Health Care Manag Sci (2014) 17:60–76
DOI 10.1007/s10729-013-9253-z

Dynamic scheduling with due dates and time windows: an application to chemotherapy patient appointment booking

Yasin Gocgun · Martin L. Puterman

Received: 19 December 2012 / Accepted: 15 September 2013 / Published online: 10 October 2013

Abstract We study a scheduling problem in which arriving patients require appointments at specific future days within a treatment specific time window. This research is motivated by a study of chemotherapy scheduling practices at the British Columbia Cancer Agency (Canada). We formulate this problem as a Markov Decision Process (MDP). Since the resulting MDPs are intractable to exact methods, we employ linear-programming-based Approximate Dynamic Programming (ADP) to obtain approximate solutions. Using simulation, we compare the performance of the resulting ADP policies to practical and easy-to-use heuristic decision rules under diverse scenarios. The results indicate that ADP is promising in several scenarios, and that a specific easy-to-use heuristic performs well in the idealized chemotherapy scheduling setting we study.

Keywords Chemotherapy scheduling · Markov decision processes · Approximate dynamic programming

1 Introduction

This paper is motivated by an issue that arose in the redesign of chemotherapy appointment scheduling practices at the Vancouver Centre of the British Columbia Cancer Agency (BCCA). Over-constrained schedules resulted in excessive wait listing, late patient appointment notifications, pharmacy congestion, unbalanced workload between nurses and considerable clerical rework. Santibanez et al. [26] describes the development, implementation and evaluation of a re-designed two-phase appointment scheduling process and a custom scheduling optimization software package, Chemo Smartbook. In the first phase, patients are assigned appointment dates and in the second phase Chemo Smartbook assigns patients appointment times. Implementing the first phase resulted in the issue addressed by this paper, namely, how to best assign randomly arriving patients to future appointment dates within clinically established time windows.

Before elaborating on specific features of the problem we provide a simplified problem description. Throughout each working day, a random number of requests for chemotherapy appointments arrive at the scheduling office. Appointments are characterized by target dates and tolerance limits. At the end of each day, the scheduler (booking clerk) reviews daily demand and available capacity (which is limited), and schedules patients to specific days in the future, taking target dates and tolerance limits into account. Figure 1 displays the wide range of appointment types (target dates/tolerance limits) that the clerk must schedule and the relative frequency of each. In it, for example, the tolerance limit (1,2) corresponds to a tolerance limit of 1 day before and 2 days after the specific target date and the tolerance limit (0,0) corresponds to an appointment which must be scheduled on a specific target date.

The BCCA chemotherapy appointment scheduling system manages two appointment types; new patient (NP) appointments and follow-up (FU) appointments. The redesigned scheduling system reserves several slots of daily

Y. Gocgun (✉)
Centre for Maintenance Optimization Reliability Engineering,
Department of Mechanical Industrial Engineering,
University of Toronto, 5 King's College Road,
Toronto, ON M5S 3G8, Canada
e-mail: gocgun@mie.utoronto.ca

M. L. Puterman
Operations and Logistics Division, Sauder School of Business,
University of British Columbia, 2053 Main Mall Vancouver,
BC V6T 1Z2 Canada
e-mail: martin.puterman@sauder.ubc.ca

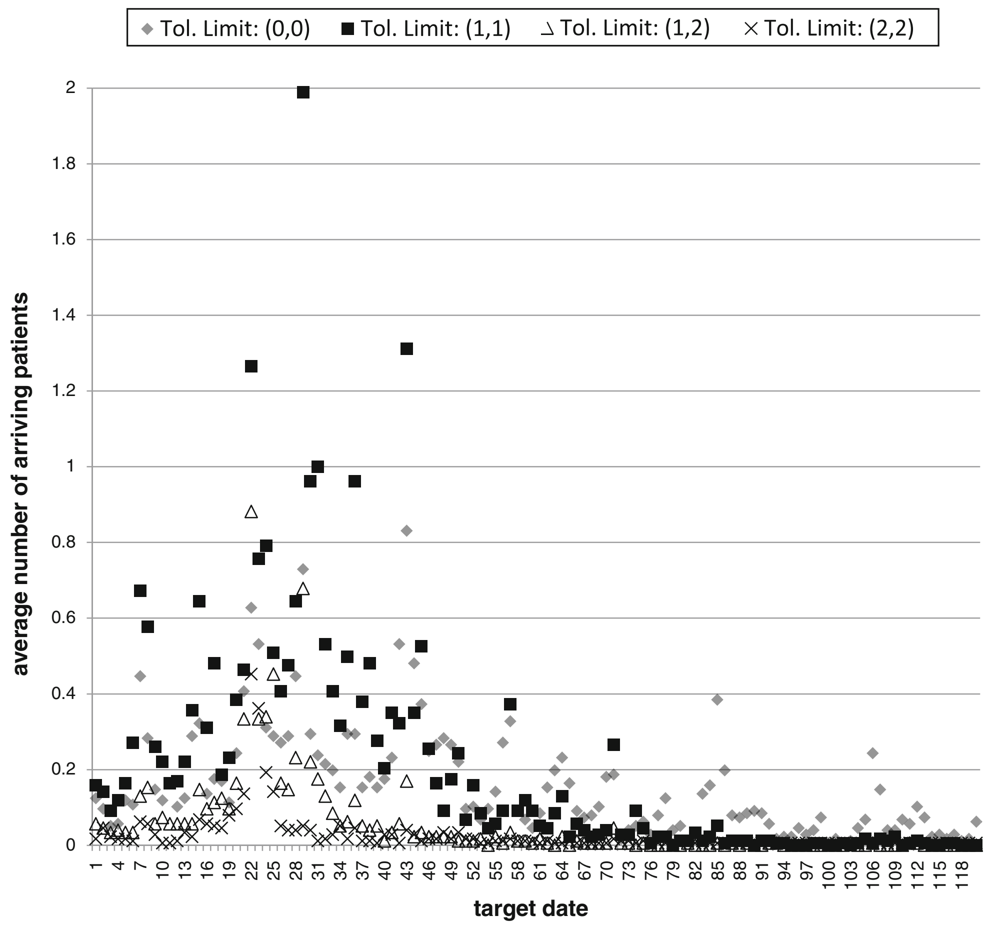

Fig. 1 Relative frequency of arriving patients dis-aggregated by tolerance and target date based on BCCA data between May 2010 and January 2011

capacity for NP appointments and the remainder for FU appointments. The goal for NP appointments is to schedule them as soon as possible hopefully conforming to urgency guidelines. Earlier work by Patrick et al. [21] applies to this situation.

The goal for FU appointments, which are the focus of this paper, is to as closely as possible follow a schedule prescribed through a treatment protocol. Treatment protocols (see Santibanez et al. [26], Table 1 for an example) are designed by clinicians to maximize the efficacy of a chemotherapy treatment. Among other things, they specify the drugs to be administered, the dosage, appointment duration, the number of days between treatments and tolerance limits. The variability in the arrival rates displayed in Fig. 1 is a result of the over 600 protocols administered by the BCCA and the relative frequency of each. For example, a treatment protocol for a specific cancer type would indicate that a treatment requires 8 sessions, they occur 14 days apart, have a tolerance of ±1 day and require an appointment of 2 h. Thus the notion of tolerance limits is a key component of the FU process; once a patient starts treatment, the protocol must be adhered to as closely as possible. Ideal would be to book all appointments on their target dates, if not, the tolerance limit allows some flexibility.

Although there is considerable variability in appointment duration; some requiring one hour while others requiring more than 8 h, we ignore this variability in the first stage

Table 1 Experimental factors and their levels

Experimental factors	Levels
Arrival pattern	Constant arrival rate
	Empirical arrival rate
Capacity	80 (55*)
	85 (60*)
	90 (65*)
Diversion cost	50 (Rigid tolerance case)
	75
	100
	125
	250
Cost function	Moderate early/late
	Sched. cost
	High early scheduling cost
	High late sched. cost

The numbers marked with an asterisk in the table are empirical case capacities of our implemented booking process. Instead we defined capacity in terms of the number of generic slots available each day over a booking horizon. This number is a result of managerial and clinical considerations. Our intent in doing this was to simplify the first phase of the booking process so as to enable patients to receive confirmed notification of the appointment date well in advance but leave notification of the appointment time to roughly a week advance when the scheduling optimizer, Chemo Smartbook is executed. Thus the first phase scheduling problem was reduced to that described above; find an appointment date within the tolerance limit. We assumed that the random nature of the arrival process would allow a good within day schedule that could be determined at the second phase. This is a form of risk pooling.

We believe the inclusion of tolerance limits is the unique feature of this problem that distinguishes it from other applications. We elaborate on this distinction in our literature review.

We note that similar problems arise in manufacturing and production management where jobs of different types must be scheduled for production to meet future, possibly flexible, target dates. In such settings, scheduling jobs early results in inventory costs whereas scheduling them late may result in penalty costs. Our models apply equally well to such settings.

The paper is structured as follows. Section 2 reviews relevant literature and places our contribution within it. Section 3 provides an MDP formulation of the chemotherapy appointment scheduling problem. We provide the details of our ADP solution approach and heuristics in Section 4. Section 5 describes our experimental set up and Section 6 provides numerical results and a discussion of their significance. We provide concluding remarks in Section 7.

2 Literature review

There is a rich literature on patient scheduling [2, 6, 10, 17]. This research can be categorized as either *allocation scheduling* or *advanced scheduling*. In allocation scheduling, arriving patient requests are either served or rejected immediately, whereas in advanced scheduling, they are scheduled to future service slots/days over a booking horizon [9]. While many papers study allocation scheduling [8, 10, 14, 19], research focusing on advanced patient scheduling is limited. This is mainly due to the fact that advanced scheduling requires more computational effort as compared to the allocation scheduling due to the need to take into account capacity restrictions for slots in a booking horizon as opposed to focusing only on the current slot. Cayirli and Veral [4] and Mondschein and Weintraub [20] survey the allocation scheduling literature. We review the advanced scheduling literature below.

Gerchak et al. [7] study the scheduling of elective surgery while accounting for random requests for bookings of emergency surgery. They develop an infinite-horizon MDP model and investigate the form and properties of the optimal policy under uncertain surgery durations. Klassen and Rohleder [12] study outpatient appointment scheduling in a dynamic multi-period environment where patients can be scheduled days or weeks in advance. One of the objectives of this work is to determine the best placement of appointment slots that should be left for urgent patients. The authors study the effect of several factors on scheduling performance such as the system load and the variability of service times. Lamiri et al. [15] address an elective surgery planning problem for operating rooms (ORs) considering elective patients and emergency patients. The authors determine the set of elective patients to be operated on in each OR in each period over a planning horizon using a stochastic mathematical programming model.

In the work most closely related to ours, Patrick et al. [21] study an advanced patient scheduling problem motivated by diagnostic imaging scheduling. They assume that arriving patients, who require a single slot, can be scheduled in available slots, diverted or delayed. They model the scheduling problem as an infinite-horizon discounted MDP, and approximately solve it using LP-based ADP. The authors also provide structural results for their problem, giving the form of the optimal linear value function approximation and the resulting policy. Their model differs from ours in the following ways:

- Patrick et al. [21] do not explicitly consider the notion of targets and tolerance limits. Patient classes correspond to appointment urgency. The system aims to schedule patients in a particular urgency class prior to a specific target date.
- In this paper, time windows are two-sided and patient class corresponds to tolerance limits.
- In this paper, patients within each class can have different target dates with different demand distributions. As Fig. 1 shows, these target dates vary considerably for a specific tolerance limit.
- In Patrick et al. [21], the system is only penalized for lateness. In this paper, the system is penalized when appointments are either early or late.
- By varying the relative magnitude of penalty costs to diversion costs, this paper allows tolerance limits to be relaxed. This is relevant to manufacturing settings where time windows are more flexible.

Liu et al. [18] study the dynamic scheduling of outpatients where patients can cancel or not show up for their appointments. They formulate the underlying problem as an MDP. Since the exact solution of their MDP model is computationally intractable, they propose heuristic policies.

More recently, Patrick [22] studies scheduling outpatients when no-show rates depend on wait times. He formulates the problem as an MDP that can be solved optimally, and shows that a scheduling policy that uses a short booking window to mitigate the effect of no-shows outperforms an open access scheduling policy.

In another methodologically related paper, Saure et al. [27] extend Patrick et al. [21] to a dynamic multi-appointment patient scheduling problem arising in radiotherapy. In their setting each appointment request triggers a sequence of appointment requests on consecutive days. Thus when choosing an appointment date, the scheduler must put aside resources for a sequence of consecutive days that varies with treatment type. The authors formulate their problem as an infinite-horizon discounted MDP, which is intractable for realistic problem sizes. They employ an LP-based ADP for approximately solving their MDP model, and perform simulations to test its performance against benchmark policies and practice.

The patient scheduling literature on chemotherapy is sparse. The most relevant paper is that of Turkcan et al. [28]. Differing from most of the previous work [5, 11, 16] the authors develop optimization methods to schedule chemotherapy treatments with the objective of decreasing patient waiting time and maximizing adherence to treatment plans. In their model, limited availability of clinic resources such as nurses, beds, and chairs are considered.

Our work also differs from Turkcan et al. [28] in a variety of aspects. First, they address a static and deterministic problem, thereby ignoring random and dynamic arrivals of patients. Hence, they use integer programming methods for solving the chemotherapy planning problem. Second, in their model, patients cannot be diverted, served through overtime, or delayed. Finally, as stated earlier, their model does not include the concepts of target dates and tolerance limits for patients.

Our paper is the first to consider patient scheduling in chemotherapy where patients of different types dynamically arrive over time, and have specific target dates along with tolerances. In this respect, our work can be viewed as an extension of Patrick et al. [21], who provide analytical results for their problem owing to the specific structures of the components of their MDP model along with certain cost parameters. However, the concepts of target dates and tolerances make our problem more challenging, and therefore deriving structural results for our problem appears to be impossible. In addition, we employ an ADP technique for approximately solving this new class of scheduling problems and investigate the policy obtained by ADP in order to provide insights on scheduling in chemotherapy. We also test the performance of a number of easy-to-use heuristic decision rules with the goal of proposing implementable scheduling rules.

3 Problem description and MDP formulation

We consider the following patient scheduling problem.

- There is an infinite time horizon but a finite rolling booking horizon.
- Patients are classified into types based on their appointment tolerances.
- Patients of each type with specific target dates arrive randomly each day. We assume that arrival distributions are stationary, and that arrivals across patient types as well as target dates are independent.
- Each appointment requires one appointment slot. Note that appointment durations vary considerably between patients but the BCCA Phase 1 booking process which we are modelling ignores these distinctions. These differences are accounted for in the Phase 2 optimization [26].
- At the end of each day, arriving patients are scheduled to future days over a booking horizon or diverted. (Diversion may correspond to overtime in the chemotherapy setting or outsourcing in the manufacturing setting.). We assume that diversion/overtime capacity is significantly higher than the maximum number of patients that need to be diverted on any day.
- Scheduling patients to days within their tolerance limits does not result in any cost, whereas a type-dependent scheduling cost per day is incurred when a patient is scheduled to a day outside the tolerance limit. These costs are introduced as in Patrick et al. [21] and Saure et al. [27] to facilitate an MDP formulation.
- Diversion costs are incurred for patients that are diverted or served through overtime, and are the same for patients of each type.
- The objective is to schedule arriving patients to available days (or divert them) in a booking horizon so as to minimize total discounted expected cost. Other performance metrics are investigated through simulation.

Scheduling costs for patients of each type are assumed to be linear in the number of days outside their time window. These costs are artificial and included to penalize the system for not respecting time windows. When they are high, patients must be scheduled in their time windows; when they are low, time windows become more flexible. Note however that the assumption of linear cost is ubiquitous in the related literature [10, 13, 23]. Further, we assume that higher scheduling costs are incurred for patients with lower tolerances. Note that diversion costs can either be real or included to penalize the system for not scheduling a patient.

Finally, although the objective of our MDP model considers only the cost of scheduling and diversion, we also take into account average resource utilization and average number of patients diverted in our comparative analysis owing to the fact that they are important practical measures for evaluating the quality of a scheduling policy.

3.1 The MDP model

3.1.1 Decision epochs

Decisions are made at the end of each day over an infinite horizon. The booking horizon is a rolling period of N days, but the planning horizon is infinite. We ignore the calendar effects of weekends and holidays in our formulation.

3.1.2 State space

The state space assumes the following form:

$$s = (x_1, \ldots, x_N; y_{11}, y_{12}, \ldots, y_{IN}), \tag{1}$$

where I is the number of patient types, x_n, for $n = 1, \ldots, N$, is the number of patients scheduled on day n, and y_{ik} for $i = 1, \ldots, I$ and $k = 1, \ldots, N$ is the number of type-i patients with target date k waiting to be scheduled. The time windows are implicit in the patient type.

The second component of the state space in Patrick et al. [21] has the form $(y_1, \ldots, y_I)$, corresponding to the number of type-i patients to be booked. Note that there is no explicit target date in that formulation. Instead, there is a latest date at which an appointment for class-i patients should be booked.

3.1.3 Action space

The set of available actions in state $s = \{x; y\}$ is denoted by $A_{\{x;y\}}$. A typical action in $A_{\{x;y\}}$ is given by:

$$a = (a_{ikn}, z_{ik}), \tag{2}$$

where a_{ikn} is the number of type-i patients with target date k to book on day n, z_{ik} is the number of diverted type-i patients with target date k. The following constraints must be satisfied for a given action:

$$x_n + \sum_{i=1}^{I}\sum_{k=1}^{N} a_{ikn} \leq C_1 \text{ for } n = 1, \ldots, N, \tag{3}$$

$$\sum_{i=1}^{I}\sum_{k=1}^{N} z_{ik} \leq C_2, \tag{4}$$

$$\sum_{k=1}^{N}\sum_{n=1}^{N} a_{ikn} + \sum_{k=1}^{N} z_{ik} = \sum_{k=1}^{N} y_{ik} \text{ for } i = 1, \ldots, I, \tag{5}$$

where C_1 is daily resource capacity and C_2 is maximum number of patients diverted or served by overtime each day.

Constraint (3) limits the total number of patients scheduled on each day to daily capacity. Constraint (4) insures that the total number of patients diverted at each period does not exceed the diversion capacity, whereas Constraint (5) requires that a decision be made for arriving patients waiting to be scheduled. Note that in our computation, we assume C_2 is sufficiently large so that Eq. 4 is always satisfied and need not be enforced explicitly.

3.1.4 Transition probabilities

Prior to each decision, the state changes according to

$$(x_1, \ldots, x_N; y) \longrightarrow \left(x_2 + \sum_{i=1}^{I}\sum_{k=1}^{N} a_{ik2}, \ldots, 0; y'\right). \tag{6}$$

Here, y'_{ik} is the number of type-i patients with target date k who arrived during the current day. This transition occurs with probability $p(y') = \prod_{i=1}^{I}\prod_{k=1}^{N} p(y'_{ik})$, where $p(y'_{ik})$ is the probability that y'_{ik} type-i patients with target date k arrive on a given day. Note that we need not assume that the components of y' are independent or that the demand in successive periods are uncorrelated. The validity of the independence assumption in practice can be checked using appropriate statistical methods and its impact on results can be observed through simulation.

3.1.5 Costs

We assume the following cost structure. The cost of scheduling a type-i patient with target date k to day n is denoted $b(i, k, n)$, and is given by:

$$b(i,k,n) = \begin{cases} 0, & \text{if } L_i \leq n \leq U_i \\ (L_i - n)c_1^i, & \text{if } L_i > n \\ (n - U_i)c_2^i, & \text{if } n > U_i, \end{cases}$$

where L_i for $i = 1, \ldots, I$ and U_i for $i = 1, \ldots, I$ are lower and upper tolerance limits for type-i patients, respectively; c_1^i for $i = 1, \ldots, I$ and c_2^i for $i = 1, \ldots, I$ are unit early and unit late costs for type-i patients, respectively. Note that the above cost function represents one of the main differences between our MDP model and those in Patrick et al. [21] and Saure et al. [27]. In these references, the only costs are for scheduling patients beyond their target dates while this formulation assumes that the cost of scheduling a patient within his/her tolerance limits is 0, whereas a linear cost is incurred for scheduling a patient either early or late. The immediate cost then becomes:

$$c(\mathbf{a}, \mathbf{z}) = \sum_{i,k,n} b(i,k,n)a_{ikn} + \sum_{i=1}^{I} d(i) \sum_{k=1}^{N} z_{ik}, \tag{7}$$

where $d(i)$ for $i = 1, \ldots, I$ is a per unit penalty cost for diverting a type-i patient.

3.1.6 Bellman's equations

We assume discounting with discount factor λ. The goal is to find a policy that minimizes the expected infinite horizon discounted cost. Bellman's equations are given by

$$v(\mathbf{x}, \mathbf{y}) = \min_{\{\mathbf{a}, \mathbf{z}\} \in A_{\{x,y\}}} \left\{ c(\mathbf{a}, \mathbf{z}) + \lambda \sum_{\mathbf{y}' \in D} (\mathbf{y}') \right. \left. v(x_2 + \sum_{i,k} a_{ik2}, \ldots, 0; \mathbf{y}') \right\}, \quad (8)$$

where D is the set of all possible demand vectors.

The state space and action space of our MDP model grow exponentially with the number of patient types and the length of the booking horizon. This makes realistic sizes of the appointment scheduling problem computationally intractable, and hence we resort to ADP for solving our model. A brief discussion about our use of ADP and heuristic policies follow.

4 Booking policies

In this section, we describe several promising and easy to use booking policies that we will evaluate through simulation. The first is obtained using ADP while the remainder are easy-to-implement practical heuristics. We consider heuristics for three reasons:

- They are intuitive and easy to implement.
- When the system is not overloaded (i.e., when the average demand is less than capacity), the ADP value function approximation (which is explained in detail in Section 4.1.1) may have all components equal to 0. Consequently, any myopic policy (that chooses the optimal action for a given state by minimizing immediate costs) is optimal with respect to the ADP approximation. Since early and late penalty costs are zero within the tolerance limits, in these cases the optimal action for a given state depends only on immediate costs, so that these policies are not well specified. Several of these heuristics correspond to specific implementations of the ADP policy in this case.
- In the absence of quality lower bounds for the ADP solution, they provide comparisons for assessing its quality.

4.1 Approximate dynamic programming

Approximate Dynamic Programming has been used to address the computational intractability of most realistic MDPs. It has been successfully employed for solving diverse problems including dynamic routing and scheduling, machine scheduling, energy management, and health resource management [23]. Examples of ADP techniques include Q-learning, Temporal Difference Learning, and Neuro Dynamic Programming [24]. ADPs may be classified on the basis of whether the approximate value function of the underlying MDP is estimated by linear programming (LP) or simulation. Adelman and Klabjan [1], Patrick et al. [21], Saure et al. [27] and Gocgun and Ghate [9] use the LP approach while Powell [23] and Saure et al.[1] consider the simulation approach.

To use any ADP approach requires expressing the value function in terms of a set of basis functions that capture relevant features of the state space. The basis functions and the value function are related through a set of tuning weights or parameters. Simulation-based ADPs update these parameters iteratively using regression-based methods [3, 24] or direct search,[2] whereas in the LP approach, these parameters are obtained from the solution of the approximate LP. The ADP policy is then obtained by substituting the approximate value function into the right hand side of the Bellman's equations (8) and solving the optimization problem as necessary. In rare cases, the form of the corresponding policy can be determined explicitly [2]. Often, the solution of the LP equals 0, resulting in a myopic policy that is not well-defined [2, 19].

4.1.1 The LP-based ADP

We now describe our LP approximation. It is similar to that in Patrick et al. [21] and Saure et al. [27] but is included for completeness. For a discounted infinite-horizon MDP (where the objective function is in minimization form as in (8) and $\alpha(\mathbf{s})$ are positive numbers indexed by states $\mathbf{s} \in S$), the equivalent LP formulation is given by Puterman [25]:

$$\begin{aligned} &\max \sum_{\mathbf{s} \in S} \alpha(\mathbf{s}) v(\mathbf{s}) \\ &s.t.\ c(\mathbf{s}, \mathbf{a}) + \lambda \sum_{\mathbf{s}' \in S} p(s' | \mathbf{s}, \mathbf{a}) v(\mathbf{s}') \geq v(\mathbf{s}) \ \ \forall s \in S, \mathbf{a} \in A_{\mathbf{s}}. \end{aligned} \quad (9)$$

If the components of α sum up to 1, which we now assume, they may be regarded as an exogenous probability distribution over the initial states of the system. When (9) is solved exactly, the resulting policy obtained through the dual LP is independent of α, but when it is solved using ADPs, the policy depends on this quantity.

[1] Saure A, Patrick J, Puterman ML (2013) Simulation-based ADP with generalized logistic functions. under review.

[2] Maxwell MS, Henderson SG, Topaloglu, H. Tuning Approximate Dynamic Programming Policies for Ambulance Redeployment via Direct Search, (submitted).

We approximate the value function by choosing the state components as basis functions and using the following affine approximation:

$$\tilde{v}(\mathbf{x}, \mathbf{y}) = W_0 + \sum_{n=1}^{N} V_n x_n + \sum_{i=1}^{I} \sum_{k=1}^{N} W_{ik} y_{ik}. \tag{10}$$

The LP formulation of our MDP model is given by:

$$\begin{aligned}
&\max_{v} \sum_{(\mathbf{x},\mathbf{y})\in S} \alpha(\mathbf{x}, \mathbf{y}) v(\mathbf{x}, \mathbf{y}) \\
&s.t. c(\mathbf{a}, \mathbf{z}) + \\
&\lambda \sum_{d\in D} p(d) v \left(x_2 + \sum_{i,k} a_{ik2}, \ldots, x_N + \sum_{i,k} a_{ikN}, 0; y'_{in} \right) \\
&\geq v(\mathbf{x}, \mathbf{y}), \quad \forall (\mathbf{x}, \mathbf{y}) \in S, \ \forall (\mathbf{a}, \mathbf{z}) \in \mathbf{A}_{(x,y)}.
\end{aligned} \tag{11}$$

Following the approach in Patrick et al. [21], we substitute (10) into (9) and rearrange terms, to obtain the following LP:

$$\begin{aligned}
&\max_{\mathbf{V},\mathbf{W}} \ W_0 + \sum_{n=1}^{N} E_\alpha(X_n) V_n + \sum_{i=1}^{I} \sum_{k=1}^{N} E_\alpha(Y_{ik}) W_{ik} \qquad (12) \\
&s.t. \ (1-\lambda) W_0 + \sum_{n=1}^{N} V_n \left(x_n - \lambda x_{n+1} - \lambda \sum_{i=1}^{I} \sum_{k=1}^{N} a_{ik(n+1)} \right) + \\
&\sum_{i=1}^{I} \sum_{k=1}^{N} W_{ik} (y_{ik} - \lambda E_\alpha(Y_{ik})) \leq c(\mathbf{a}, \mathbf{z}), \ \forall (\mathbf{x}, \mathbf{y}) \in S, \\
&\forall (\mathbf{a}, \mathbf{z}) \in A_{(\mathbf{x},\mathbf{y})} \\
&V_n \geq 0, \ n = 1, \ldots, N \\
&W_{ik} \geq 0, \ i = 1, \ldots, I \text{ and } k = 1, \ldots, N.
\end{aligned}$$

In (12), the variables X_n and Y_{ik} are random variables with respect to α, and represent the number of patients scheduled on day n and the number of type-i patients with target date k waiting to be scheduled. Since the LP in (12) still has a very large number of constraints, we seek a solution to the dual LP using column generation. Our approach, which follows that in Patrick et al. [21], is described in Appendix A.

We note that our exploratory analysis of the BCCA patient data revealed that it may be correlated across patient type. However, this would not affect the implementation of ADP for this class of problems since the only change in the model would be in the expression for the transition probability. Namely, the state transition in that case would occur with probability $p(\mathbf{y}') = \prod_{k=1}^{N} p\left(y'_{1k}, \ldots, y'_{Ik}\right)$. However, the reduced model (12) would remain the same since it only depends on the components of the demand distribution through their means.

4.2 Heuristic policies

The following heuristic policies describe a series of practical rules that make sense to schedulers and are easy to use. Each of the heuristics prioritizes patients based on the width of their tolerance intervals; patients with narrower tolerance limits are given higher priority because they are harder to schedule and are assigned first.

Target policy This policy attempts to schedule arriving patients of each type at their target date in the order of their priority. The policy ignores the tolerance limits and diverts patients when there is no capacity available on their target dates. Note that this policy is not a myopic policy for the MDP unless the diversion cost equals 0. Further, we include this policy because it corresponds to practice at the BCCA prior to the study reported in Santibanez et al. [26].

Tolerance policy This policy attempts to schedule arriving patients of each type to a day within their tolerance limits by first considering their target date and then other days, prioritizing days that are closer to the target date and choosing later days first. In case of insufficient capacity for days within the tolerance limits, the policy diverts patients.

Capacity policy This policy attempts to schedule arriving patients to days within their tolerance limits in order of available capacity. In case of ties, it schedules patients to the earliest available day. As in other heuristics, it resorts to diversion when patients cannot be scheduled to any day within their time window.

Earliest policy The Earliest policy schedules patients to the earliest available day within their respective time windows. The policy diverts patients when none of the days within the respective time windows are available for appointment.

Latest policy The Latest policy schedules patients to the latest available day within their respective tolerance limits. In case of insufficient capacity for days within the tolerance limits, the policy resorts to diversion.

For cases in which diversion costs are greater than early/late scheduling costs, it may be optimal to schedule patients (or jobs) outside of their respective tolerance limits instead of diverting them. In this case, we modify each of these heuristics except the target policy to seek the least costly option instead of automatically diverting patients when no capacity remains in his/her time window. Specifically, for each patient, each heuristic identifies days for which early scheduling and late scheduling cost is the smallest, compares unit scheduling cost for those days against the unit cost of diversion, and chooses the least costly option.

5 Numerical experiments

In this section, we describe the experimental setup we use to compare the ADP policy to the above heuristics.

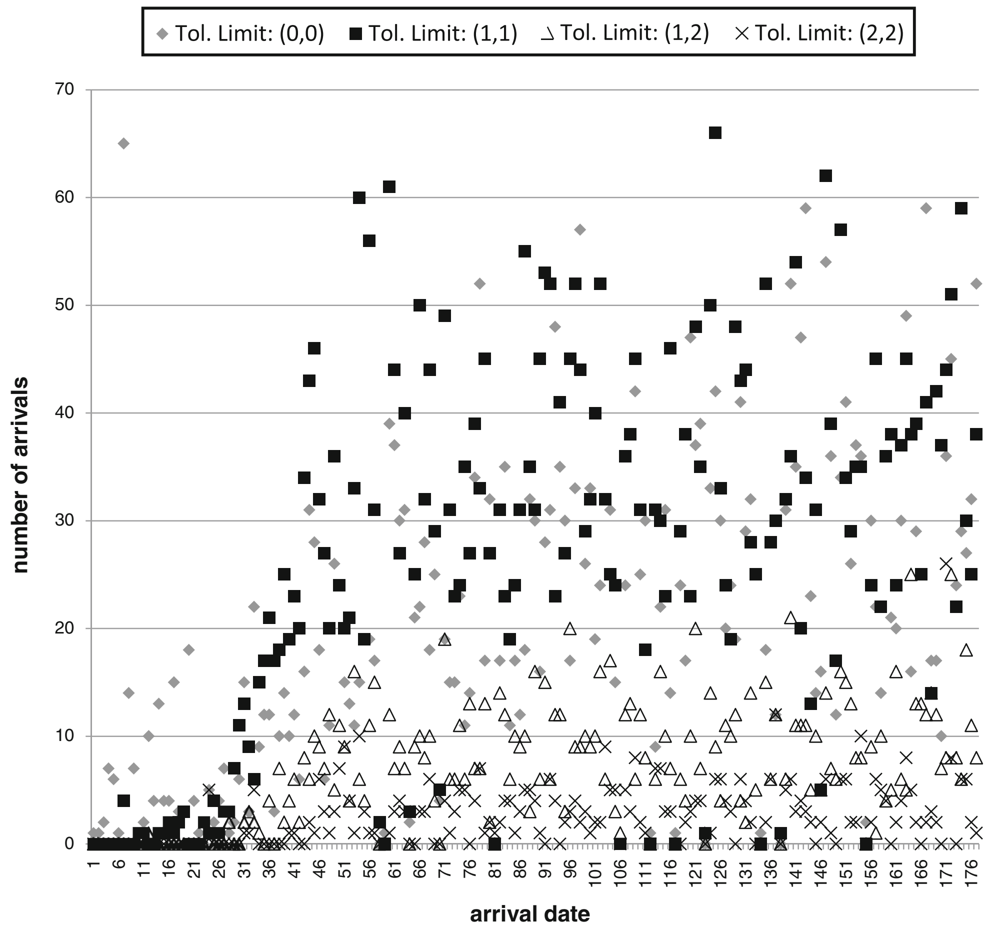

Fig. 2 Time series of patient demand by each patient type (tolerance limits) based on BCCA data between (May 2010 and January 2011). Note that weekends are excluded

Table 1 summarizes its key features. It shows that we consider two arrival patterns and a range of cost parameters. The key differences are the number of patient classes (three or four), the demand distributions (constant and empirical), and the relationship between the diversion costs and the scheduling costs (low and high diversion costs). We vary the cost relationships so that our experiments correspond to both the chemotherapy and manufacturing applications.

5.1 Demand data

To specify the patient appointment demand data requires setting the number of patient classes, each corresponding to a specific tolerance limits, and the distribution of daily demand by class over the booking horizon which we assume to be 30 days.

- **The Constant Demand Case:** We assume three patient classes with tolerance limits (0,0), (1,1), and (2,2). The arrival distribution is assumed to be independent Poisson with mean 1 for each class for each day over the booking horizon with the exception that demand for days for which the corresponding tolerance limits lie beyond the booking horizon are set to 0. That means that without loss of generality, on any given day, we sample independent realizations of y_{ik} for $i = 1, 2, 3$ and $k = 1, \ldots, 30$ from a Poisson (1) distribution. Under this demand distribution, we assume capacity levels of 80, 85, and 90. Since the total mean daily demand nearly equals 85, we refer to these scenarios as high, medium and moderate congestion. (The total mean daily demand for type-1 patients is 30 whereas those for type-2 and type-3 patients are set to 28 and 26, respectively, since we assumed that the tolerance limits for a given target date cannot lie beyond the booking horizon. More specifically, mean arrival rates for type-2 patients with target dates 1 and 30 and those for type-3 patients with target dates 1, 2, 29, and 30 are set to 0.
- **The Empirical Demand Case:** This pattern is representative of the BCCA follow-up data displayed in Figs. 1 and 2. Note that Fig. 1 illustrates the clumpy nature of demand over target date, whereas Fig. 2 demonstrates that the total daily demand for each patient type is highly variable. We smooth the data so that demand is concentrated on the most frequent days which are 7, 14, 21 or 28 days in the future

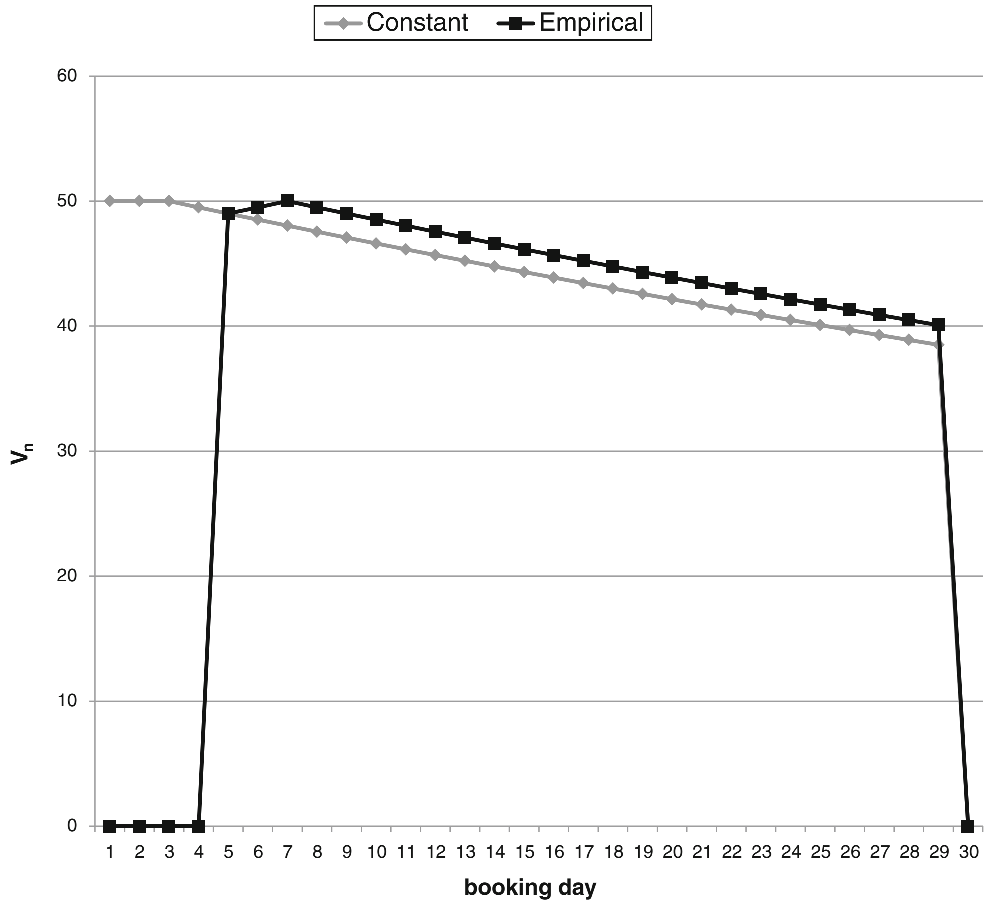

Fig. 3 Plots of V_n versus booking day for the Rigid Tolerance Case

and truncate it at 30 days which is the length of the booking horizon. This periodic nature of appointments reflects respective chemotherapy treatment protocols. We assume demand is Poisson with means derived from the data; note that we aggregated over the tolerance limits when computing means. For reasons noted in the introduction, we do not consider new patient demand. In line with the BCCA data, we consider 4 patient types with tolerance limits (0,0), (1,1),(1,2), and (2,2). In this setting, we choose capacity levels equal to 55,60 and 65 corresponding to high, medium and moderate congestion.

In each case the Poisson distribution is truncated at ten times its mean value. We set the diversion capacity equal to 500 to insure that each patient is either scheduled or diverted.

5.2 Costs

We need to specify values for the parameters c_1^i and c_2^i in the scheduling cost $b(i, k, n)$ and the diversion cost $d(i)$. The relative values of these quantities impact the trade-off of scheduling outside the tolerance limits and diversion. In the BCCA-motivated application, scheduling outside the tolerance limits is not clinically advisable so that c_1^i and c_2^i will be set relatively high. In a manufacturing setting in which it may be permissible to schedule jobs outside the tolerance limits, the diversion costs may be set high relative to c_1^i and c_2^i.

We assume that the narrower the tolerance limit, the higher the scheduling cost. This is so if there is available capacity on a given day, preference be given to those appointments with less flexibility. In the constant demand case with three patient classes, we set c_1^i, the unit early scheduling costs per day, to 100, 75, and 50, whereas c_2^i, the unit late scheduling costs per day, is set to 125, 100, and 75. In the empirical demand case with four patient classes, c_1^i is set to 100, 75, 75, and 50, respectively, whereas c_2^i is set to 125, 100, 75, and 75, respectively. Note that the earlier costs are lower than the later costs so that when a patient is scheduled outside the tolerance limit, preference is given towards scheduling early.

Diversion cost per patient for each patient type is set to 50, 75, 100 and 125. Note that 50 corresponds to the lowest scheduling cost per day. We refer to the case when the diversion cost equals 50 as the “rigid tolerance case”,

Table 2 Results for the rigid tolerance case with constant arrival rates

C	V_n	ADP	Target	Tolerance	Capacity	Earliest	Latest
80	>0	0.99 – 4.73– 23822	0.98 – 5.95– 29874	0.99 – 4.73– 23738	**1.00–4.27**– **21545**	0.99 – 4.39– **22048**	0.99 – 4.97– 25018
85	0	–	0.96 – 3.17– 15887	0.96 – 2.01– 10125	**0.98–1.08**– **5487**	0.97 – 1.55– 7749	0.96 – 2.29– 11581
90	0	–	0.92 – 1.44– 7196	0.92 – 0.78– 3956	**0.93–0.11** **586**	**0.93**–0.51– 2529	0.92 – 0.95– 4757

The numbers that are bolded indicate that the performance of the underlying policy with respect to the respective metric is statistically better than that of other policies for which the respective numbers are not bolded. C denotes system capacity

corresponding to the BCCA appointment booking setting. When there is insufficient capacity within the time window, patients will in practice either be served through overtime or wait-listed in hope of a cancellation. The diversion cost penalizes the system for doing so. In practice cancellations are usually for medical reasons and occur one day before the appointment. Historically the cancellation rate is about 5 %. We refer to the other choices for the diversion costs as "relaxed tolerance cases" since the tolerance limits need not be rigidly enforced.

In order to examine the impact of different slopes of the cost function on the performance of the system, we consider the cases where early (late) scheduling costs are much higher than late (early) scheduling cost for the relaxed tolerance case (the results for the "rigid tolerance case" would not be affected by such a change since early/late scheduling is not permissible). In particular, the unit early scheduling cost per day for each patient type is set to 300, 225, and 150, respectively for the case of high early scheduling cost, whereas the levels for the late scheduling cost per day for each patient type are 250, 200, and 150, respectively.

5.3 Simulation details

Simulation run lengths and the warm-up period were set to 1,500 days and 500 days, respectively. The number of runs for instances of the rigid tolerance case is set to 50, whereas the number of runs for instances of the relaxed tolerance case was 10. This is because the former case required more accurate estimates than the latter case when comparing policies.

Finally, the discount factor is set to 0.99, and total cost is discounted beginning after the warm-up period.

The column generation approach and the decision retrieval process of ADP were implemented using GAMS 23.5 with CPLEX 12.2 as the solver on a computer running Windows XP with 3 GHz of processing speed and 16 GB of RAM. The simulations were coded in GAMS and required about 10 hours to evaluate 50 replicates for the ADP policy, and nearly 40 min for each of the heuristic policies.

6 Results and discussion

6.1 ADP solution properties

We begin by discussing some features of the ADP solution and their effect on the resulting policy. As in Patrick et al. [21] and Saure et al. [27], we observed that all of the V_n values equalled zero when the system was not over-capacitated. This corresponds to capacities of 85 and 90 in the constant demand case and 60 and 65 in the empirical demand

Table 3 Results for the rigid tolerance case with empirical arrival rates

C	V_n	ADP	Target	Tolerance	Capacity	Earliest	Latest
55	0	1.0 – 5.09– 25130	0.98 – 6.19– 30661	1.0 – 5.17– 25508	1 – 5.20– 25655	1.0 – 5.08– 25005	0.99 – 5.43– 26876
60	0	–	0.95 – 3.09– 15290	0.97 – 1.58– 7643	0.97 – 1.65– 8043	**0.98–1.23**– **5850**	0.96 – 2.25– 11161
65	0	–	0.90 – 1.24– 6167	0.92 – 0.28– 1327	**0.92**–0.29– 1359	**0.92–0.16**– **745**	0.91 – 0.78– 3877

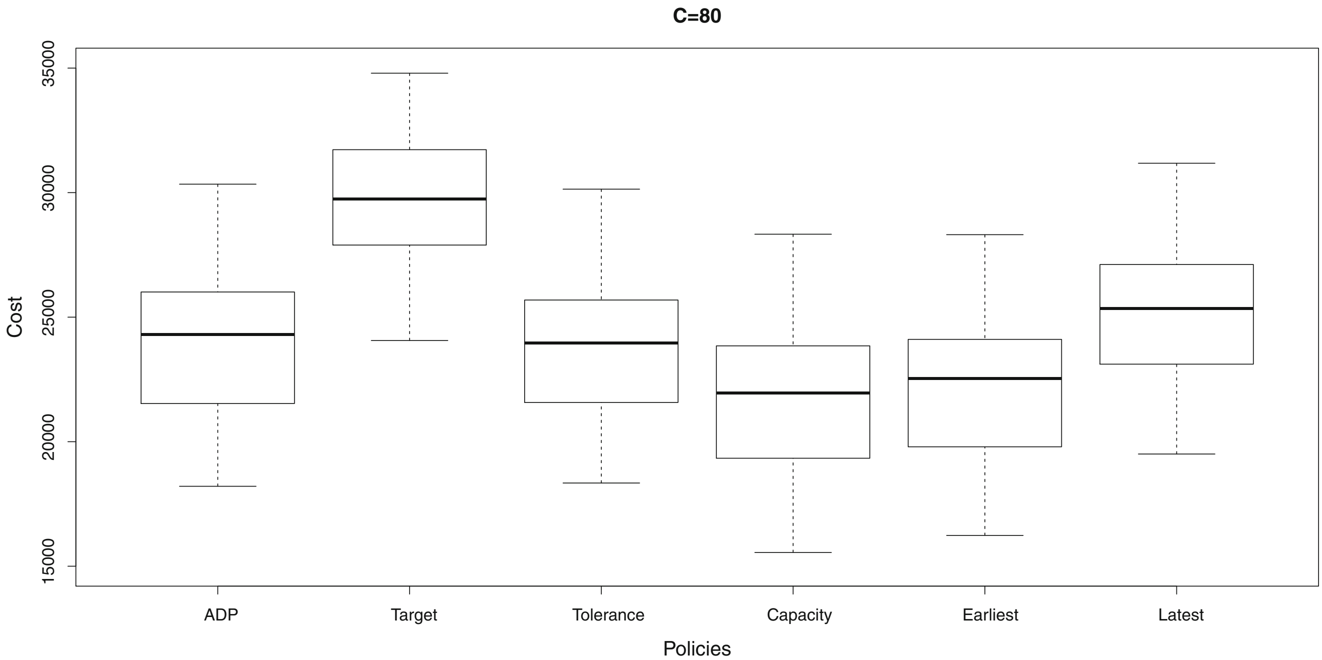

Fig. 4 Boxplots of costs obtained by the policies for the rigid tolerance case with constant arrival rates (number of replications is set to 50)

case. The values were non-zero in the high congestion cases corresponding to a capacity of 80 in the constant demand case and 55 in the empirical demand case. This is a shortcoming of the affine approximation. Recent work by Saure et al. [27] uses a non-linear value function approximation to address this shortcoming.

As noted above, to obtain a policy from the ADP solution, we substitute V_n into the right hand side of Equation (8) and perform the following minimization for any system state that arises in practice or in the simulation:

$$\min_{(\boldsymbol{a},\boldsymbol{z})\in A_{(x,y)}} \sum_{i,k,n} b(i,k,n)a_{ikn} + \sum_{i=1}^{I} d(i) \sum_{k=1}^{N} z_{ik} + \lambda \left(V_1 \left(x_2 + \sum_{i,k} a_{ik2} \right) + \ldots + V_{N-1} \left(x_N + \sum_{i,k} a_{ikN} \right) \right), \quad (13)$$

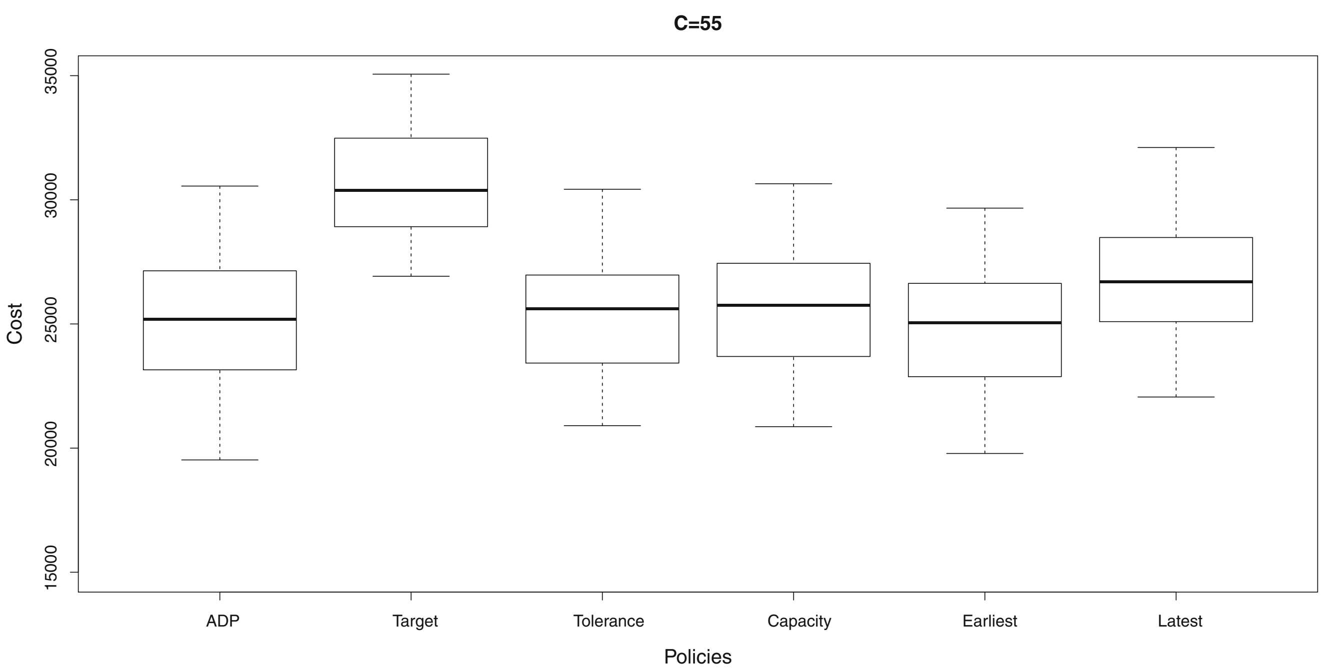

Fig. 5 Boxplots of the costs obtained by the policies for the rigid tolerance case with empirical arrival rate case over 50 replications

Table 4 Results for the relaxed tolerance case with constant arrival rates. D.C. denotes diversion cost

C	D.C.	V_n	ADP	Tolerance	Capacity	Earliest	Latest
	75		0.99 – 4.70–34400	0.99 – 4.78–34010	**1.0–4.34–30630**	**1.0–4.46–**31340	0.99 – 4.92–37760
	100		0.99 – 4.64–45920	0.99 – 4.78–45370	**1.0–4.34–**40830	**1.0**–4.46–41800	0.99 – 4.56–59980
80	125	>0	0.99 – 4.5–57610	0.99 – 4.78–56690	**1.0–4.34–51050**	1.0 – 4.46 52240	1.0 – 4.35–82800
	250		1.0 – 4.12–**134200**	1.0 – 4.15–185700	1.0 – 4.13–185700	1.0 – 4.11–181300	1.0 – 4.18–188000
	75		–	0.97 – 2.05–14150	**0.98–1.1–7125**	0.97 – 1.58–10804	0.96 – 2.25–16240
	100		–	0.97 – 2.05–18880	**0.98–1.1–9490**	0.97 – 1.58–14422	0.97 – 1.72–24480
85	125	0	–	0.97 – 2.05–23590	**0.98–1.1–11867**	0.97 – 1.58–18020	0.98 – 1.24–32920
	250		–	0.98 – 0.74–52290	0.98–**0.62–37750**	0.98–**0.56–41570**	0.98 – 0.84–58720
	75		–	0.93 – 0.80–5501	0.93–**0.11–584**	0.93 – 0.53–3483	0.92 – 0.93–6296
	100		–	0.93 – 0.51–7549	0.93–**0.11–779**	0.93 – 0.53–4644	0.93 – 0.56–9010
90	125	0	–	0.93 – 0.18–9570	0.93–**0.11–973**	0.93 – 0.52–5805	0.9 – 0.21–11059
	250		–	0.93 – 0.06–11975	0.93–**0.02–1806**	0.93 – 0.03–7720	0.93 – 0.07–14190

Note that we removed the variables W_{ik}'s from the above objective function as they do not have any impact on the actions chosen at a given state. Note also that the column V_n in Tables 2 and 3 indicates whether all V_n terms are 0 or some (most) are positive. When all V_n terms equal 0, (13) shows that the optimal action depends only on the immediate costs $b(i,k,n)$ and $d(i)$. In this case, we say that the ADP policy is *myopic* or *greedy*. Since the scheduling costs are zero within the tolerance limits, the ADP policy is not uniquely specified for patients who have flexibility in their appointment dates. The Tolerance, Capacity, Earliest and Latest heuristics described above all provide exact policy specifications in this case.

When the V_n are positive, the ADP policy trades off the short-term and long-term impact of the patient appointment day choice through the appropriate terms in (13). Figure 3 shows how V_n varies with n in the rigid tolerance case for the two of the demand patterns. Observe that in the constant demand case, V_n is constant to day 3 then decreasing. This suggests that, when there is enough capacity in the respective time-window, costs will be minimized by scheduling as late as possible within the tolerance limits for demand more than 3 days in the future and to any date in the first three if the time window falls within the the first three days. Hence the ADP should perform roughly similarly to the "Latest" heuristic policy. As for the empirical demand case, we see that V_n is 0 for days 1–4, increasing for days 5–7 then decreasing. Since in this data, the earliest appointment request is for 7 days in the future and the maximum tolerance is 2, it suggests that, when scheduling the within time-window is feasible, appointment requests for 7 days in the future should be scheduled as early as possible and all others should be scheduled as late as possible. This scheduling rule does not correspond to any of the heuristics but suggests a very plausible and implementable policy.

Note that in the relaxed tolerance cases, the V_n plots have the same form. Hence, we would expect the ADP policy to divert less frequently as compared to the rigid tolerance case and schedule patients or jobs outside of their tolerance limits.

6.2 The rigid tolerance case

We begin with the results for a scenario in which the mean arrival rate for patients of each type equals 1 for each day in the booking horizon and the diversion costs are high. We evaluated three instances corresponding to different daily capacity levels. The results are reported in Table 2 (see also

Table 5 Results for the relaxed tolerance case with empirical arrival rates

C	D.C.	V_n	ADP	Tolerance	Capacity	Earliest	Latest
	75		1.00 – 5.07–39060	1.00 – 5.10–41880	1.00 – 5.12–41640	1.00 – 5.04–41170	1.00 – 5.26–43130
	100		1.00 – 5.05–**51870**	1.00 – 5.04–72690	1.00 – 5.03–80500	1.00 – 5.01–72980	1.00 – 5.07–78860
55	125	>0	1.00 – 5.06–**65010**	1.00 – 5.01–97990	1.00 – 5.01–106800	1.00 – 5.00 97670	1.00 – 5.03–105780
	250		1.00 – 5.00–**129000**	1.00 – 5.00–207500	1.00 – 5.00–218500	1.00 – 5.00–208300	1.00 – 5.00–216700
	75		–	0.98 – 1.49–12262	0.97 – 1.56–12670	0.98–**1.16**–**9571**	0.97 – 2.06–17330
	100		–	0.98 – 1.28–21940	0.98 – 1.26–24610	0.98–**0.98**–18770	0.98 – 1.51–30670
60	125	0	–	0.99 – 0.92–39700	0.98 – 0.93–45460	0.99–**0.77**–33530	0.98 – 1.07–47440
	250		–	0.99-0.6-75260	0.99-**0.69**-96230	0.99-**0.60**-75260	0.99-0.74-97320
	75		–	0.92 – 0.26–2179	0.92 – 0.26–1908	0.92 – 0.15–**1204**	0.91 – 0.71–5776
	100		–	0.92 – 0.17–3039	0.92 – 0.12–2793	0.92–**0.05**–**1675**	0.92 – 0.32–8069
65	125	0	–	0.92 – 0.02–4542	0.92 – 0.02–3790	0.92–**0.008**–**2070**	0.9 – 0.059–10568
	250		–	0.92 – 0.01–4991	0.92 – 0.01–4245	0.92–**0.001**–**2118**	0.9 – 0.009–11666

Fig. 4 for boxplots of the cost values for the case when V_n is not identically zero). For each policy, the table gives average capacity utilization, average number of diverted patients and average discounted cost over the 50 replicates, respectively. The numbers that are bolded in each table indicate that the performance of the underlying heuristic policy with respect to the respective metric is statistically better than that of other policies on the basis of a paired-t test at the .05 level. Observe that for each policy, performance improves as capacity increases. Further and more importantly, when the capacity is 80, in which case the ADP solution is not degenerate, the Capacity heuristic performs best with respect to all three metrics. Since the ADP policy is not optimal for the original MDP, we can view the capacity policy as a reference for assessing the quality of the ADP solution. In this case, the average cost obtained by the ADP policy is within 10 % of the capacity policy (see also Fig. 4 for the boxplots for these two policies). When the capacity is set at 85

Table 6 The best policy under the range of scenarios considered in this study

	Const. D.			Empir. D.		
	High Cong.	Medium Cong.	Moder. Cong.	High Cong.	Medium Cong.	Moder. Cong.
Rigid Toler.	Cap.	Cap.	Cap.	ADP/Earl./Cap.	Earl.	Earl.
Relaxed Toler.	Cap.	Cap.	Cap.	ADP	Earl.	Earl.

Table 7 Percentage difference values for the costs incurred through the ADP policy and the best heuristic policy for each scenario

Scenarios	D.C.	Perc. Diff.
Rigid toler. - constant arrival rates	50	9.6
Rigid toler. -empirical arrival rates	50	0.5
Relaxed toler. -constant arr. rates	75	11.0
	100	11.1
	125	11.4
	250	−35.1
Relaxed toler. -empirical arr. rates	75	−5.4
	100	−40.1
	125	−50.2
	250	−60.9

or 90, the Capacity policy remains the best of the heuristics. Observe also that Target policy (which ignores the tolerance limits) is the worst for all capacity levels.

Next we discuss results when the data are generated using empirical arrival rates. In the low capacity case, (i.e., when capacity equals 55) the Earliest heuristic and ADP policy perform similarly on all three metrics (see also Fig. 5 for the boxplots for these policies). This suggests that the ADP policy may be close to optimum here. In cases when there is adequate capacity (i.e., when the capacity equals 60 or 65), the Earliest heuristic performs best.

6.3 The relaxed tolerance case

Results for the relaxed tolerance case appear in Tables 4, 5, 8, and 9. Note that the tables for this case do not include the results for the Target policy because this policy would be very similar to the Tolerance policy in the relaxed tolerance case. The results for the constant arrival rate case reveal that the Capacity policy outperforms other policies in the high congestion case except for the case with high diversion cost, as well as in the medium and moderate congestion cases (see Table 4). Moreover, as in the rigid tolerance case, the Target policy and the Latest policy are outperformed by the other policies.

The results for the relaxed tolerance case with empirical arrival rates are quite different than those in the mean arrival rate case. To begin with, the ADP policy outperforms heuristic policies in the high congestion case (see Table 5). On the other hand, the Earliest policy performs better than the other policies in the medium and moderate congestion cases. In terms of the average number diverted, the Earliest policy performs better than other policies in the medium and moderate congestion case, as well.

Finally, we discuss the results for the cases in which early (late) scheduling costs are significantly higher than their base-case values. The results are reported in Tables 8 and 9 (see Appendix B). We begin with the results for the high early cost case. The ADP policy performs best with respect to cost in the high congestion case with high cost of diversion, whereas the Capacity policy and the Earliest policy perform better than the ADP policy in the same case with respect to average number diverted. In addition, the Capacity policy outperforms other policies with respect to average cost and average number diverted in the medium and moderate congestion cases. As for the high late cost case, the Capacity policy performs the best with respect to cost for the high congestion case with low and moderate values of diversion cost. The ADP policy, however, outperforms other policies in the same case with significantly high diversion cost. As in the high early cost case, the Capacity policy performs better than other policies when the system has medium or moderate congestion.

6.4 Discussion

Table 6 summarizes results reported in the last section by describing the best policy or policies for each scenario. Recall that the rigid tolerance case with empirical demand generated arrival rates and high congestion corresponds most closely to chemotherapy scheduling setting that motivated this research. As Table 3 shows for this case, the ADP policy is not degenerate and yields comparable performance on all three metrics to the Earliest heuristic. Further, in cases with less congestion, the ADP solution is degenerate and does not uniquely determine a policy. In these cases, the Earliest heuristic achieves the best results by, on average, diverting the fewest patients. Since in practice, diverting patients is troublesome, this suggests that this policy should be considered in practice. We were surprised to find that this policy gave the best results since, prior to carrying out this study, we expected that the Capacity policy would outperform the other heuristics. This result can be explained as follows: utilizing slots belonging to respective tolerance limits for each patient type as early as possible is crucial in the empirical case with moderate/medium congestion. This is due to the fact that, at a given day, demand for initial few slots in a booking horizon is 0 in the lumpy demand case, which implies that an efficient policy must utilize such slots efficiently by prioritizing earlier slots at each decision point. The Earliest heuristic does so, whereas the Capacity Policy overlooks the utilization of such slots each day, and thus performs less efficiently than the Earliest heuristic.

We believe that the relaxed tolerance cases with both smooth and lumpy demand best represent scheduling challenges that arise in manufacturing settings. As before, in these cases, the ADP policies were not degenerate in the high congestion cases but were degenerate when capacity exceeded the mean demand. Results were a bit mixed in this setting. For the congested case (capacity 80) with constant arrival rates, Table 4 shows that the Capacity heuristic is best with low diversion costs but the ADP policy is best when the diversion cost is highest (250). This is interesting because in this case, the decision maker has the most flexibility in scheduling jobs outside of the time windows since few jobs will be diverted.

Table 5 suggests that the same phenomenon occurs in the congested empirical demand case except that the ADP policy is best at all diversion cost levels. In the non-congested empirical demand cases, the Earliest heuristic (modified to allow scheduling outside of the time window) diverts the fewest jobs and in most cases produces the lowest average discounted cost.

Finally, Table 7 provides a comparison of the ADP policy and the best heuristic policy in terms of percentage difference in total cost. As shown in this table, the

ADP policy generally performs worse than the best heuristic in all scenarios except the relaxed tolerance case with empirical arrival rates. The performance of the ADP policy in this case is on average 40 % higher than the best heuristic.

7 Conclusion

Motivated by issues identified when reviewing chemotherapy appointment scheduling processes at the BCCA, we studied a class of appointment scheduling problems in which arriving patients must be scheduled to receive treatment on specific days within a booking horizon. Novel features of this problem are that patients of each type have specific target dates and tolerance limits and demand varies by patient type and target date. This paper shows that such problems can be formulated as ADPs and solutions can be obtained through linear programming. Since our formulation uses a linear approximation for the value function and no bounds are available for its quality, we compared the performance of its solution to several practical heuristics through simulation. Furthermore, our simulations addressed a setting in which time windows can be relaxed that may be applicable to manufacturing problems.

We draw the following conclusions from this study:

- Complex appointment scheduling problems can be formulated as MDPs and solved using ADPs. ADP solution quality is best in highly congested cases when decision rules involve complex trade-offs and impact system performance.
- ADP solutions based on an affine function approximation are degenerate when there is excess capacity. This suggests the need for better value function approximations in such cases.
- The Earliest heuristic works best in the BCCA chemotherapy appointment setting and is worthy of consideration for implementation.
- The ADP policy performs well for cases in which patients can be scheduled outside their respective tolerance limits; it generally outperforms the best heuristic policy for these cases. Hence, the LP-based ADP policy can also be successfully applied in a manufacturing setting.

There are several practical realities in chemotherapy appointment scheduling that are not addressed in our model; most notably choosing the optimal amount of capacity to reserve for new patients, different appointment durations, and cancellations. These issues need to be further studied before applying the results of this paper to that setting. A closely related issue is using the models herein to address when to add a block of surge capacity in the future so as to address peaks in demand proactively.

Acknowledgments This research was supported by Martin L. Puterman's NSERC Discovery Grant and the NSERC CREATE Program in Healthcare Operations and Information Management program. We wish to also thank Antoine Saure for providing us the ADP GAMS code he wrote for solving the kind of problem studied in this research.

Appendix A: Column generation approach

As discussed earlier, the column generation procedure solves the dual of the LP formulation given in (12):

$$\min_X \sum_{\{\mathbf{x},\mathbf{y}\}\in S,\{\mathbf{a},\mathbf{z}\}\in A_{\{\mathbf{x},\mathbf{y}\}}} X(\mathbf{x},\mathbf{y},\mathbf{a},\mathbf{z})c(\mathbf{a},\mathbf{z}) \tag{14}$$
$$s.t.\ (1-\lambda)\sum_{\{\mathbf{x},\mathbf{y}\}\in S,\{\mathbf{a},\mathbf{z}\}\in A_{\{\mathbf{x},\mathbf{y}\}}} X(\mathbf{x},\mathbf{y},\mathbf{a},\mathbf{z}) = 1$$
$$\sum_{\{\mathbf{x},\mathbf{y}\}\in S,\{\mathbf{a},\mathbf{z}\}\in \mathbf{A}_{\{\mathbf{x},\mathbf{y}\}}} X(\mathbf{x},\mathbf{y},\mathbf{a},\mathbf{z})\left(x_n-\lambda x_{n+1}-\lambda\sum_{i,k}a_{ik(n+1)}\right)$$
$$\geq E_\alpha(X_n),\ n=1,\ldots,N.$$
$$\sum_{\{\mathbf{x},\mathbf{y}\}\in S,\{\mathbf{a},\mathbf{z}\}\in A_{\{\mathbf{x},\mathbf{y}\}}} X(\mathbf{x},\mathbf{y},\mathbf{a},\mathbf{z})(y_{ik}-\lambda E_\alpha(Y_{ik}))$$
$$\geq E_\alpha(Y_{ik}),\ i=1,\ldots,I,\ k=1,\ldots,N.$$
$$X\geq 0.$$

The dual of our LP is still intractable because it has a variable for each state-action pair. The column generation approach deals with the intractability of the above LP by solving a linear program that has a small number of variables. Then it solves a new LP at each iteration by adding new variables determined by finding a violated constraint in the primal. The algorithm stops when either no violated constraints can be found or a certain stopping condition holds. In our problem, finding the most violated constraint is given by

$$z(\mathbf{V},\mathbf{W}) = \min_{\{\mathbf{x},\mathbf{y}\}\in S,\{\mathbf{a},\mathbf{z}\}\in \mathbf{A}_{\{\mathbf{x},\mathbf{y}\}}} \sum_{i,k,n} b(i,k,n)a_{ikn}$$
$$+\sum_{i=1}^{I} d(i)\sum_{k=1}^{N} z_{ik} - \sum_{n=1}^{N} V_n\left(x_n-\lambda x_{n+1}-\lambda\sum_{i,k}a_{ik(n+1)}\right)$$
$$-\sum_{i=1}^{I}\sum_{k=1}^{N} W_{ik}(y_{ik}-\lambda E_\alpha(Y_{ik}))-(1-\lambda)W_0. \tag{15}$$

Fortunately, the above integer program is optimally solved in a reasonable amount of time, and therefore we are able to compute the optimal values of the approximation parameters.

Appendix B: Results for the relaxed tolerance case with high early/late scheduling costs

Table 8 The results for the relaxed tolerance early scheduling case with constant arrival rates

C	D.C.	V_n	ADP	Tolerance	Capacity	Earliest	Latest
	75		0.99 − 4.79− 34350	0.99 − 4.75− 35557	**1.00−4.33**− 32843	**1.00**−**4.44**− 32720	0.99 − 4.98− 37587
	100		0.99 − 4.79− 45790	0.99 − 4.60− 53498	1.00−**4.29**− 51867	1.00−**4.36**− 50730	0.99 − 4.80− 55664
80	125	>0	0.99 − 4.80− **57310**	1.00 − 4.33− 80756	1.00−**4.20**− 76805	1.00−**4.20** 76800	1.00 − 4.42− 83559
	250		0.99 − 4.65− **115815**	1.00 − 4.17− 186942	1.00−**4.13**− 185830	1.00−**4.11**− 182100	1.00 − 4.21− 190093
	75		−	0.97 − 2.05− 14209	**0.98**−**1.10**− **7254**	0.97 − 1.58− 10849	0.96 − 2.30− 16203
	100		−	0.97 − 1.93− 20052	**0.98**−**1.05**− **10915**	0.97 − 1.51− 15318	0.96 − 2.15− 22725
85	125	0	−	0.98 − 1.22− 31104	0.98−**0.82**− **18626**	0.98 − 0.93− 24287	0.97 − 1.37− 34894
	250		−	0.98 − 0.80− 56801	0.98 − 0.62− 38262	0.98 − 0.60− 44670	0.98 − 0.92− 63686
	75		−	0.93 − 0.80− 5501	0.93−**0.11**− **584**	0.93 − 0.53− 3483	0.92 − 0.94− 6279
	100		−	0.92 − 0.13− 1542	0.93−**0.10**− **798**	0.93 − 0.52− 4678	0.92 − 0.91− 8484
90	125	0	−	0.93 − 0.23− 10591	0.93−**0.04**− **1382**	0.93 − 0.13− 6868	0.93 − 0.28− 12069
	250		−	0.93 − 0.07− 14016	0.93−**0.02**− **1907**	0.93 − 0.03− 8719	0.93 − 0.09− 16143

Table 9 The results for the relaxed tolerance late scheduling case with constant arrival rates

C	D.C.	V_n	ADP	Tolerance	Capacity	Earliest	Latest
	75		0.99 − 4.70− 34380	0.99 − 4.75− 34090	1.00−**4.34**− **30640**	1.00−**4.45**− 31370	0.99 − 4.98− 36280
	100		0.99 − 4.64− 45960	0.99 − 4.59− 48350	1.00−**4.33**− 41260	1.00 − 4.38− 43970	0.99 − 4.74− 52250
80	125	>0	0.99−**4.50**− 57610	0.99 − 4.59− 59230	1.00−**4.33**− **51460**	1.00−**4.38** 54260	0.99 − 4.74− 63590
	250		1.00 − 4.27− **111660**	1.004.27− 157300	1.00 − 4.20− 153300	1.00 − 4.18− 151200	1.00 − 4.33− 160800
	75		−	0.97 − 2.03− 14186	**0.98**−**1.10**− **7126**	0.97 − 1.57− 10826	0.96 − 2.27− 16125
	100		−	0.97 − 1.69− 19903	**0.98**−**1.08**− **9704**	0.97 − 1.37− 15158	0.97 − 1.87− 23106

Table 9 (continued)

C	D.C.	V_n	ADP	Tolerance	Capacity	Earliest	Latest
85	125	0	–	0.97 – 1.69–23710	**0.98–1.08–12001**	0.97 – 1.37–18222	0.97 – 1.87–27405
	250		–	0.98 – 1.08–54462	0.98–**0.82–36367**	0.98 – 0.86–43039	0.98 – 1.20–59961
	75		–	0.93 – 0.79–5501	0.93–**0.11–584**	0.93 – 0.52–3483	0.92 – 0.93–6293
	100		–	0.93 – 0.53–7487	0.93–**0.10–787**	0.93 – 0.38–4754	0.93 – 0.59–8904
90	125	0	–	0.93 – 0.53–8659	0.93–**0.10–967**	0.93 – 0.38–5567	0.93 – 0.59–10219
	250		–	0.93 – 0.17–16185	0.93–**0.04–2649**	0.93 – 0.11–10899	0.93 – 0.21–18115

References

1. Adelman D, Klabjan D (2011) Computing near-optimal policies in generalized joint replenishment. INFORMS J Comput, doi:10.1287/ijoc.1100.0433
2. Bailey NT (1952) A study of queues and appointment systems in hospital out-patient departments, with special reference to waiting-times. J R Stat Soc B14:185–99
3. Bertsekas D, Tsitsiklis J (1996) Neuro-dynamic programming. Athena Scientific
4. Cayirli T, Veral E (2003) Outpatient scheduling in health care: a review of literature. Prod Oper Manag 12:519–549
5. Diedrich J, Plank DP (2003) Efficient system to schedule chemotherapy and support therapies for oncology nurses. Oncol Nurs Soc Congr Abstr, 76
6. Fries BE, Marathe VP (1981) Determination of optimal variable-sized multiple-block appointment systems. Oper Res 29:324–45
7. Gerchak Y, Gupta D, Henig M (1996) Reservation planning for elective surgery under uncertain demand for emergency surgery. Manag Sci 42(3):321–334
8. Gocgun Y, Bresnahan B, Ghate A, Gunn M (2011) A markov decision process approach to multi-category patient scheduling in a diagnostic facility. Artif Intell Med 53:73–81
9. Gocgun Y, Ghate A (2012) Lagrangian relaxation and constraint generation for allocation and advanced scheduling. Comput Oper Res 39:2323–2336
10. Green LV, Savin S, Wang B (2008) Managing patient service in a diagnostic medical facility. Oper Res 54:11–25
11. Hawley E, Carter NG (2009) An acuity rating system for infusion center nurse staffing: the cleveland clinic cancer center at hillcrest hospital experience. Oncol Issues November/December, 34–37
12. Klassen KJ, Rohleder TR (2004) Outpatient appointment scheduling with urgent clients in a dynamic, multi-period environment. Int J Serv Ind Manag 15(2):167–186
13. Kleywegt AJ, Papastavrou JD (1998) The dynamic and stochastic knapsack problem. Oper Res 46(1):17–35
14. Kolisch R, Sickinger S (2008) Providing radiology health care services to stochastic demand of different customer classes. OR Spectrum 30:375–395
15. Lamiri M, Xie X, Dolgui A, Grimaud F (2008) A stochastic model for operating room planning with elective and emergency demand for surgery. Eur J Oper Res 185(3):1026–1037
16. Langhorn M, Morrison C (2001) Redesigning processes in ambulatory chemotherapy: creating a patient appointment scheduling system: part II. Can Oncol Nurs J 11(3):157–159
17. Lindley DV (1952) The theory of queues with a single server. Math Proc Camb Philos Soc 48:277–289
18. Liu N, Ziya S, Kulkarni VG (2010) Dynamic scheduling of outpatient appointments under patient no-shows and cancellations. Manuf Serv Oper Manag 12(2):347–364
19. Min D, Yih Y (2010) An elective surgery scheduling problem considering patient priority. Comput Oper Res 37:1091–1099
20. Mondschein SV, Weintraub GY (2003) Appointment policies in service operations: a critical analysis of the economic framework. Prod Oper Manag 12:266–286
21. Patrick J, Puterman M, Queyranne M (2008) Dynamic multipriority patient scheduling for a diagnostic resource. Oper Res 56(6):1507–1525
22. Patrick J (2012) A Markov decision model for determining optimal outpatient scheduling. Health Care Manag Sci 15(2):91–102. doi:10.1007/s10729-011-9185-4
23. Powell WB (2007) Approximate dynamic programming: solving the curses of dimensionality. Wiley
24. Powell WB (2008) Approximate dynamic programming: lessons from the field. In: Proceedings of the 2008 winter simulation conference, pp. 205–214
25. Puterman M (1994) Markov decision processes. Wiley, New Jersey
26. Santibanez P, Aristizabal R, Chow VS, Huangh W, Kollmannsberger C, Nordin T, Runzer N, Puterman ML, Tyldesley S (2012) Improving chemotheraphy scheduling and delivery through process redesign and advanced analytics. Jt Comm J Qual Patient Saf 34(12):541–553
27. Saure A, Patrick J, Tyldesley S, Puterman ML (2012) Dynamic multi-appointment patient scheduling for radiation therapy. Eur J Oper Res 223:573–584
28. Turkcan A, Zeng B, Lawley M, Chemotherapy operations planning and scheduling. IIE Trans on Healthc Sys Eng, to appear