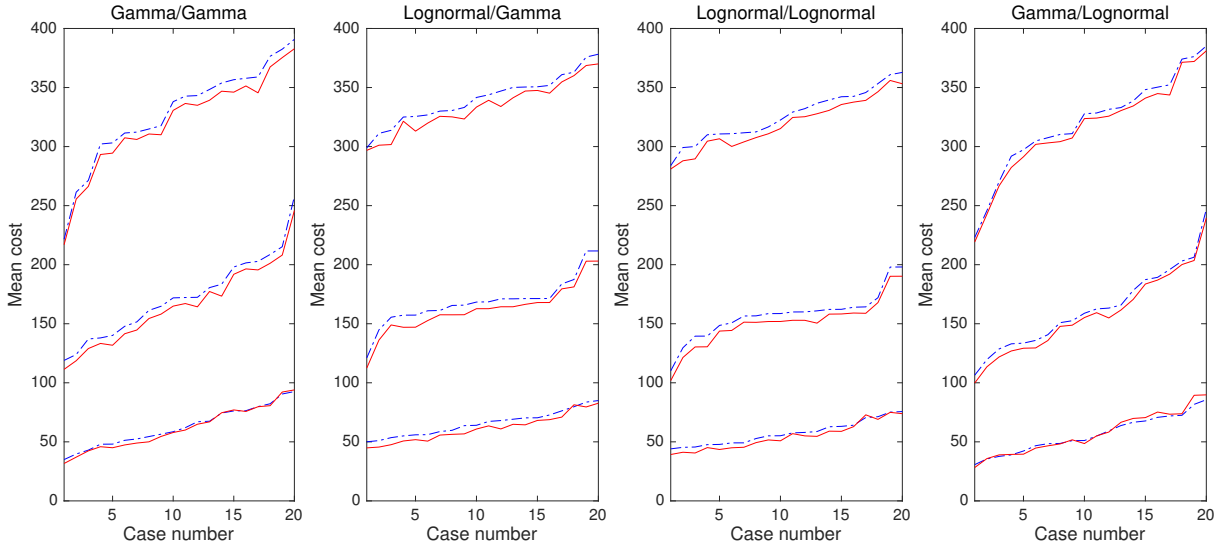


Table 4 **Mean percentage differences (in %) between IS and DDR solutions (MB solutions) for means and percentiles. The pair of distributions refers to actual/assumed distributions, where the L and G stand for lognormal and gamma, respectively.**

Mean utilization	Performance measure	Distributions			
		G/G	L/G	L/L	G/L
0.85	Mean	−3.0	−6.8	−5.6	+0.7
	75% percentile	−1.7	−5.3	−4.5	+1.5
	85% percentile	+2.1	−2.5	−1.5	+4.9
	95% percentile	+7.4	+1.9	+2.8	+10.0
1.00	Mean	−4.1	−4.1	−4.3	−3.2
	75% percentile	−3.4	−3.3	−3.4	−2.5
	85% percentile	−2.7	−2.9	−2.9	−1.9
	95% percentile	−1.8	−2.6	−2.5	−0.9
1.15	Mean	−2.2	−2.1	−2.1	−1.5
	75% percentile	−2.8	−2.8	−2.8	−2.2
	85% percentile	−3.0	−2.9	−2.9	−2.3
	95% percentile	−2.7	−2.8	−2.8	−2.2

for punctuality, in less than three minutes on average. When allowing for sequencing, IS required approximately 30 minutes, on average, to solve. For details, we refer the reader to Appendix B.2.2. Next, we explore using IS for practice-relevant problems at DFCI that are larger-sized and involve both punctuality and sequencing.

7. Performance Evaluation: Appointment Sequencing at DFCI

In this section, we compare our IS appointment sequencing approach with means-based sequencing (detailed below) in the context of DFCI infusion operations. The latter approach is prevalent in outpatient settings (Berg and Denton 2012). In particular, we use our high-resolution RTLS data to generate multiple experimental scenarios, whereby daily infusion appointment sequencing tasks, as faced by DFCI, are carried out by the two approaches. Typical scenarios for a DFCI infusion unit involve approximately $n = 90$ daily appointments that are served by 25+ infusion beds/chairs during business hours.

The comparison of the two approaches yields the second important insight of this work: our approach, which accounts for variability in a real-world multiple-server system, significantly outperforms the means-based approach. In particular, the analysis here suggests that deployment of our IS appointment sequencing approach at DFCI could provide total cost (costs of waiting plus overtime) reduction in the order of 15%–60%.

We consider two experimental setups. The first one involves the classical offline appointment sequencing problem we have dealt with thus far, which is already a very good proxy of how infusion operations are run at DFCI. Dealing with offline sequencing, and abstracting away from particularities of the DFCI infusion appointment process, enables us to illustrate the value of using IS in a general-purpose well-understood setting when calibrated using real data.

In the second setup, we model additional features of the DFCI appointment processes that are specific to DFCI, yet also prevalent in other practices. So doing enables us to illustrate that the IS approach continues to be practically useful and to perform very well even in realistic appointment systems that deviate from classical offline sequencing.

7.1. Offline Appointment Sequencing at DFCI

Consider a *specific* historical day at DFCI for which one can observe the original appointments that were scheduled, including their "type:" the latter is defined by day/time of the week, appointment's duration, associated disease center and whether it followed a linked exam. Lacking access to the true statistical properties of the appointments' durations/punctuality for that day (we only observe a single sample path), we use a bootstrapping procedure to "reconstruct" these properties from the data by using observed punctuality/service samples of same-type appointments on other days. In particular, we obtain empirical values for punctuality (duration) by sampling from a set of punctuality (duration) of appointments that occurred on the same day of the week and interval of the day (morning vs. afternoon), were scheduled for the same nominal duration, were from the same disease center, and had the same (or lacked) exam linkage. To this end, we use the data recorded over the 34 Wednesdays, as detailed in Section 5.2.

The availability of infusion chairs is specified so that it mimics DFCI nurse-staffing policies. In particular, there is no available capacity before 7:00. Capacity then increases linearly between 7:00 and 10:00 (in intervals of 30 minutes), remains at a constant number (maximum) between 10:00 and 17:00, before decreasing linearly between 17:00 and 19:00 (in intervals of 30 minutes) to reach five available chairs at 19:00 until the day ends. Overtime starts at 19:00. Thus, the capacity profile is parametrized by a single value: maximum capacity (between 10:00 and 17:00). This value is set based on a desired utilization, computed over business hours (7:00-19:00): we compute the average day-demand in infusion hours (based on the original schedule), and then reduce the maximum capacity until a desired value of utilization is reached. As in Section 5.2, the unit of time is chosen to be five minutes. Appointments can be made between 7:00 and 17:00.

Experiments and Results Within this experimental setup, we sought to tackle the offline sequencing problem and produce schedules using our IS sequencing approach and a standard means-based sequencing approach. The latter is an approach where all random quantities are treated as deterministic and equal to their mean values; then state-of-the-art mixed-integer programming techniques are employed to optimize the schedule. In Appendix C.1, we elaborate on the exact means-based sequencing algorithm we used. Computed schedules were simulated and averages are obtained based on 10^6 samples. Numerical results are presented in Figure 8. In particular, on the left plot, we provide an overtime-waiting tradeoff for XX-YYY-2014 (which is the Wednesday in Figure 2), on the ninth floor of DFCI. On that day, 87 patients had infusion appointments and received treatment. The overtime-waiting tradeoff for the IS approach (blue line) is obtained by varying the parameter $\tilde{\gamma}$ (see Section 5.3); and for the means-based approach (black line), it is obtained by scaling capacity (see Appendix C.1 for details).

In addition, we performed the same analysis for ZZ-WWW-2014 (another anonymous Wednesday, not identified due to privacy considerations) for the eighth floor (72 scheduled patients) using the same time period. Since the two floors are occupied by different disease centers, their statistical properties are not identical. In particular, in the considered data set, the average (across patients) mean duration and cancellation probabilities are similar (141 minutes vs. 136 minutes and 1.9% vs. 1.3% for floors eight and nine, respectively), but the average coefficient of variation is higher for the eighth floor (0.77 vs. 0.62).

Table 5 reports the percentage cost decrease that our approach achieves, compared with the means-based approach for different overtime rates γ. We observe that it delivers uniformly a significant improvement, ranging between 18% and 42% across both infusion units and different ratios of overtime to waiting time costs.

Figure 8 **Overtime-waiting tradeoffs for two days on two different DFCI floors. The higher curves correspond to the higher utilization level.**

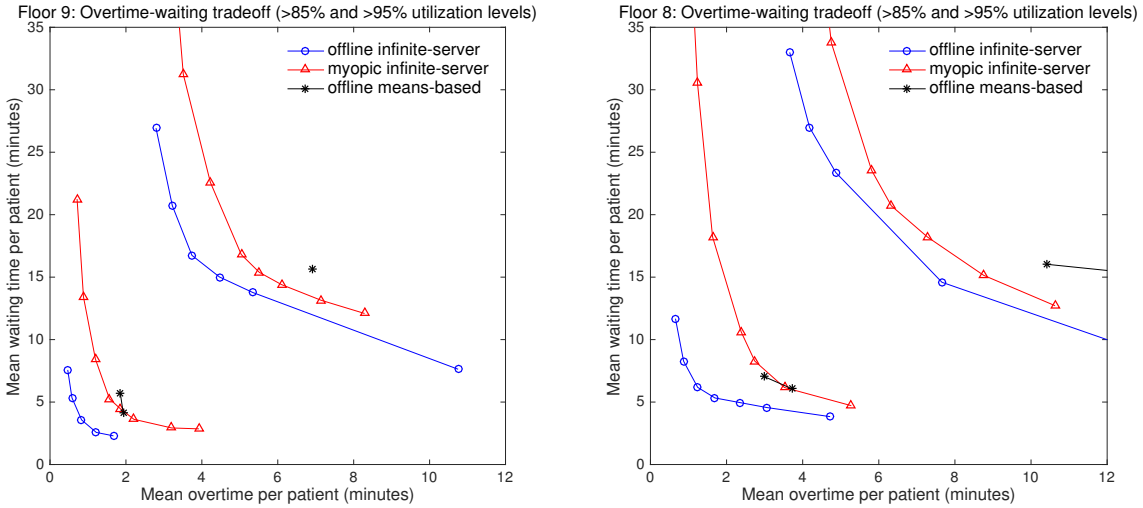


Table 5 **Percentage decrease (in %) for total mean cost, between offline IS and means-based solutions at DFCI.**

		Per-unit overtime cost γ				
	Utilization	1/3	1/2	1	2	3
Floor 9	>0.85	-40.9	-39.2	-38.0	-37.7	-39.1
	>0.95	-37.5	-31.9	-18.6	-18.8	-23.2
Floor 8	>0.85	-25.9	-23.1	-28.2	-33.5	-38.2
	>0.95	-41.9	-34.4	-17.6	-18.8	-20.5

Robustness Checks To assess the robustness of the IS approach, we perform the following checks. We consider: (i) the situation where limited data is available; (ii) a workday at DFCI on which demand patterns might differ from the Wednesday's that we have considered so far; (iii) penalizing server idle time; (iv) measuring wait time from appointment time (vs. arrival time). Finally, we also provide a comparison with DDR for a scaled-down, simplified experimental setup using DFCI data. For consistency and to have some common benchmark, we conduct all our robustness checks for Wednesdays and for the ninth floor, to the extent possible.

- *Limited data* To assess the robustness of the IS approach when limited data is available, we repeat the experiments we conducted for $\sim 85\%$ utilization, by making available to IS only a subset of the available data. The benchmark remains the means-based approach that has access to all data.[‡] When 50% of data was available, the IS cost reduction ranged between 24–33%, approximately, depending on the wait time/overtime tradeoff parameter. As more data was made available, IS had access to more accurate probabilistic distributions and achieved a higher cost

[‡] Instead of making available to the means-based approach the same subset of the data as well, we opted for having a fixed benchmark so as to focus on the robustness of the IS approach.

reduction, eventually ranging between 37–41%. These findings indeed illustrate that IS could fare favorably even with somewhat noisy estimates of service durations/punctuality, while benefiting further from additional data. Additional details appear in Appendix C.2.

- *Different day of the week* We repeat our experiments using data that was collected on Fridays and processed in the exact same way as was the data in our experiments for Wednesdays. Of note, patient arrival punctuality on Fridays was less variable (standard deviation of punctuality was roughly 15% higher on Wednesdays). Across different utilization levels and overtime cost parameters we considered, the IS cost reduction was in the order of 20%, ranging between 10–30%, approximately. These findings illustrate that IS maintains significant cost reductions over means-based sequencing under alternative demand patterns, albeit somewhat lower when the underlying variability is lower. Additional details appear in Appendix C.3.

- *Different cost definition* We repeat our experiments as before, but we now compare the simulated performance of the IS and means-based approaches under a cost function that also penalizes server idle time (besides patient wait time and overtime; see our modeling choice discussion in Section 4). At a high level, the IS cost reduction was found to be of similar order as before, ranging between 15–40%, approximately, across our considered different utilization levels, overtime cost and idle cost parameters. Specifically, server idle time was recorded under the schedules of both approaches, and was inevitably larger under the lower utilization scenarios. The overall costs then increased in absolute terms for all the schedules by an amount that scaled with the idle cost parameter. Consequently, the IS relative cost reduction was found to decrease as the idle cost parameter increased, more so under the lower utilization scenarios that involved larger idle times. Additional details appear in Appendix C.4.

- *Different wait time definition* We now measure wait time for a patient in the simulation only if that waiting occurred after the patient's *scheduled* arrival time (see our modeling choice discussion in Section 4). Note that waiting times under this alternative definition are always smaller than under our original definition. The IS cost reduction ranged between 10–40%, approximately, across different utilization levels and overtime cost parameters. In particular, performance was similar to the one before, except for lower utilization scenarios and lower overtime cost parameters, for which the IS cost advantage decreased. Among the reasons for this decrease was that under lower utilization and the alternative definition, wait times for patients were notably smaller; furthermore, because lower overtime cost parameters discounted overtime, overall costs were smaller and, in turn, there was less room for improvement. Additional details appear in Appendix C.5.

- *Comparison with DDR* We investigate whether the difference in performance between IS and DDR, elicited in the previous section using synthetic data, remains the same when using DFCI data. In particular, because of DDR's inability to deal with punctuality, sequencing and

instances that involve more than 40 patients, we assume that all patients are punctual, we fix the order of scheduling to the one observed in practice, and we only consider 40 patients. Under this scaled-down and simplified version of our experiments, we solve the scheduling problem using the DDR and IS approaches. The IS cost reduction over DDR was found to be similar as in our previous experiments, averaging 6% approximately, across different utilization levels and overtime cost parameters. Additional details appear in Appendix C.6.

7.2. Appointment Template Optimization at DFCI

The infusion appointment process at DFCI is based on *templates*, which are used to determine the times of the day that different patient types could be scheduled at. In particular, for any given day of the week, a template comprises, for each patient type, a set of appointment times. (Patient types are defined in the same way appointment types are defined in Section 7.1; that is, a patient type is defined by the distributions of treatment duration and punctuality, the corresponding disease center, and a flag whether a linked or unlinked visit is required.) A patient requesting an appointment for that day is presented with appointment times in the template that match that patient's type, if any are available. In this case, the patient books one of these times, which then becomes unavailable. If no time matching the patient's type is available, the patient is assigned to a time of a "similar" type, *i.e.*, one with similar expected duration. In the rare occasion that no times are available, the patient is "overbooked," *i.e.*, assigned to an already booked time of a matching or similar type patient. For each floor, DFCI usually maintains five templates, each of which is used weekly on each workday of the week. Patients request appointments asynchronously, typically anytime between few weeks or few days in advance, depending on their type. Therefore, the resulting schedule for each day is random.

Template appointment times must be carefully chosen, because they heavily influence resulting schedules. If requests received for a given day match that day's template patient types, for example, the resulting schedule will precisely follow the template. DFCI faces some but by no means significant variability in patient types requesting appointments for each day of the week, to the extent that roughly 70% of the requests are booked at time slots of matching type. Therefore, to a large degree, optimizing templates at DFCI resembles offline appointment sequencing, whereby one needs to decide on the appropriate template appointment times for the expected patient type requests. Strictly speaking, to arrive at an optimal template, one would need to account for potential mismatches due to variability in patient type requests. To be sure, tackling this precise problem is beyond the scope of the present paper. For our purposes, we ignore variability in patient type requests and we utilize the IS and means-based sequencing approaches to produce templates for the expected patient type requests. Using real data, we then evaluate their performance by simulating,

first, asynchronous and variable patient type requests, whereby mismatches with the template could occur; and, second, the delays and overtime for the resulting schedules. This important experiment will enable us to assess the robustness of IS in case the appointment process deviates in practice from what the model assumed.

Experimental Setup We sought to optimize DFCI's Wednesday template for the ninth floor. Using historical data from the 34 Wednesdays (see Section 5.2), we chose the total number of template appointment times to be roughly the median of total daily appointments, which was equal to 82. These 82 times were proportionally allocated to the different patient types. Punctuality and service distributions for each type were estimated as in the preceding section. The capacity profile was also set as before so that utilization was approximately 85% for the template. The overtime cost was set to $\gamma = 2$ as in Section 6.

The precise template appointment times were determined by solving the resulting offline sequencing problem instance either using IS or means-based algorithms, arriving at what we henceforth call an IS and a means-based template, respectively.

We backtested the performance of the two templates as follows. Each template was used to schedule patients on all 34 Wednesdays in our data. In particular, for each Wednesday, we retrieved the patient types that were actually scheduled on that day.[§] We then randomly permuted these types to produce 1,000 arrival orders. For each such arrival order, we simulated the appointment process in the way described above. In case a patient could be assigned to multiple times, we chose one of them randomly. Once the resulting schedule was determined, we simulated punctuality and service durations to estimate wait time and overtime costs as before. For each simulated arrival order, we considered 1,000 simulation runs of its resulting schedule. This process enabled us to calculate and compare, for each of the 34 Wednesdays, the average wait time and overtime costs when the IS template or the means-based template were used.

Results Compared with the means-based template, the IS template resulted in a relative cost reduction of 58% on average, ranging roughly between 20% to 80%. Table 6 reports for each day the percentage cost reduction of IS relative to the means-based template, alongside the day's utilization and the average mismatch between patient types and their assigned appointment time types.

The results illustrate that IS maintains its edge over means-based algorithms. For utilization levels above 85%, relative cost reduction is close to 40%, which is consistent with the results we obtained for the offline sequencing experiments in the preceding section (cf. Table 5, floor 9, $> 85\%$ utilization, $\gamma = 2$). For lower utilization levels, IS produced higher cost reduction. The performance of IS in these realistic experiments illustrates its robustness, its potential to be used in practice, and that it could also serve as a useful building block in the design of online appointment algorithms.

[§] The prevalence of appointment being booked and subsequently cancelled is typically very low for oncology services. Therefore, we ignored such occurrences.

Table 6 Mean percentage differences (in %) between IS and MB solutions for means for 34 Wednesdays using a template with 82 appointments. The average mismatch column provides a mean fraction of patients that were given an appointment that corresponds to a different patient class.

Day index	Utilization	Relative cost, %	Avg. mismatch
8	0.62	−77.92	0.21
3	0.67	−77.39	0.32
28	0.67	−73.67	0.24
5	0.70	−70.75	0.32
24	0.71	−74.00	0.33
11	0.71	−62.96	0.33
31	0.73	−74.37	0.25
10	0.75	−61.11	0.33
33	0.77	−70.75	0.40
22	0.78	−69.52	0.37
4	0.78	−60.15	0.43
34	0.78	−67.11	0.40
12	0.79	−70.07	0.32
7	0.79	−65.96	0.39
27	0.80	−65.94	0.36
18	0.80	−59.27	0.37
14	0.81	−64.06	0.35
6	0.81	−64.33	0.34
16	0.82	−58.84	0.37
25	0.83	−57.40	0.31
30	0.83	−55.94	0.36
32	0.84	−55.52	0.37
23	0.85	−49.74	0.32
21	0.85	−49.23	0.35
26	0.86	−51.88	0.33
29	0.86	−47.78	0.37
19	0.86	−43.25	0.32
2	0.88	−42.82	0.32
1	0.88	−45.37	0.40
17	0.88	−43.05	0.34
20	0.88	−42.95	0.33
9	0.90	−35.13	0.29
13	0.92	−29.46	0.29
15	0.95	−20.19	0.37
Mean	0.80	−57.58	0.34
Std	0.08	14.13	0.05
Min	0.62	−77.92	0.21
Max	0.95	−20.19	0.43

8. Concluding Remarks and Further Research

In this paper, we considered appointment scheduling and sequencing under a time-varying number of servers, in a data-rich environment where service durations and punctuality are uncertain. Based on infinite-server queues and a CLT-type approximation, we proposed a data-driven approach that can accommodate hundreds of jobs and servers. To test for practical performance, we conducted

extensive numerical studies, using both synthetic and real data. In particular, we leveraged a unique dataset from the Dana-Farber Cancer Institute, that combines real-time locations, electronic health records and appointments log. Focusing on one of the center's infusion units, we found our approach able to reduce expected waiting and overtime cost in the order of 15%-40% consistently, under a wide range of experimental setups. Due to the underlying CLT approximation and based on our numerical experiments, we expect our approach to work particularly well when the average number of busy servers (the product of utilization with number of servers) is no less than a dozen, approximately.

We remark that, in this paper, we focused on the offline appointment sequencing problem: the list of patients that are to be scheduled for a given day is known in advance. As future work, it is of theoretical and practical interest to consider an online version of the problem, where patients make appointments asynchronously. In Figure 8, we also plot the overtime-waiting tradeoff (red curve) for the myopic (online) version of the infinite-server heuristic. That is, patients are scheduled in a greedy manner one by one, without any ability to modify appointment times of already scheduled patients. The presented averages are based on 4000 samples, where for each sample patients arrive in random order. The gap between the offline and myopic versions of the algorithm is due to the non-anticipatory nature of the myopic algorithm. (The average relative increase in total cost, compared to the offline version, is approximately 32%.) This gap can be hopefully reduced by employing an online algorithm that anticipates future appointment demand—we leave this topic for a natural and worthy followup research. Additionally, one could also evaluate the impact of patient preferences on the overall performance of appointment-based systems. Such preferences were elicited from patients in Liu et al. (2017), in order to model their willingness to wait for their preferred doctor, as opposed to an earlier appointment by an alternative doctor.

Another interesting line of research would be to derive an analytical characterization of the difference between the actual occupancy process and the IS approximation. This might shed further light on technical conditions under which our method would work well.

Finally, we aspire to deploy IS and to run a field experiment at DFCI, given the promising results that our research yielded. To this end, we plan to collect new infusion operations data at DFCI in order to establish a non-intervention benchmark and also to calibrate our IS models using the most up-to-date data. The results from such an implementation would provide further evidence for the value of our work.

References

Ahmadi-Javid A, Jalali Z, Klassen K (2017) Outpatient appointment systems in healthcare: A review of optimization studies. *Eur. J. Oper. Res.* 258(1):3–34. 2

Bailey N (1952) A study of queues and appointment systems in hospital out-patient departments, with special reference to waiting times. *J. Royal Stat. Soc.* 14:185–199. 2

Bandi C, Bertsimas D (2012) Tractable stochastic analysis in high dimensions via robust optimization. *Math. Program.* 134(1):23–70. 6.1

Barysauskas C, Hudgins G, Gill K, Camuso K, Bagley J, Rozanski S, Kadish S (2016) Measuring chemotherapy appointment duration and variation using real-time location systems. *J. Healthc. Qual* 38(6):353–358. 1

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

Ben-Tal A, El Ghaoui L, Nemirovski A (2009) *Robust Optimization* (Princeton University Press). 2

Berg B, Denton B (2012) Appointment planning and scheduling in outpatient procedure centers. Hall R, ed., *Handbook of Healthcare System Scheduling*, volume 168 of *International Series in Operations Research & Management Science*, chapter 6, 131–154 (Springer). 1, 2, 7

Birge J, Louveaux F (1997) *Introduction to Stochastic Programming* (New York: Springer). 2

Bravo F, Braun M, Farias V, Levi R, Lynch C, Tumolo J, White R (2017) Optimization-driven framework to understand healthcare networks cost and resource allocation, preprint. 1

Cardoen B, Demeulemeester E, Belien J (2010) Operating room planning and scheduling: A literature review. *Eur. J. Oper. Res.* 201(3):921–932. 2

Castaing J, Cohn A, Denton B, Weizer A (2016) A stochastic programming approach to reduce patient wait times and overtime in an outpatient infusion center. *IIE Trans. Healthc. Syst. Eng.* 6(3):111–125. 2

Cayirli T, Veral E (2003) Outpatient scheduling in health care: A review of literature. *Prod. and Oper. Management* 12(4):519–549. 2

Dafny L, Duggan M, Ramanarayanan S (2012) Paying a premium on your premium? Consolidation in the US health insurance industry. *Am. Econ. Rev* 102(2):1161–1185. 1

Deng Y, Shen S (2016) Decomposition algorithms for optimizing multi-server appointment scheduling with chance constraints. *Math. Program., Ser. B* 157(1):245–276. 2

Deng Y, Shen S, Denton B (2017) Chance-constrained surgery planning under uncertain or ambiguous surgery duration, working paper. 2

Denton B, Gupta D (2003) A sequential bounding approach for optimal appointment scheduling. *IIE Trans.* 35(11):1003–1016. 1, 1, 1, 2, 6.2

Denton B, Viapiano J, Vogl A (2007) Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health Care Management Sci.* 10(1):13–24. 2

Dunn P, Ghobadi K, Lennes I, Levi R, Marshall A, Rieb W, Zenteno C (2017) Real-time outpatient scheduling with patient choice, preprint. 2

Gocgun Y, Puterman M (2014) Dynamic scheduling with due dates and time windows: An application to chemotherapy patient appointment booking. *Health Care Manag. Sci.* 17(1):60–76. 2

Gupta D, Denton B (2008) Appointment scheduling in health care: Challenges and opportunities. *IIE Trans.* 40(9):800–819. 2

Hooker J, Ottosson G (2003) Logic-based Benders decomposition. *Math. Program.* 96(1):33–60. A

Jackson R (1964) Design of an appointment system. *Oper. Res. Quarterly* 15:219–224. 2

Kaandorp G, Koole G (2007) Optimal outpatient appointment scheduling. *Health Care Manage Sci.* 10(3):217–229. 2

Kim SH, Vel P, Whitt W, Cha W (2015) Poisson and non-Poisson properties in appointment-generated arrival processes: The case of an endocrinology clinic. *Oper. Res. Lett.* 43(3):247–253. 2

Kim SH, Whitt W, Cha W (2017) A data-driven model of an appointment-generated arrival process at an outpatient clinic, working paper. 2

Klassen K, Yoogalingam R (2014) Strategies for appointment policy design with patient unpunctuality. *Decision Sciences* 45(5):881–911. 4

Kong Q, Lee CY, Teo CP, Zheng Z (2013) Scheduling arrivals to a stochastic service delivery system using copositive cones. *Oper. Res.* 61(3):711–726. 1, 1, 1, 1, 2, 4, 6.2, 6.2, 6.3, B.1

Kong Q, Lee CY, Teo CP, Zheng Z (2016) Appointment sequencing: Why the smallest-variance-first rule may not be optimal. *Eur. J. Oper. Res.* 255(3):809–821. 2

Liu N, Finkelstein S, Kruk M, Rosenthal D (2017) When waiting to see a doctor is less irritating: Understanding patient preferences and choice behavior in appointment scheduling. *Management Sci.* To appear. 8

Liu Y, Whitt W (2011) A network of time-varying many-server fluid queues with customer abandonment. *Oper. Res.* 59(4):835–846. ∗

Mak HY, Rong Y, Zhang J (2014) Sequencing appointments for service systems using inventory approximations. *Manufacturing Service Oper. Management* 16(2):251–262. 2

Mak HY, Rong Y, Zhang J (2015) Appointment scheduling with limited distributional information. *Management Sci.* 61(2):316–334. 1, 1, 1, 2, 4, 6.2

Mancilla C, Storer R (2012) A sample average approximation approach to stochastic appointment sequencing and scheduling. *IIE Trans.* 44(8):655–670. 2

Pinedo M (2009) *Planning and Scheduling in Manufacturing and Services* (New York: Springer), 2nd edition. 1

Rieb W (2015) *Increasing patient throughput in the MGH cancer center infusion unit.* Master's thesis, Massachusetts Institute of Technology, Cambridge, MA. 3

Santibáñez P, Aristizabal R, Puterman M, Chow V, Huang W, Kollmannsberger C, Nordin T, Runzer N, Tyldesley S (2012) Operations research methods improve chemotherapy patient appointment scheduling. *Jt. Comm. J. Qual. Patient Saf.* 38(12):541–553. 1, 2

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

White J, Pike M (1964) Appointment systems in out-patients' clinics and the effect of patients' unpunctuality. *Medical Care* 2:133–142. 2

Zacharias C, Armony M (2017) Joint panel sizing and appointment scheduling in outpatient care. *Management Sci.* 63(11):3978–3997. 2

Zacharias C, Pinedo M (2014) Appointment scheduling with no-shows and overbooking. *Prod. and Oper. Management* 23(5):788–801. 2

Zacharias C, Pinedo M (2017) Managing customer arrivals in service systems with multiple servers. *Manufacturing Service Oper. Management* 19(4):639–656. 2

Appendix A: Data-Driven Robust Scheduling

We provide a tractable reformulation of the DDR problem (7) and develop a Benders decomposition approach that provably solves it to optimality. To ease exposition, we assume first that $\delta_{ij} > 0$ for all $i = 1, \ldots, n$ and $j = 1, \ldots, J$, and then relax this assumption to accommodate no shows. We begin by considering the adversary's problem.

Adversary's problem: Worst-case durations Given an appointment schedule a, the adversary's problem is to pick the service durations that maximize the incurred cost. To derive a strong formulation for this problem, we map each schedule time variable a_i to $\bar{T}$ binary variables α_{it}, $t = 1, \ldots, \bar{T}$, that indicate whether service i has been scheduled by time t or not. Viewed differently, variable α_{it} indicates whether customer i is available for service at time t. Clearly, there is a one-to-one mapping between the a and α variables:

$$\alpha_{it} = 1_{\{a_i \le t\}}, \quad a_i = \sum_{\tau=1}^{\bar{T}} \tau \cdot (\alpha_{i\tau} - \alpha_{i,\tau-1}), \quad i = 1, \ldots, n,\ t = 1, \ldots, \bar{T}, \tag{8}$$

where $\alpha_{i0} := 0$ for all i.

The main decision variables for the adversary are the service duration assignment variables u; the actual durations can be retrieved using (6). Let b_{it} and e_{it} be auxiliary decision variables that indicate whether service for the ith customer has begun and ended, respectively, by time t, in a similar fashion as the availability variables α. The adversary's problem can then be formulated as an integer optimization problem:

$$\begin{aligned}
W(\alpha) := \text{maximize} \quad & \sum_{i=1}^{n}\sum_{t=1}^{\bar{T}}(\alpha_{it} - b_{it}) + \gamma \sum_{i=1}^{n}\sum_{t=T+1}^{\bar{T}}(b_{it} - e_{it}) && \text{(9a)}\\
\text{subject to} \quad & b \le \alpha && \text{(9b)}\\
& e \le b && \text{(9c)}\\
& b_{it} \le b_{i,t+1}, \quad i = 1, \ldots, n,\ t = 1, \ldots, \bar{T} - 1 && \text{(9d)}\\
& e_{it} \le e_{i,t+1}, \quad i = 1, \ldots, n,\ t = 1, \ldots, \bar{T} - 1 && \text{(9e)}\\
& b_{it} \le b_{i-1,t}, \quad i = 2, \ldots, n,\ t = 1, \ldots, \bar{T} && \text{(9f)}\\
& \sum_{i=1}^{n}(b_{it} - e_{it}) \le c_t, \quad t = 1, \ldots, \bar{T} && \text{(9g)}\\
& \sum_{i=1}^{n}(b_{it} - e_{it}) \ge f_t c_t, \quad t = 1, \ldots, \bar{T} && \text{(9h)}\\
& \sum_{i=1}^{n}(\alpha_{it} - b_{it}) \le n f_t, \quad t = 1, \ldots, \bar{T} && \text{(9i)}\\
& \sum_{t=1}^{\bar{T}}(b_{it} - e_{it}) = \sum_{j=1}^{J}\delta_{ij}u_{ij}, \quad i = 1, \ldots, n && \text{(9j)}\\
& u \in \mathcal{U} && \text{(9k)}\\
& b, e, f \quad \text{binary}, && \text{(9l)}
\end{aligned}$$

with aforementioned variables $u \in \{0,1\}^{n \times J}$, $b \in \{0,1\}^{n \times \bar{T}}$, $e \in \{0,1\}^{n \times \bar{T}}$, and variables $f \in \{0,1\}^{\bar{T}}$ that indicate whether all c_t servers are busy at time t. Constraints (9b-9c) ensure that service begins after customers are available and ends after it has started, respectively. Constraints (9d-9e) enforce time connectivity: if

service for customer i has begun (ended) by time t, then b_{it} (e_{it}) has a value of one for all later time periods. Constraint (9f) enforces first-come first-served policy. Constraint (9g) is a capacity constraint. In conjunction with it, constraint (9h) ensures that $f_t = 1$, if c_t servers are busy at time t. Constraint (9i) allows waiting only if $f_t = 1$. Finally, constraint (9j) sets the duration of each service to be as assigned by variables u.

Note that $W(\alpha)$ is the optimal value of (9). That is, if the appointment schedule is α, $W(\alpha)$ is the worst-case waiting plus overtime cost incurred.

Data-driven Robust appointment scheduling We equivalently use the availability variables α as the decision variables to reformulate DDR problem (7); schedule times a can then be retrieved using (8).

$$\begin{aligned}
\text{minimize} \quad & W(\alpha) && && \text{(10a)}\\
\text{subject to} \quad & \alpha_{it} \le \alpha_{i,t+1}, && i = 1,\dots,n,\ t = 1,\dots,\bar{T}-1 && \text{(10b)}\\
& \alpha_{it} \le \alpha_{i-1,t}, && i = 2,\dots,n,\ t = 1,\dots,\bar{T} && \text{(10c)}\\
& \alpha_{it} = 1, && i = 1,\dots,n,\ t = T,\dots,\bar{T} && \text{(10d)}\\
& \alpha \quad \text{binary.} && && \text{(10e)}
\end{aligned}$$

Constraint (10b) enforces time connectivity, (10c) enforces the scheduling sequence, and (10d) ensures that all appointments are scheduled during regular business hours.

Next, we illustrate how to solve Problem (10) to optimality using Benders decomposition. To ease notation, let $\mathcal{A}$ be the set of feasible solutions for (10), that is,

$$\mathcal{A} = \{\alpha : \alpha \text{ satisfy constraints } (10b - 10e)\}.$$

Our Benders decomposition procedure works as follows. Given a solution $\bar{\alpha}$ to (10), let $d(\bar{\alpha})$ be worst-case durations that are optimal for nature's problem (9). We apply a Benders cut $z \ge \beta_{\bar{\alpha}}(\alpha)$, where

$$\begin{aligned}
\beta_{\bar{\alpha}}(\alpha) := \text{minimize} \quad & \sum_{i=1}^{n}\sum_{t=1}^{\bar{T}}(\alpha_{it} - b_{it}) + \gamma \sum_{i=1}^{n} \sum_{t=T\wedge d_i(\bar{\alpha})+1}^{\bar{T}} (b_{it} - b_{i,t-d_i(\bar{\alpha})}) && \text{(11a)}\\
\text{subject to} \quad & b \le \alpha && \text{(11b)}\\
& b_{it} \le b_{i,t+1}, \quad i = 1,\dots,n,\ t = 1,\dots,\bar{T}-1 && \text{(11c)}\\
& b_{it} \le b_{i-1,t}, \quad i = 2,\dots,n,\ t = 1,\dots,\bar{T} && \text{(11d)}\\
& \sum_{i=1}^{n}(b_{it} - b_{i,t-d_i(\bar{\alpha})}) \le c_t, \quad t = 1,\dots,\bar{T} && \text{(11e)}\\
& b \quad \text{binary,} && \text{(11f)}
\end{aligned}$$

where $b_{it} := 0$, for all $i = 1,\dots,n$, and $t \le 0$. Below we provide a detailed outline of the procedure we use, together with a proof of its convergence.

PROPOSITION 1. *Procedure 1 terminates after a finite number of steps.*

Proof. We first show that

$$W(\alpha) \ge \beta_{\bar{\alpha}}(\alpha), \quad \forall \alpha, \bar{\alpha} \in \mathcal{A}.$$

Procedure 1 Benders decomposition approach to solve (10)

Input: initial $\alpha^0 \in \mathcal{A}$

Output: optimal solution $\alpha^\star$ to (10)

$\bar{z} \leftarrow -\infty$, $k \leftarrow 0$

while $W(\alpha^k) > \bar{z}$ **do**

solve

$$\text{minimize} \quad z \tag{12a}$$

$$\text{subject to} \quad z \geq \beta_{\alpha^j}(\alpha), \quad j = 0, \ldots, k \tag{12b}$$

$$\alpha \in \mathcal{A} \tag{12c}$$

$k \leftarrow k+1$

let $\tilde{\alpha}$, $\tilde{z}$ be an optimal solution to (12)

$a^k \leftarrow \tilde{a}$

$\bar{z} \leftarrow \tilde{z}$

end while

$\alpha^\star \leftarrow \alpha^k$

To this end, fix some $\alpha, \bar{\alpha} \in \mathcal{A}$, and let b be an optimal solution to (11). We now construct a feasible solution to (9) that achieves an objective value equal to $\beta_{\bar{\alpha}}(\alpha)$. In particular, let

$$e_{it} = \begin{cases} b_{i,t-d_i(\bar{\alpha})}, & t = d_i(\bar{\alpha})+1, \ldots, \bar{T} \\ 0, & \text{otherwise,} \end{cases} \quad \text{and} \quad u_{ij} = 1_{\{\delta_{ij} = d_i(\bar{\alpha})\}}, \quad j = 1, \ldots, J,$$

for all $i = 1, \ldots, n$. We argue that there exists an $f \in \{0,1\}^{\bar{T}}$ such that (u, b, e, f) is feasible for (9). Feasibility of b for (11) implies (9b, 9d, 9f). By the definition of e, we also get that (9c, 9e, 9g) hold trivially. To show that there exists an $f \in \{0,1\}^{\bar{T}}$ such that (u, b, e, f) is feasible for (9), we need to argue that, for all $t = 1, \ldots, \bar{T}$,

$$\sum_{i=1}^{n} (\alpha_{it} - b_{it}) > 0 \quad \text{implies} \quad \sum_{i=1}^{n} (b_{it} - e_{it}) = c_t.$$

For the sake of reaching a contradiction, assume that this condition fails for some time k. Then,

$$\sum_{i=1}^{n} (b_{ik} - e_{ik}) < c_k,$$

and there exists a service index I such that

$$\alpha_{Ik} = 1 \quad \text{and} \quad b_{Ik} = 0.$$

If multiple such indices exist, let I be the smallest among them. In other words, the Ith customer is available at time k, but he starts getting serviced only at some later time $\tau > k$. That is,

$$\tau = \min\{t : b_{It} = 1\}.$$

Consider now $\tilde{b}$ such that customer I starts getting serviced at $\tau-1$ instead, that is

$$\tilde{b}_{I,\tau-1}=1 \quad \text{and} \quad \tilde{b}_{it}=b_{it},\ i\neq I,\ t\neq\tau-1.$$

Note that $\tilde{b}$ is feasible for (11). Its first constraint is satisfied since $\alpha_{I,\tau-1}\geq\alpha_{Ik}=1$, which follows from $\alpha\in\mathcal{A}$ and $\tau>k$. The second constraint follows from the feasibility of the original solution b. The third is relevant only if $I>1$ and follows from $b_{I-1,\tau-1}\geq b_{I-1,k}=1$, which is true since I was picked as the smallest index for which $b_{ik}=0$. Finally, the capacity constraint is satisfied since there is slack capacity at $k,\ldots,\tau-1$, given that no new service starts at these periods. Given that $\tilde{b}$ achieves a strictly lower objective value for (11), this contradicts the optimality of b.

We now show that

$$W(\alpha)=\beta_\alpha(\alpha),\quad \alpha\in\mathcal{A}.$$

To this end, let $W(\alpha,\bar{d})$ be the optimal value of (9), where $\bar{d}$ is a service duration vector that corresponds to some assignment vector $\bar{u}\in\mathcal{U}$, when u is constrained to equal $\bar{u}$. Similarly, let $\beta(\alpha,\bar{d})$ be the optimal value of (11), when $d(\bar{\alpha})$ is equal to $\bar{d}$. As a first step, note that $W(\alpha,\bar{d})=\beta(\alpha,\bar{d})$. To see this, recall we argued above that, for any $\alpha\in\mathcal{A}$, if b is feasible for (11) then there exists a feasible solution to (9) that achieves the same objective, with the same service duration vector. Thus, $W(\alpha,\bar{d})\geq\beta(\alpha,\bar{d})$. Conversely, if $(\bar{u},b,e,f)$ is feasible for (9), then we have that $e_{it}=b_{i,t+\bar{d}_i}$ by (9j). Thus, b is feasible for (11), with the same durations vector, achieving the same objective value. Consequently, $W(\alpha,\bar{d})\leq\beta(\alpha,\bar{d})$. To complete our argument, note that

$$W(\alpha)=W(\alpha,d(\alpha))=\beta(\alpha,d(\alpha))=\beta_\alpha(\alpha).$$

The statement of the proposition follows from Theorem 2 in Hooker and Ottosson (2003). □

Relaxing the assumption $\delta_{ij}>0$ involves only some straightforward modifications of problems (9) and (11). In particular, one possible way it to introduce auxiliary variables w_i, $i=1,\ldots,n$, to capture wait times of patients. Then, the first term of the objective in both problems would be replaced by $\sum_{i=1}^n w_i$. To ensure that variables w correspond to the underlying wait times, constraints

$$w_i\leq\sum_{t=1}^{\bar{T}}(\alpha_{it}-b_{it}),\quad w_i\leq\bar{T}\sum_{j=1}^{J}\delta_{ij}u_{ij}$$

would need to added to (9), for all $i=1,\ldots,n$. Similarly, if auxiliary variables $s\in\{0,1\}^n$ indicate whether patients showed up, constraints

$$w_i\geq 0,\quad w_i\geq\sum_{t=1}^{\bar{T}}(\alpha_{it}-b_{it})-\bar{T}(1-s_i),\quad \bar{T}s_i\geq d_i(\bar{\alpha})$$

would need to added to (11), for all $i=1,\ldots,n$. Under these modifications, the validity of the formulations and of Proposition 1 can be readily checked.

Appendix B: Benchmarking using DDR

B.1. Single-Server Problem

We employ an experimental setup similar to Kong et al. (2013). In particular, we consider scheduling $n=7$, 12 or 15 customers to be served by a single server. Punctuality is perfect, *i.e.*, $P_i=0$ with probability one. Service durations are random and follow either a lognormal or gamma distribution. The mean duration of the ith customer, d_i, is uniformly drawn over $[2/3, 4/3]$. Conditional on its mean d_i, the standard deviation of each duration is uniformly drawn over $\left[2d_i/(3\sqrt{3}),\, 4d_i/(3\sqrt{3})\right]$, so that the coefficient of variation is uniform on $[2/3, 4/3]$ (as in Kong et al. (2013)). Business hours are taken to be $T=\sum_{i=1}^{n} d_i/\rho$, where ρ is a utilization parameter taking values $\rho=0.85$, 1.00 or 1.15. Overtime cost rate is $\gamma=2$ and $\gamma=3$. The DDR parameters are $\Gamma=0.5$ and $J=7$.

For each possible combination of distribution, number of customers n, utilization parameter ρ, and overtime cost rate γ, we generate 20 instances and produce schedules according to the DDR, SO and DR approaches. Various statistics for resulting costs of each approach were obtained by simulating 10^6 paths. Table 7 displays a relative comparison of DDR's mean, 75-, 85- and 95-percentile costs with SO's and DR's, only for the case of $\gamma=2$; results for $\gamma=3$ are very similar. As remarked in Section 6.2, our experiments suggest that the performance of DDR is near-optimal for single-server problems, in comparison with DR and SO.

B.2. Multiple-Server Problem

B.2.1. Comparison with Means-based Scheduling We first outline the MB scheduling approach we used in the experimental setup described in Section 6.2 for multiple-server problems. Assume given mean service durations for each customer, denoted by $d_1,\ldots,d_n$. Our MB approach borrows from the DDR approach we outlined in Appendix A:

$$\begin{aligned}
\text{minimize}\quad & \sum_{i=1}^{n}\sum_{t=1}^{\bar{T}}(\alpha_{it}-b_{it})+\gamma\sum_{i=1}^{n}\sum_{t=T+1}^{\bar{T}}(b_{it}-e_{it})\\
\text{subject to}\quad & (9b)-(9g)\\
& \sum_{t=1}^{\bar{T}}(b_{it}-e_{it})=d_i,\quad i=1,\ldots,n\\
& \alpha\in\mathcal{A},\quad b,\,e\ \text{ binary},
\end{aligned}$$

with variables $\alpha\in\{0,1\}^{n\times\bar{T}}$, $b\in\{0,1\}^{n\times\bar{T}}$, $e\in\{0,1\}^{n\times\bar{T}}$. An optimal schedule a is retrieved using (8).

Table 8 displays a relative comparison of DDR's mean, 75-, 85- and 95-percentile costs with MB's for the experimental setup described in Section 6.2 for multiple-server problems. As remarked in Section 6.2, our experiments suggest that DDR provides substantial cost reductions in comparison with MB scheduling.

B.2.2. Comparison with IS We implemented IS using MATLAB R2016a, and DDR using Gurobi 6.5.2. The CPU used to measure computation time was Intel Xeon CPU X5660.

Table 9 reports the computation times of DDR and IS employed in the experimental setup in Section 6.3. As we remarked, the required computation time of IS is smaller by one to two orders of magnitude. Furthermore, we considered two extensions: in the first we allowed for punctuality, and in the second we increased

Table 7 Mean percentage differences (in %) between DDR and SO solutions (DR solutions) for means and percentiles.

Mean utilization	Performance measure	Distribution	Number of jobs 7	12	15
0.85	Mean	Lognormal	+1.8 (−0.1)	+3.3 (−4.6)	+3.7 (−3.9)
		Gamma	+1.9 (−0.2)	+3.0 (−4.3)	+2.6 (−3.7)
	75% percentile	Lognormal	+2.3 (+1.5)	+3.9 (+0.0)	+2.8 (−1.1)
		Gamma	+2.1 (+2.0)	+2.6 (−0.6)	+1.8 (−1.0)
	85% percentile	Lognormal	+2.0 (+3.8)	+5.9 (+8.2)	+5.2 (+6.4)
		Gamma	+3.6 (+4.8)	+4.3 (+6.1)	+2.7 (+4.0)
	95% percentile	Lognormal	+1.4 (+2.7)	+6.8 (+14.0)	+7.1 (+13.7)
		Gamma	+3.4 (+3.3)	+6.1 (+12.5)	+4.0 (+9.5)
1.00	Mean	Lognormal	+1.4 (−3.6)	+2.2 (−3.4)	+2.3 (−3.2)
		Gamma	+1.6 (−1.4)	+2.0 (−2.6)	+2.8 (−1.7)
	75% percentile	Lognormal	+1.4 (+5.4)	+2.5 (+1.6)	+2.7 (+2.0)
		Gamma	+0.6 (+4.5)	+1.9 (+1.5)	+1.8 (+1.4)
	85% percentile	Lognormal	+1.3 (+9.7)	+2.4 (+6.3)	+3.3 (+7.4)
		Gamma	+0.3 (+6.8)	+2.1 (+5.3)	+1.9 (+4.6)
	95% percentile	Lognormal	+0.8 (+8.3)	+2.0 (+7.0)	+3.1 (+9.1)
		Gamma	−0.0 (+5.3)	+1.8 (+5.9)	+1.4 (+5.2)
1.15	Mean	Lognormal	+0.8 (−1.4)	+1.5 (−2.9)	+1.4 (−2.6)
		Gamma	+1.0 (−0.9)	+1.4 (−2.0)	+1.5 (−2.0)
	75% percentile	Lognormal	+0.6 (+1.8)	+1.6 (+1.7)	+1.1 (+2.0)
		Gamma	+0.4 (+1.2)	+0.9 (+1.5)	+0.1 (+0.7)
	85% percentile	Lognormal	+0.4 (+3.2)	+1.8 (+4.9)	+1.1 (+4.8)
		Gamma	+0.1 (+1.7)	+0.6 (+3.5)	−0.7 (+2.1)
	95% percentile	Lognormal	+0.0 (+2.7)	+1.4 (+4.7)	+0.9 (+4.6)
		Gamma	−0.2 (+1.0)	+0.5 (+3.1)	−0.9 (+1.6)

Table 8 Mean percentage differences (in %) between DDR and MB solutions for means and percentiles.

Mean utilization	Performance measure	Distributions Gamma	Lognormal
0.85	Mean	−11.7	−9.0
	75% percentile	−11.0	−8.3
	85% percentile	−11.4	−8.5
	95% percentile	−10.8	−7.9
1.00	Mean	−11.7	−9.7
	75% percentile	−10.5	−8.9
	85% percentile	−9.0	−7.7
	95% percentile	−6.2	−5.3
1.15	Mean	−10.4	−11.0
	75% percentile	−8.2	−8.4
	85% percentile	−6.2	−6.2
	95% percentile	−3.3	−3.1

Table 9 Mean computation times (standard deviation) in minutes of DDR and IS for the experimental setup in Section 6.3.

Mean utilization	Distribution	DDR	IS
0.85	Lognormal	170 (70)	2.4 (0.2)
	Gamma	145 (14)	2.4 (0.1)
1.00	Lognormal	165 (64)	2.1 (0.2)
	Gamma	139 (45)	2.1 (0.1)
1.15	Lognormal	32 (16)	2.2 (0.2)
	Gamma	47 (27)	2.2 (0.3)

Table 10 Mean computation times (standard deviation) in minutes of IS for the extensions of the experimental setup in Section 6.3.

Mean utilization	Distribution	Punctuality	$n=100$
0.85	Lognormal	3.3 (0.2)	26.5 (2.2)
	Gamma	3.4 (0.2)	25.2 (2.1)
1.00	Lognormal	2.6 (0.1)	28.4 (2)
	Gamma	2.6 (0.1)	29.3 (2.7)
1.15	Lognormal	2.7 (0.1)	33.2 (2)
	Gamma	2.6 (0.1)	31.4 (2.6)

the number of customers to $n=100$. DDR failed to compute within four hours for both extensions. The computation times for IS are reported in Table 10.

As a closing remark, please note that DDR was solved using a commercial-grade software implementation (Gurobi), and therefore speed ups, if possible, would require further research in identifying sharper formulations. In contrast, we expect a commercial-grade software implementation of IS to significantly reduce the reported computation times.

Appendix C: Implementation at DFCI

C.1. Means-based Sequencing Approach

Assume given mean service durations for each customer, denoted by $d_1,\ldots,d_n$, and a capacity vector c_t, $t=1,\ldots,\bar{T}$. At a high level, our means-based approach decides on schedule times $a_1,\ldots,a_n$, so as to minimize makespan, namely the time it takes to serve all customers. Such a schedule would be tailored towards minimizing overtime costs. To obtain solutions that would trade off overtime with waiting costs, we scale down by a constant factor the available capacity during business hours c_t, $t=1,\ldots,T$, and re-solve. In so doing, the re-optimized schedule allows for slack capacity to mitigate waiting at the expense of larger makespan (*i.e.*, possible overtime).

Our means-based sequencing formulation models queueing dynamics in a similar fashion to our DDR:

$$\begin{aligned}
\text{maximize} \quad & \sum_{t=1}^{\bar{T}} x_t \\
\text{subject to} \quad & \alpha_{it} \le \alpha_{i,t+1}, \quad i=1,\ldots,n,\ t=1,\ldots,\bar{T}-1 \\
& \alpha_{it}=1, \quad i=1,\ldots,n,\ t=T,\ldots,\bar{T}
\end{aligned}$$

$$\sum_{i=1}^{n}(\alpha_{it}-\alpha_{i,t-d_i})\le c_t,\quad t=1,\dots,\bar{T}$$
$$x_t\le x_{t+1},\quad t=1,\dots,\bar{T}-1$$
$$x_t\le\frac{1}{n}\sum_{i=1}^{n}\alpha_{i,t-d_i},\quad t=1,\dots,\bar{T}$$
$$\alpha,\,x\quad\text{binary;}$$

here $\alpha_{it}:=0$, for all $i=1,\dots,n$, and $t\le 0$. Variables $\alpha\in\{0,1\}^{n\times\bar{T}}$ have the same interpretation as before. Variables $x\in\{0,1\}^{\bar{T}}$ indicate whether all customers have been served by time t.

C.2. Robustness Check: Limited Data

We repeat the experiments conducted for the ninth floor and $\sim 85\%$ utilization, as outlined in Section 7.1, by making available to IS only a subset of the available data. In particular, we make 50%, 75% or 100% of the data gathered over the 34 Wednesdays available to IS. The means-based approach has access to all data.

Figure 9 depicts the wait time/overtime cost tradeoff curves that the two approaches achieve—the three curves for IS approach correspond to the three different cases outlined above. Table 11 reports the percentage cost reductions that IS achieves for different overtime cost parameters and the three different cases we consider.

Figure 9 Overtime-waiting tradeoffs for one day on the 9th floor at DFCI. The higher curves for IS correspond to less data being available.

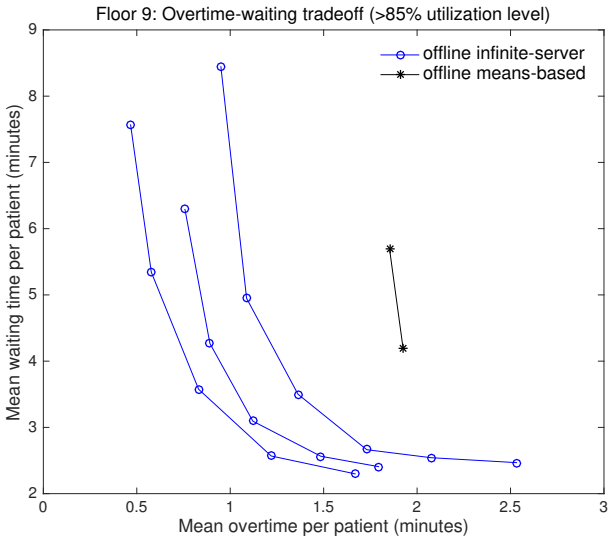


Table 11 Percentage decrease (in %) for total mean cost, between different IS solutions, which were produced using different fractions of the available training set data, and the means-based solution, which was produced using all data.

	Per-unit overtime cost γ				
Training set fraction	1/3	1/2	1	2	3
0.50	−33.0	−31.5	−28.1	−23.8	−23.8
0.75	−37.8	−35.8	−33.7	−33.6	−35.2
1.00	−40.9	−39.2	−38.0	−37.7	−39.1

C.3. Robustness Check: Different Day of the Week

We repeat the experiments conducted for the ninth floor as outlined in Section 7.1, but using data that was collected on Fridays and processed in the exact same way as was the data in our experiments for Wednesdays. In particular, a time period containing 34 Fridays in 2014 was considered (from February 14 to October 24). Two days were excluded from our analysis (March 7 and June 13) due to RTLS system interruptions on those particular days. The number of patients varied from 44 to 86 a day (the average is 63.4, while the standard deviation is 8.9). Of note, patient arrival punctuality on Fridays was less variable compared with Wednesdays (standard deviation of punctuality was roughly 15% higher on Wednesdays).

Figure 10 provides an overtime-waiting tradeoff curve for QQ-RRR-2014 (which was a Friday), on the ninth floor of DFCI. On that day, 67 patients had infusion appointments and received treatment. Table 12 reports the percentage cost decrease that the IS approach achieves, compared with the means-based approach for different overtime rates γ.

Figure 10 Overtime-waiting tradeoffs for a Friday on DFCI Floor 9. The higher curves correspond to the higher utilization level.

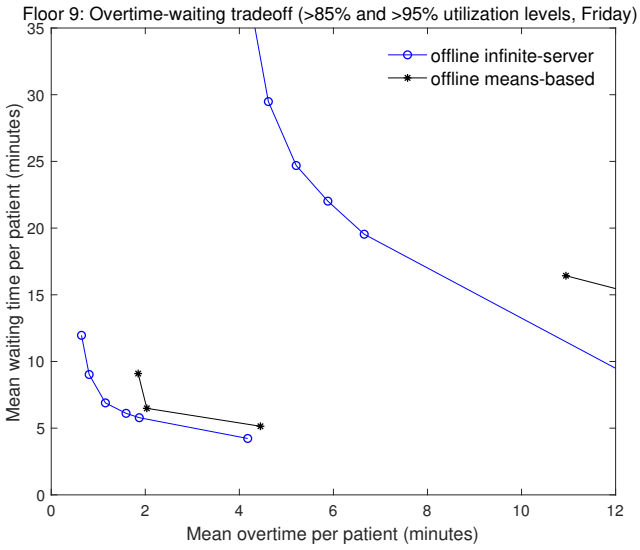


Table 12 Percentage decrease (in %) for total mean cost, between IS and means-based solutions for a Friday on DFCI Floor 9.

	Per-unit overtime cost γ				
Utilization	1/3	1/2	1	2	3
>0.85	−15.2	−14.3	−10.1	−12.8	−17.7
>0.95	−24.5	−26.4	−22.3	−14.3	−19.8

C.4. Robustness Check: Different Cost Definition

We compare the simulated performance of the appointment schedules that the IS and means-based approaches produced for Wednesday, XX-YYY-2014, on the ninth floor (as outlined in Section 7.1) under a cost function that now also penalizes server idle time, besides patient wait time and overtime. In particular, we add to the expected cost in (1) an extra term equal to βI, where β is the per-unit idle cost parameter, and I is the average idle time per server per patient. Table 13 reports the percentage cost decrease that the IS schedules achieve, compared with the means-based schedules for different overtime rates γ and idle cost parameters β.

Table 13 Percentage decrease (in %) for total mean cost, between IS and means-based solutions for DFCI Floor 9 in experiments in Section 7.1, when server idle time is included in the cost.

		Per-unit overtime cost γ				
Utilization	Per-unit idle cost β	1/3	1/2	1	2	3
>0.85	0	−40.3	−38.6	−37.3	−37.3	−38.4
	1/3	−37.6	−36.3	−35.5	−35.9	−37.2
	1/2	−36.5	−35.2	−34.6	−35.2	−36.7
	1	−33.3	−32.3	−32.3	−33.4	−35.2
	2	−28.5	−27.9	−28.6	−30.3	−32.5
	3	−24.9	−24.5	−25.7	−27.7	−30.3
>0.95	0	−31.4	−27.6	−18.7	−18.4	−22.9
	1/3	−30.9	−27.2	−18.4	−18.4	−22.9
	1/2	−30.6	−27.0	−18.2	−18.4	−22.9
	1	−29.9	−26.3	−17.8	−18.4	−22.9
	2	−28.5	−25.1	−17.0	−18.5	−23.0
	3	−27.2	−24.0	−16.2	−18.5	−23.1

C.5. Robustness Check: Different Wait Time Definition

We compare the simulated performance of the appointment schedules that the IS and means-based approaches produced for Wednesday, XX-YYY-2014, on the ninth floor (as outlined in Section 7.1), but we now measure wait time for a patient in the simulation only if that waiting occurred after the patient's scheduled arrival time. In particular, wait time for the ith patient is now measured as $(S_i - a_i - (P_i)^+)^+$. Table 14 reports the percentage cost decrease that the IS schedules achieve, compared with the means-based schedules for different overtime rates γ under this alternative wait time definition.

Table 14 Percentage decrease (in %) for total mean cost, between IS and means-based solutions for DFCI Floor 9 in experiments in Section 7.1, when wait time is measured only if it occurred after the scheduled arrival time.

	Per-unit overtime cost γ				
Utilization	1/3	1/2	1	2	3
>0.85	-9.8	-13.3	-19.8	-32.3	-38.6
>0.95	-31.8	-27.2	-20.6	-27.3	-31.3

C.6. Robustness Check: Comparison with DDR

We consider a scaled-down and simplified version of the experiment we conducted in Section 7.1 for DFCI's ninth floor on Wednesday, XX-YYY-2014, so as to compare our IS and DDR approaches. In particular, we assume that all patients are punctual, we fix the order of scheduling to the one observed on that day, and we only consider 40 patients. Capacity for that day is adjusted accordingly so that resulting utilization levels are approximately 85% and 95%. We then solve the scheduling problem using the DDR and IS approaches. The DDR approach is calibrated in the same way as in all the experiments in Section 6. Figure 11 provides an overtime-waiting tradeoff curve for the two approaches and different utilization levels. Table 15 reports the percentage cost decrease that the IS schedules achieve, compared with the DDR schedules, for different overtime rates γ.

Figure 11 Overtime-waiting tradeoffs for scheduling on a scaled-down day on DFCI Floor 9. The higher curves correspond to the higher utilization level.

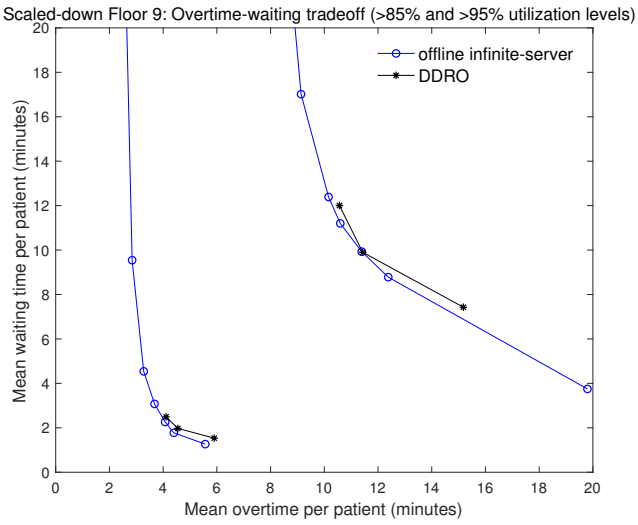


Table 15 Percentage decrease (in %) for total mean cost, between IS and DDR solutions for scheduling on a scaled-down Wednesday at DFCI Floor 9.

	Per-unit overtime cost γ				
Utilization	1/3	1/2	1	2	3
>0.85	-10.5	-6.4	-5.3	-2.8	-4.8
>0.95	-17.1	-9.1	-0.8	-1.1	-1.9