

RESEARCH ARTICLE

Dynamic scheduling with cancellations: an application to chemotherapy appointment booking

Yasin Göçgün

Department of Industrial Engineering, Altinbas University, Turkey
yasin.gocgun@altinbas.edu.tr

ARTICLE INFO

Article History:

Received 27 April 2017

Accepted 02 March 2018

Available 22 April 2018

Keywords:

Dynamic scheduling

Markov decision processes

Approximate dynamic programming

AMS Classification 2010:

90-08, 41-04, 60JXX

ABSTRACT

We study a dynamic scheduling problem that has the feature of due dates and time windows. This problem arises in chemotherapy scheduling where patients from different types have specific target dates along with time windows for appointment. We consider cancellation of appointments. The problem is modeled as a Markov Decision Process (MDP) and approximately solved using a direct-search based approximate dynamic programming (ADP) technique. We compare the performance of the ADP technique against the myopic policy under diverse scenarios. Our computational results reveal that the ADP technique outperforms the myopic policy on majority of problem sets we generated.



1. Introduction

Dynamic scheduling problems arise in diverse fields such as manufacturing, transportation, project management, and healthcare. In these problems, unlike static scheduling problems, jobs arrive randomly at the system. Depending on the field, jobs could be orders, patients, and tasks. Arriving jobs are allocated to resources such as machines, operating rooms, and surgery rooms. Dynamic scheduling problems have received considerable attention in the operations research field in the last decade.

One of the recent variations of dynamic scheduling problems considers jobs with target dates and time windows. In these problems, jobs arriving randomly at a facility are ideally scheduled to specific target dates; if this is not possible, they are scheduled to days within a time window. Scheduling jobs outside their time windows results in penalty. These problems typically arise in chemotherapy appointment booking where jobs are considered as patients. Patients from each type have specific target dates and tolerance limits. In such settings, inefficient patient scheduling causes excessive wait listing, late patient

appointment notifications, pharmacy congestion, unbalanced workload between nurses and considerable clerical work [1]. Similar problems also arise in manufacturing where jobs from different types are scheduled for production considering target dates. In such settings, early scheduling causes inventory cost, whereas late scheduling results in penalty cost.

Chemotherapy appointment booking takes into account treatment protocols which are designed to maximize the efficacy of a chemotherapy treatment. Treatment protocols specify things such as the drugs to be administered, the dosage, appointment duration, the number of days between treatments, and tolerance limits [1].

In chemotherapy settings, appointments may be cancelled due to the fact that patients' protocols may change. It is fair to say that cancellations were generally ignored in the earlier work on chemotherapy appointment booking since the inclusion of cancellations makes the respective mathematical model significantly complex. However, cancellations should be among important features of realistic models.

Towards that end, we consider the chemotherapy appointment booking problem with cancellations.

In line with the literature, we model the problem as a discounted infinite horizon Markov Decision Process (MDP). Owing to intractability in state and action spaces, we resort to direct-search based approximate dynamic programming (ADP) to approximately solve the problem.

In this paper, we make the following contributions:

- We consider chemotherapy patient appointment booking problem with cancellations of treatments.
- We formulate the problem as an MDP.
- We employ a direct-search based ADP technique for approximately solving the underlying MDP model.
- We compare the performance of the direct-search based ADP technique against a myopic policy.
- Our computational results reveal that the direct-search based ADP improves the solution of the myopic policy on majority of problem sets we generated.

The paper is structured as follows. Section 2 reviews the relevant literature. In Section 3, the chemotherapy appointment booking problem with cancellations is described and its MDP formulation is provided. Section 4 includes the description of the Direct search-based ADP. Numerical results are provided in Section 5. Section 6 includes concluding remarks.

2. Literature review

Patient scheduling has been widely studied in the literature [2–4]. Green et al. [2] studied the issue of designing the outpatient appointment schedule and establishing dynamic priority rules for admitting patients into service. They formulated the problem as a finite-horizon dynamic program and identified the properties of the optimal policy. Cardoen et al. [3] reviewed operations research papers that discuss operating planning and scheduling. Hulshof et al. [4] reviewed research on resource capacity planning and control in health care.

Based on how patients are scheduled, patient scheduling can be divided into allocation scheduling and advance scheduling. Allocation scheduling requires that arriving patient requests be served or rejected immediately, whereas in advance scheduling, they are scheduled to specific slots/days in a booking horizon. Examples of research on allocation scheduling can be found in [2, 5–7]. Since the focus of this research is on

advance scheduling, we review work on advance scheduling below.

Patrick et al. [8] studied problems where patients of different types are scheduled to future days in a diagnostic facility. They provided a Markov Decision Process (MDP) formulation of these problems; owing to intractability in state and action spaces, they employed a linear-programming based approximate dynamic programming (ADP) to obtain an approximate solution to underlying MDP.

Lamiri et al. [9] studied an operating room planning problem, considering two types of demand for surgery: elective surgery and emergency surgery. The planning problem is assigning elective cases to different periods over a planning horizon. The authors proposed a stochastic mathematical programming model to solve their problem.

Liu et al. [10] proposed a model for dynamic appointment scheduling problems, taking into account no-shows, cancellation, patient preferences, and overtime. Their results indicate that heuristic dynamic policies they proposed outperform benchmark policies.

Saure et al. [11] studied a dynamic patient scheduling problem in radiation therapy units. The authors provided an MDP formulation of their problem and employed linear programming-based ADP for obtaining an approximate solution. Their results reveal that the ADP technique outperforms the myopic policy in diverse problem sets.

Geng et al. [12] studied dynamic outpatient scheduling for a diagnostic facility with two waiting-time targets. The authors developed a finite-horizon MDP model for this problem, and characterized the optimal scheduling policy by proving the monotonicity and concavity properties of the components of the MDP model.

Tsai and Teng [13] proposed a stochastic appointment system for patients with a dynamic call-in sequence to outpatient clinics with multiple resources. In their model, the schedule for a single-service period includes a fixed number of blocks of equal length. Their results indicate that their stochastic model was able to schedule patients more efficiently as compared to traditional appointment systems.

There are several studies about patient scheduling on chemotherapy in the literature. These are summarized below.

Turkcan et al. [14] developed operations planning and scheduling methods for chemotherapy

patients, aiming to minimize deviation from optimal treatment plans due to limited availability of clinic resources. Their model differs from our model in that they used integer programming methods for solving their problem since they ignored random and dynamic arrival of patients.

Hahn-Goldberg et al. [15] studied dynamic optimization of chemotherapy outpatient scheduling with uncertainty. The authors developed dynamic template scheduling, combining proactive and on-line optimization.

Hahn-Goldberg et al. [16] modeled a deterministic chemotherapy outpatient scheduling problem using constraint programming. They compared the performance of their best constraint model against that of a mixed-integer programming (MIP) model. Their results revealed that constraint programming outperforms the MIP model.

In a recent study, Alvarado and Ntaimo [17] studied the problem of scheduling individual chemotherapy patient appointments and resources. The authors developed mean-risk stochastic integer programming models for solving their problem.

Our work differs from the abovementioned studies in that we consider dynamic scheduling with specific target dates and tolerance limits.

Gocgun and Puterman [1] extended the work in [8] by considering patients of different types with specific target dates and tolerance limits. They studied a chemotherapy patient appointment booking problem and formulated the problem as an MDP and employed linear-programming based ADP for approximately solving the MDP model. Their model differs from our model in that we consider cancellations of treatments.

3. Problem description and Markov decision process model

We consider the following dynamic patient scheduling problem (see [1] for a similar description).

- We consider an infinite time horizon and a finite rolling booking horizon. The booking horizon is a rolling period of N days.
- Patients are classified on the basis of their appointment tolerances.
- Each day patients of each type with specific target dates arrive at the facility randomly. Arrival distributions are assumed to be stationary, and arrivals across patient types and target dates are assumed to be independent.
- In line with the literature [1], we assume that each appointment requires one appointment slot.
- At the end of each day, scheduling to future days over a booking horizon is performed for arriving patients or those patients are diverted (i.e., they are sent to another hospital or served through overtime). (As stated in [1], diversion can be thought as overtime in chemotherapy settings or outsourcing in the manufacturing setting). Diversion/overtime capacity is assumed to be significantly higher than the maximum number of patients that need to be diverted on any day.
- There is no cost incurred for scheduling patients to days within their tolerance limits, whereas scheduling a patient to a day outside the tolerance limit results in a type-dependent scheduling cost per day.
- Diverting patients or serving them through overtime results in diversion cost, which is the same for patients of each type.
- We consider cancellation of appointments. In line with the related literature [18] we assumed that once an appointment is cancelled, it is not rescheduled to a future date. This is due to the fact that rescheduling cancelled appointments to a future day would further complicate the problem.
- The objective is to perform scheduling of arriving patients to available days (or divert them) in a booking horizon so as to minimize total discounted expected cost.

In line with the literature [1, 2, 19], we assume that scheduling costs for patients of each type are linear in the number of days outside their time window. The function of these costs is to penalize the system for violating time windows. As stated in [1], higher scheduling costs are incurred for patients with lower tolerances. The function of diversion costs is to penalize the system for not scheduling a patient [1].

It is worth noting that in our problem, the diversion is not necessarily considered as overtime. Utilizing overtime means the appointment can be dealt with through overtime on the day the decision is given. Hence, in line with the literature [1], we do not consider scheduling an appointment to a future day and at the same time performing it through overtime.

3.1. Markov decision process model

3.1.1. State Space

The state space $s \in S$ takes the following form:

$$s = (x_{ikn}, y_{11}, y_{12}, \dots, y_{IN}),$$

where I is the number of patient types, x_{ikn} for $i = 1, \dots, I, k = 1, \dots, N, n = 1, \dots, N$ is number of type- i patients with target date k scheduled to day n , y_{ik} for $i = 1, \dots, I, k = 1, \dots, N$ is number of type- i patients with target date k waiting to be scheduled.

3.1.2. Action sets

The decision to be made at the end of each day is to determine patients to book on specific days and diverted patients with specific target dates. $A_{x;y}$ denotes the set of available actions in state $s = (x; y)$. Any action in A_{xy} is represented by:

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

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 set of feasible actions must satisfy the following constraints:

$$\sum_{i=1}^I \sum_{k=1}^N x_{ikn} + \sum_{i=1}^I \sum_{k=1}^N a_{ikn} \leq C_1, n = 1, \dots, N \quad (1)$$

$$\sum_{i=1}^I \sum_{k=1}^N z_{ik} \leq C_2 \quad (2)$$

$$\sum_{k=1}^N \sum_{n=1}^N a_{ikn} + \sum_{k=1}^N z_{ik} = \sum_{k=1}^N y_{ik}, i = 1, \dots, I \quad (3)$$

where C_1 is daily resource capacity and C_2 is maximum number of patients diverted or served through overtime each day. Constraint 1 ensures that total number of patients scheduled on each day is limited by daily capacity. Constraint 2 dictates that total number of patients diverted (or served through overtime) is limited by diversion capacity. Constraint 3 ensures that all of the arriving patients must be either scheduled or diverted.

3.1.3. Transition probabilities

After all scheduling actions are taken, the number of new requests for each type of patient and number of treatments of each type that are cancelled affect the transition to the next state of the system.

Let d_{ikn} be the number of type- i patients with target date k booked on day n that are cancelled, and $Pr(q_{ik})$ be the probability that q_{ik} new requests for type- i patients arrived. The probability that d_{ikn} treatments for type- i patients with target date k scheduled to day n are cancelled is represented by the term $Q_{ikn}(d_{ikn})$, which is expressed as follows:

$$Q_{ikn}(d_{ikn}) = \binom{x_{ikn}}{d_{ikn}} p_{ikn}^{d_{ikn}} (1 - p_{ikn})^{x_{ikn} - d_{ikn}},$$

where p_{ikn} is the probability that a treatment of type i with target date k booked to day n is cancelled. On selecting decision a in state s , one component of the next state equals

$$(x'_{ikn}, y'_{ik}) = (x_{i,k,n+1} + a_{i,k,n+1} - d_{i,k,n+1}, y_{ik} - (\sum_{i=1}^I \sum_{n=1}^N a_{ikn} + \sum_{i=1}^I \sum_{k=1}^N z_{ik}) + q_{ik})$$

with probability

$$Pr(q_{ik})Q_{ikn}(d_{ikn})$$

Hence the state changes according to

$$P(s' | s, a) = \prod_{i=1}^I \prod_{k=1}^N Pr(q_{ik}) \prod_{i=1}^I \prod_{n=1}^N \prod_{k=1}^N Q_{ikn}(d_{ikn}),$$

if s' satisfies Eqs. (4),(5), and (6).

$$x'_{i,k,n} = x_{i,k,n+1} + a_{i,k,n+1} - d_{i,k,n+1}, \quad (4)$$

$$i = 1, \dots, I; k = 1, \dots, N; n = 1, \dots, N - 1$$

$$x'_{i,k,N} = 0; i = 1, \dots, I; k = 1, \dots, N \quad (5)$$

$$y'_{i,k} = y_{ik} - (\sum_{i=1}^I \sum_{n=1}^N a_{ikn} + \sum_{i=1}^I \sum_{k=1}^N z_{ik}) + q_{ik}, \quad (6)$$

$$i = 1, \dots, I; k = 1, \dots, N$$

Equations (4) and (5) show the new number of type- i patients with target date k booked on day n . Equation (6) defines the new number of type- i patients with target date k waiting to be scheduled.

3.1.4. Costs

As stated in [1], the cost of scheduling a type- i patient with target date k to day n is denoted $b(i, k, n)$. It is given by

$$b(i, k, n) = \begin{cases} 0, & \text{if } L_i \leq n \leq U_i \\ (L_i - n)c_1^i & \text{if } L_i \geq n \\ (n - U_i)c_2^i & \text{if } n > U_i, \end{cases} \quad (7)$$

where L_i for $i = 1, \dots, I$ and U_i for $i = 1, \dots, I$ are lower and upper tolerance limits for type- i patients, respectively. c_1^i for $i = 1, \dots, I$ and c_2^i for $i = 1, \dots, I$ are unit early and unit late costs for type- i patients, respectively. The immediate cost is expressed as

$$c(a, z) = \sum_{i,k,n} b(i, k, n)a_{ikn} + \sum_{i=1}^I d(i) \sum_{k=1}^N z_{ik} \quad (8)$$

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

3.1.5. Bellman's equations

Discounting with discount factor λ is assumed in our model. Bellman's equations for finding a policy that minimizes the expected infinite horizon discounted cost are expressed as follows:

$$v(x, y) = \min_{(a,z) \in A_{x,y}} \left\{ c(a, z) + \lambda \sum_{y' \in D} (y')v(x', y') \right\}, \quad (9)$$

where D is the set of all possible demand vectors. However, our MDP model is intractable owing to the fact that state and action spaces grow exponentially with the number of patient types and the length of the booking horizon. Therefore we resort to approximate dynamic programming (ADP) for approximately solving our model. The ADP technique we utilized in this research is described next.

4. Direct-search based approximate dynamic programming

ADP has been extensively used for solving intractable MDPs in diverse fields such as manufacturing, healthcare, and revenue management [20]. ADP techniques are mainly classified as linear programming (LP)-based ADPs and simulation-based ADPs. In the LP-based ADP approach, the underlying MDP model is transformed into the equivalent LP version of Bellman's equations, and then approximate value function is used to make the LP model tractable [21–23]. Whereas simulation-based ADP techniques find an approximate solution to the Bellman's equations by simulating the evolution of the system over a number of initial states in order to tune the parameters [24,25]. They employ simulation models such as statistical sampling and reinforcement learning methods for estimating the value functions [26].

In ADP, the value function is approximated through a combination of basis functions, which represent some important features of the state of the system. There are certain ways for doing so, and one of them is linear approximation, which takes the following form:

$$V(s) \approx \sum_{k=1}^K r_k \Phi_k(s),$$

where r_k for $k = 1, \dots, K$ are tuning parameters and $\Phi_k(s)$ for $k = 1, \dots, K$ are basis functions. After the value function is approximated, an ADP policy is obtained by tuning the approximation parameters iteratively. In particular, the goal of ADP techniques is to find the optimal parameter vector that minimizes a certain performance metric such as the sum of squared differences between the approximate cost-to-go function and the estimated cost-to-go function over sampled states [27]. The resulting optimization problem is generally solved using regression-based techniques [20, 24]. However, we utilize an approach that solves an optimization problem for tuning the ADP parameters to achieve the best policy resulting from those parameters. We then obtain the ADP policy using the approximate value functions.

4.1. Retrieving the ADP Policy from the Approximate Value Function

After the end of the parameter tuning phase which makes the approximate value of a given state available, the ADP policy is retrieved by computing a decision vector for any desired state of the system. As stated in [27], that decision vector is myopic with respect to the value function approximation of our MDP. The decision retrieval problem for a particular state s of our MDP model is given by

$$\min_{(a,z) \in A_{x,y}} \left\{ c(a, z) + \lambda \sum_{y' \in D} (y')\tilde{v}(x', y') \right\}, \quad (10)$$

where $\tilde{V}(x', y')$ is the approximate value of state (x', y') .

Details about basis functions for the ADP implementation are discussed next.

4.2. Basis Functions

The basis function we chose for the ADP technique are as follows:

$$\begin{aligned}
 \Phi(s) = & \sum_{k,n:n=k} (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{1kn})^2/2 \\
 & + \sum_{k=2}^N \sum_{n=1:n=k-1}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{2kn})^2/8 \\
 & + \sum_{k=1}^N \sum_{n=1:n=k}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{2kn})^2/8 \\
 & + \sum_{k=1}^{N-1} \sum_{n=1:n=k+1}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{2kn})^2/8 \\
 & + \sum_{k=3}^N \sum_{n=1:n=k-2}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{3kn})^2/32 \\
 & + \sum_{k=2}^N \sum_{n=1:n=k-1}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{3kn})^2/32 \\
 & + \sum_{k=1}^N \sum_{n=1:n=k}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{3kn})^2/32 \\
 & + \sum_{k=1}^{N-1} \sum_{n=1:n=k+1}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{3kn})^2/32 \\
 & + \sum_{k=1}^{N-2} \sum_{n=1:n=k+2}^N (C_1 - \sum_{i=1}^I \sum_{kk=1}^N x_{i,kk,n} - a_{3kn})^2/32.
 \end{aligned}$$

Our basis function utilizes available capacity for each day in retrieving the ADP policy and dictates that patients with lower tolerance limits have higher priority when scheduling is performed. The first term corresponds to available capacity for tolerance (0,0), the next three terms correspond to available capacity for tolerance (1,1), and the next five terms correspond to available capacity for tolerance (2,2).

4.3. Direct Search

We tune parameters using direct search (see [27] and [28] for the implementation of direct search). Unlike regression-based techniques, direct search considers the ultimate goal of finding good policies. In particular, direct search deals with an optimization problem where the variables consist of feasible r 's and the objective function value is the expected cost of the policy induced by the corresponding parameter vector [27]. The resulting optimization problem is expressed as follows:

$$\min_{r \in R^N} E \left[\sum_{t=0}^T c(s_t, \pi_r(s_t)) \right], \tag{11}$$

where T is a random variable denoting the final step of the search, s_t is the state at step t of the search, π_r is the policy obtained by the parameter vector r , $\pi_r(s_t)$ is the action dictated by the policy π_r in the state at stage t , and $c(s_t, \pi_r(s_t))$

is immediate cost incurred at step t as a result of choosing $\pi_r(s_t)$. Here, π_r is obtained by solving the aforementioned decision retrieval problem via the parameter vector r used to approximate the value function for each possible state visited during the search. The objective function in Eq. (11) is the expected cost of the policy π_r [27].

During the implementation of the direct search-based ADP, we let r range from 0 to 10000 in increments of 500. We choose the value based on the best policy performance when decisions are made by solving Equation 10.

5. Numerical results

This section focuses on the comparison of the performance of the direct search-based ADP policy with that of the myopic policy. The myopic policy is obtained by solving the problem $\min_{(a,z) \in A_{x,y}} c(a,z)$ for any state $s \in S$. We used AMPL with CPLEX 12 for solving all integer programs.

5.1. Experimental design

We consider a chemotherapy center where the length of the booking horizon is 10 (for larger problems the length of the booking horizon is set to 20). In line with the literature [1], we set the number of patient types to 3 and tolerance limits to (0,0), (1,1), and (2,2). We assume that arrival distribution is independent Poisson with mean 1 for each type for each day over the booking horizon. Probability that a scheduled treatment is cancelled has three levels: 0.01, 0.05, and 0.1. We assume capacity levels of 25 and 30; the former level corresponds to the low capacity case whereas the latter represents the medium capacity case owing to the fact that the total mean daily demand nearly equals 30 (for larger problems, those levels are set to 55 and 60.). In line with the literature [1], 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.

We set diversion cost per patient for each patient type to 50 and 250. The case when the diversion cost equals 50 is referred by [1] as "rigid tolerance case", whereas other choices for the diversion costs are referred as "relaxed tolerance cases". We also consider the cases where early (late) scheduling costs are much higher than late (early) scheduling cost for the relaxed tolerance case. In line with the literature [1], we set the unit early scheduling cost per day for each patient type to 300, 225, and 150, respectively for

the high early scheduling cost case whereas the unit late scheduling cost per day for each patient type was set to 250, 200, and 150, respectively for the high late scheduling cost case. Discount factor is set to 0.99.

By changing the length of the booking horizon, cancellation probability, levels of capacity, diversion cost, and scheduling cost, we generated 32 problem sets. Simulation run lengths are set to 50. For each problem set, we ran 50 independent simulations.

5.2. Results

We present our results in Tables 1-8. Each row of the below tables represents a different problem set. For each problem, we utilized *Paired - t* test to statistically identify whether the ADP policy or the myopic policy performs better. Total costs obtained by the ADP policy and the myopic policy are listed in the second and third column of each table, respectively. The percentage improvement obtained by the ADP policy over the myopic policy is given in the fourth column of each table. The final column of each table reveals whether there is a statistically significant improvement over the myopic policy.

We begin with the analysis of results for the rigid tolerance cases. As mentioned earlier, in the rigid tolerance case diversion cost per patient is 50. The results indicate that in reasonable-sized problems (i.e., $N = 10$), the ADP policy statistically performs better than the myopic policy except for the case where the level of capacity is 30 and cancellation probability is 0.5 (see Table 1). Additionally, the ADP policy generally performs better in low capacity situations as compared to medium capacity situations. It is also worth noting that the percentage difference between the two policies is significantly high in the case where the level of capacity is medium and cancellation probability is high. Finally, the superiority of the ADP policy over the myopic policy in the rigid tolerance case diminishes in large-sized problems (i.e., $N = 20$) (see Table 3).

In the relaxed tolerance case, the ADP policy significantly performs better than the myopic policy for reasonable-sized problems (see Table 2). The corresponding percentage improvement over the myopic policy is higher than that in the rigid tolerance case. Further, unlike in the rigid tolerance case, the performance of ADP increases when the level of capacity is increased from low to medium level. Yet, for large-sized problems, using ADP results in very small improvement when cancellation probability is small; if this probability is

high, the performance of ADP and the myopic policy turns out to be the same (see Table 4).

The results for the high early scheduling cost case indicate that the performance of the ADP policy is significantly higher than that of the myopic policy for reasonable-sized problems (see Table 5). The corresponding percentage improvement over the myopic policy is higher as compared to that for the rigid tolerance case. However, the performance of the ADP policy for this case decreases when the problem size increases (see Table 6). We have similar observations for the high late scheduling cost case (see Tables 7 and 8).

Table 1. Results for the rigid tolerance case with $N = 10$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(25,0.01)	4141	4434	6.6	YES
(25,0.05)	2137	2266	5.7	YES
(25,0.1)	938	956	1.8	YES
(30,0.01)	1584	1655	4.3	YES
(30,0.05)	326	327	0.2	NO
(30,0.1)	5	17	72	YES

Table 2. Results for the relaxed tolerance case with $N = 10$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(25,0.01)	22911	25348	9.6	YES
(25,0.05)	9852	10625	7.3	YES
(25,0.1)	3719	3796	2	YES
(30,0.01)	6892	8141	15.3	YES
(30,0.05)	1110	1356	18.1	YES
(30,0.1)	10	36	72	YES

Table 3. Results for the rigid tolerance case with $N = 20$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(55,0.01)	16737	17110	2.2	YES
(55,0.05)	3064	3064	0	NO
(55,0.1)	1621	1621	0	NO
(60,0.01)	11936	12010	0.6	NO
(60,0.05)	214	214	0	NO
(60,0.1)	0.9	0.9	0	NO

Table 4. Results for the relaxed tolerance case with $N = 20$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(55,0.01)	86732	90672	4.3	YES
(55,0.05)	13310	13310	0	NO
(55,0.1)	7499	7499	0	NO
(60,0.01)	59620	62248	4.2	YES
(60,0.05)	856	856	0	NO
(60,0.1)	2	2	0	NO

Table 5. Results for the relaxed tolerance high early scheduling case with $N = 10$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(25,0.01)	22995	25360	9.3	YES
(25,0.05)	9947	10693	7.0	YES
(25,0.1)	3824	3899	1.9	YES
(30,0.01)	6969	8019	13.1	YES
(30,0.05)	1289	1365	5.6	YES
(30,0.1)	12	45	73	YES

Table 6. Results for the relaxed tolerance high early scheduling case with $N = 20$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(55,0.01)	87078	91287	4.6	YES
(55,0.05)	13976	13976	0	NO
(55,0.1)	7526	7526	0	NO
(60,0.01)	59976	62650	4.3	YES
(60,0.05)	1087	1087	0	NO
(60,0.1)	0.9	0.9	0	NO

Table 7. Results for the relaxed tolerance high late scheduling case with $N = 10$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(25,0.01)	21797	23610	7.7	YES
(25,0.05)	10436	11621	10.2	YES
(25,0.1)	4288	4394	2.4	YES
(30,0.01)	7672	8477	9.5	YES
(30,0.05)	1401	1782	21.4	YES
(30,0.1)	12	38	67	YES

Table 8. Results for the relaxed tolerance high late scheduling case with $N = 20$.

(C_1, p_{ikn})	ADP	Myopic	% impr.	Sign.?
(55,0.01)	84263	87521	3.7	YES
(55,0.05)	14320	14320	0	NO
(55,0.1)	7866	7866	0	NO
(60,0.01)	59242	61145	3.1	YES
(60,0.05)	990	990	0	NO
(60,0.1)	0.9	0.9	0	NO

6. Conclusion

We studied a chemotherapy appointment booking problem where patients have specific target dates, are classified based on their tolerance limits, and are scheduled to days in advance. Unlike the relevant literature, we considered cancellations of treatments. We provided an MDP formulation

of this problem and because of huge state and action spaces, we approximately solved the problem using a direct search-based ADP.

Direct search-based ADP can be considered as a relatively new technique among other ADP techniques. In this research, we demonstrated that it can be a viable method for solving chemotherapy appointment booking problems. In particular, our work revealed that the performance of the myopic policy can be significantly improved through the implementation of direct search-based ADP. It is worth noting that further improvements may be achieved by trying various basis functions as part of direct search-based ADP.

Cancellation of the treatments in the chemotherapy appointment booking problem with target dates and tolerance limits were not considered in the relevant literature because of resulting computational challenges. By including cancellations into our model, we filled this gap in this area. As a future research, overbooking and no-shows can be incorporated into our model and direct search-based ADP can be utilized for solving other dynamic scheduling problems.

References

- [1] Gocgun, Y., & Puterman, M. L. (2014). Dynamic scheduling with due dates and time windows: an application to chemotherapy patient appointment booking. *Health Care Manag Sci.*, 17, 60-76.
- [2] Green, L. V., Savin, S., & Wang, B. (2006). Managing Patient Service in a Diagnostic Medical Facility. *Operations Research*, 54, 11-25.
- [3] Cardoen, B., Demeulemeester, E., & Belien, J. (2010). Operating room planning and scheduling: A literature review. *European Journal of Operational Research*, 201, 921-932.
- [4] Hulshof, P., Kortbeek, N., Boucherie, R., Hans, E., & Bakker, P. (2012). Taxonomic classification of planning decisions in health care: a structured review of the state of the art in OR/MS. *Health Systems*, 1, 129-175.
- [5] Gocgun, Y., Bresnahan, B., Ghate, A., & Gunn, M. (2011). A markov decision process approach to multi-category patient scheduling in a diagnostic facility. *Artificial Intelligence in Medicine*, 53, 73-81.
- [6] Kolisch, R., & Sickinger, S. (2008). Providing radiology health care services to stochastic demand of different customer classes. *OR Spectrum*, 30, 375-395.
- [7] Min, D., & Yih, Y. (2010). An elective surgery scheduling problem considering patient priority. *Computers and Operations Research*, 37, 1091-1099.
- [8] Patrick, J., Puterman, M. L., & Queyranne, M. (2008). Dynamic multi-priority patient scheduling for a diagnostic resource. *Operations Research*, 56, 1507-1525.
- [9] Lamiri M., Xie X., Dolgui, A., & Grimaud, F. (2008). A stochastic model for operating room planning with elective and emergency demand for surgery. *European Journal of Operational Research*, 185, 1026-1037.

- [10] Liu, N., Ziya, S. & Kulkarni, V. G. (2010). Dynamic scheduling of outpatient appointments under patient no-shows and cancellation. *Manufacturing and Service Operations Management*, 12, 347-364.
- [11] Saure, A., Patrick J., Tyldesley, S., & Puterman, M. L. (2012). Dynamic multi-appointment patient scheduling for radiation therapy. *European Journal of Operational Research*, 223, 573-584.
- [12] Geng, N., & Xiaolan, X. (2016). Optimal dynamic outpatient scheduling for a diagnostic facility with two waiting time targets. *IEEE Transactions on Automatic Control*, 61, 3725-3739.
- [13] Tsai, P. J., & Teng, G. (2014). A stochastic appointment scheduling system on multiple resources with dynamic call-in sequence and patient no-shows for an outpatient clinic. *European Journal of Operational Research*, 239, 427-436.
- [14] Turkcan, A., Zeng, B., & Lawley, M. (2012). Chemotherapy operations planning and scheduling. *IIE Transactions on Healthcare Systems Engineering*, 2, 31-49.
- [15] Goldberg, S., Carter, M., Beck, J., Trudeau, M., Sousa, P., & Beattie, K. (2014). Dynamic optimization of chemotherapy outpatient scheduling with uncertainty. *Healthcare Management Science*, 17, 379-392.
- [16] Goldberg, S., Beck, J., Carter, M., Trudeau, M., Sousa, P., & Beattie, K. (2014). Solving the chemotherapy outpatient scheduling problem with constraint programming. *Journal of Applied Operational Research*, 6, 135-144.
- [17] Alvarado, M. & Ntaimo, L. (2018). Chemotherapy appointment scheduling under uncertainty using mean-risk stochastic integer programming. *Healthcare Management Science*, 21, 87-104.
- [18] Parizi, M. S., & Ghate, A. (2016). Multi-class, multi-resource advance scheduling with no-shows, cancellations and overbooking. *Computers Operations Research*, 67, 90-101.
- [19] Kleywegt, A. J., & Papastavrou, J. D. (1998). The Dynamic and Stochastic Knapsack Problem. *Operations Research*, 46, 17-35.
- [20] Powell W. B. (2007). *Approximate Dynamic Programming: Solving the curses of dimensionality*. John Wiley and Sons.
- [21] Adelman, D. (2003). Price-directed replenishment of subsets: methodology and its application to inventory routing. *Manufacturing and Service Operations Management*, 5, 348-371.
- [22] Adelman, D. (2004). A price-directed approach to stochastic inventory routing. *Operations Research*, 52, 499-514.
- [23] De Farias, D. P., & Roy, B. V. (2004). The linear programming approach to Approximate Dynamic Programming. *Operations Research*, 51, 850-865.
- [24] Bertsekas, D., & Tsitsiklis, J. (1996). *Neuro-Dynamic Programming*. Athena Scientific.
- [25] Sutton, R. S. & Barto, A. G. (1998). *Reinforcement Learning*. MIT Press.
- [26] Chang, H. S., Fu, M. C., Hu, J., & Marcus, S. I. (2007). *Simulation-based algorithms for Markov Decision Processes*. Springer.
- [27] Gocgun, Y. (2018). Approximate dynamic programming for optimal search with an obstacle (Submitted).
- [28] Maxwell, M. S., Henderson, S. G., & Topaloglu, H. (2013). Tuning Approximate Dynamic Programming Policies for Ambulance Redeployment via Direct Search. *Stochastic Systems*, 3, 1-40.

Yasin Göçgün earned his B.S. degree in industrial engineering from Bilkent University in Turkey in 2003. He earned his M.S. degree in industrial engineering from the same university in 2005. Dr. Gocgun earned his Ph.D. degree in industrial engineering from the University of Washington in 2010. He then worked as a postdoctoral fellow at the University of British Columbia in Vancouver, Canada for two years. After his first post-doc, he worked as a postdoctoral fellow at the University of Toronto in Toronto, Canada for two years. He has been working as an assistant professor in the Department of Industrial Engineering at Altinbas University in Istanbul, Turkey since September 2014. His research interests are stochastic dynamic optimization, operations research in healthcare, Markov decision processes, approximate dynamic programming, scheduling, and simulation.

An International Journal of Optimization and Control: Theories & Applications (<http://ijocta.balikesir.edu.tr>)



This work is licensed under a Creative Commons Attribution 4.0 International License. The authors retain ownership of the copyright for their article, but they allow anyone to download, reuse, reprint, modify, distribute, and/or copy articles in IJOCTA, so long as the original authors and source are credited. To see the complete license contents, please visit <http://creativecommons.org/licenses/by/4.0/>.