



University of  
New Haven

University of New Haven  
**Digital Commons @ New Haven**

---

Mechanical and Industrial Engineering Faculty  
Publications

Mechanical and Industrial Engineering

---

6-2017

# Strategic Level Proton Therapy Patient Admission Planning: A Markov Decision Process Modeling Approach

Shengfan Zhang

*University of Arkansas, Fayetteville*

Ridvan Gedik

*University of New Haven, rgedik@newhaven.edu*

Chase Rainwater

*University of Arkansas, Fayetteville*

Follow this and additional works at: <http://digitalcommons.newhaven.edu/mechanicalengineering-facpubs>



Part of the [Industrial Engineering Commons](#), and the [Mechanical Engineering Commons](#)

---

## Publisher Citation

Gedik, R., Zhang, S., & Rainwater, C. (2016). Strategic level proton therapy patient admission planning: a Markov decision process modeling approach. *Health Care Management Science*, 20(2), 286–302.

## Comments

This is the authors' accepted version of the article published in *Health Care Management Science*. The final publication is available at Springer via <http://dx.doi.org/10.1007/s10729-016-9354-6>

# Strategic Level Proton Therapy Patient Admission Planning: A Markov Decision Process Modeling Approach

Ridvan Gedik ·  
Shengfan Zhang ·  
Chase Rainwater

the date of receipt and acceptance should be inserted later

**Abstract** A relatively new consideration in proton therapy planning is the requirement that the mix of patients treated from different categories satisfy desired mix percentages. Deviations from these percentages and their impacts on operational capabilities are of particular interest to healthcare planners. In this study, we investigate intelligent ways of admitting patients to a proton therapy facility that maximize the total expected number of treatment sessions (fractions) delivered to patients in a planning period with stochastic patient arrivals and penalize the deviation from the patient mix restrictions. We propose a Markov Decision Process (MDP) model that provides very useful insights in determining the best patient admission policies in the case of an unexpected opening in the facility (i.e., no-shows, appointment cancellations, etc.). In order to overcome the curse of dimensionality for larger and more realistic instances, we propose an aggregate MDP model that is able to approximate optimal patient admission policies using the fixed weight aggregation technique. Our models are applicable to healthcare treatment fa-

Ridvan Gedik  
Department of Mechanical and industrial Engineering  
University of New Haven  
Tel.: (203) 9321167  
E-mail: rgedik@newhaven.edu

Shengfan Zhang  
Department of Industrial Engineering  
University of Arkansas  
Tel.: (479) 575-3571  
E-mail: shengfan@uark.edu

Chase Rainwater  
Department of Industrial Engineering  
University of Arkansas  
Tel.: (479) 575-2687  
E-mail: cer@uark.edu

cilities throughout the United States, but are motivated by collaboration with the University of Florida Proton Therapy Institute (UFPTI).

**Keywords** Patient admission policy · proton therapy · Markov decision process · state aggregation

## 1 Introduction

Proton therapy is a relatively new and fast growing form of radiation therapy for cancer patients. Its strength relies on its ability to transform higher doses of radiation to cancer cells with more accuracy than the traditional X-ray radiation techniques. Thus, the healthy tissue surrounding the cancer cells and vital organs are protected as the right doses of radiation is applied to the target cells. Due to the effectiveness and limited availability of proton therapy, the demand for this form of treatment is extremely high in relationship to the available capacity. Moreover, it is an expensive treatment procedure to deliver. According to the study by Goitein et al. [20], the cost-per-fraction in proton therapy is more than twice the cost-per-fraction of X-ray therapy. A portion of these high costs can be attributed to the small number of proton therapy facilities in the country. An equally important driver of cost is the meticulous process that must be followed to successfully deliver this form of treatment. In fact, in radiation therapy, a treatment protocol consists not only of a prescribed total delivered dose to so-called targets, but must also specify treatment times, sequences, and frequencies (fractionation schedule). It is known that dose fractionation contributes to the preservation of healthy tissue throughout the treatment (see, e.g., Yamada et al. [42]). However, a shorter fractionation schedule provides a more economical use of the radiation therapy facilities while still improving, albeit marginally, a patient's quality of life (see, e.g., Shelley et al. [39]). Because of these important tradeoffs, achieving the most efficient patient admission policies is a growing area of interest to healthcare professionals. Although proton therapy and other healthcare planning and scheduling problems share common properties (treatment continuity, capacity, staff/physician requirements, etc.), patient admission planning in proton therapy differs by aiming to accommodate patients from different categories without assigning them any priority weights related to stage of their disease. Instead, categories represent different patient types (i.e., chest/abdomen, prostate, brain cancer patients). Each one of these types has different treatment time and number of fractions to be delivered by a *gantry*. A Gantry is a device used for rotating the radiation delivery apparatus around the patient during

the treatment so that the body can be treated from different angles.

Stochastic patient arrivals and appointment cancellations (no-shows) are the most common sources of uncertainty in patient scheduling and admission planning problems. All other studies [26, 36] assume that the issue of treatment session cancellations is less significant in proton therapy treatment planning than in other healthcare applications. Due to the large demand for this therapy, it has been assumed that any appointment cancellations or no shows can be instantaneously replaced by the patients waiting in the line. Hence, these studies tackle the proton therapy patient scheduling and admission planning problem by formulating (mixed) integer linear programs. Gedik [17] proposes a bicriteria integer programming model that obtains capacity planning strategies for a proton therapy facility with several operational restrictions. Since it is very difficult to satisfy an integer number of patients in light of the mix requirements in addition to several other challenging constraints, they relax the integrality restrictions on the decision variables and solve the bicriteria problem under several operational settings in order to assess the tradeoffs between maximizing the total treatment sessions and minimizing the total deviation from the patient mix requirements. Men [26] and Salari et al. [36] propose a rolling horizon based dynamic algorithm that provides near-optimal solutions to a very similar integer programming model designed for patient allocations to a proton therapy facility that aims to maximize the total number of treated patients. Several other researchers utilize the integer programming approaches in providing efficient solutions for emerging healthcare problems. Agnetis et al. [3] and Agnetis et al. [2] develop integer programming models in order to support decision makers in a hospital in making fast operating room planning and scheduling decisions. On the other hand, Ciavotta et al. [11] and Ciavotta et al. [10] investigate intelligent solutions for operations scheduling of activities in pharmaceutical manufacturing plants and other healthcare operations where the problem has multiple objectives and is solved by rollout algorithms. As the costs associated with building proton therapy facility decrease due to advancing technology, handling stochastic patient arrivals will be vital in the near future to keep the utilization of the resources at their maximum levels. Thus, we propose a Markov decision process (MDP) model that accounts for gantry capacity and patient mix requirements as well as stochastic patient arrivals to a proton therapy facility. Even though this is a powerful way of handling stochasticity associated with patient arrivals, it takes quite amount of computational efforts

to solve this multi-category patient admission planning problem due to the exponentially growing state space. Therefore, this study also proposes an approximate aggregate MDP model to overcome this special type of *curse of dimensionality*. The aggregate MDP model clusters original states into subgroups that results in a more tractable state space. Of course, the aggregate MDP provides only approximate optimal patient admission policies. Thus, we also assess the trade-off between the traditional and aggregated models in terms of solution time and quality.

Patient mix optimization, the requirement that the mix of patients treated satisfy desired percentages, is a relatively new consideration in proton therapy planning. These percentages are determined by the decision managers before each planning horizon in order to be used for the upcoming resource allocation problems. From a managerial perspective, patient mix preferences can address how to plan resources with respect to the needs of different patient types and, from a clinical perspective, they refer to the conditions of patients that are being treated in the facility and the difficulty level of associated treatments [6]. Hence, a well planned patient mix preferences in a facility can help decision makers improve the efficiency of resources and health care providers position their treatment capabilities with the projected treatment types required by patients. *Diagnosis Related Groups (DRGs)* is a very similar concept that has been used by health care providers to classify patient groups based on patient types (i.e. case mix, patient mix) and their treatment costs incurred by hospitals [6]. DRGs play an important role in Medicare's hospital reimbursement system, whereas a hospital's case mix or patient mix preferences determine the costs of all the services provided for different patient types.

Petrovic et al. [29] highlight the fact that the radiotherapy patient admission planning problem is very complex since it accounts for several constraints such as large number of patients and patient categories, machine and doctor availability. Furthermore, several objectives including but not limited to minimizing the average waiting time, the number of patients who do not meet waiting time targets, machine idle time, overtime slots, costs and so on might need to be considered while dealing with this problem. For instance, Petrovic et al. [30] investigate the best patient admission policy to a radiation therapy center that minimizes both the average waiting time and average tardiness of patients. Gedik [18] illustrates the impacts of patient mix preferences on the capacity of a proton therapy facility by comparing the two objectives: (i) total treated number of patients and (ii) total deviations from patient

mix levels. In the presence of several operational constraints and deterministic patient arrivals, the tradeoff between these two objectives is quantified by identifying the efficient frontiers. The patient/case mix optimization studies analyzed in Section 2 aim to find the most suitable patient mix levels such that the maximum benefit is achieved. On the other hand, the UFPTI focuses on patient scheduling where the patient mix levels are taken as an input and the impacts of their interactions with the other operational constraints are demonstrated on the facility capacity. Hence, we formulate the objective function in order to maximize the total number of fractions delivered to all patient categories and at the same time, minimize the deviations from the predetermined patient mix levels for each patient type. In terms of constraints, only two important restrictions that play the most crucial roles in determining the best patient admission strategy are taken into account in our model: (i) gantry capacities and (ii) patient mix restrictions. Other operational level restrictions such as anesthesia surveillance, gantry specialization and snout change times can easily be adopted by the models introduced here with some additional changes.

The rest of this study is organized as follows. Section 2 describes how stochastic patient arrival and admission planning in healthcare facilities have been handled by MDP models in the literature. Then, two MDP models, exact and aggregate, are introduced in Section 3. Section 4 demonstrates the performance of these models, and finally, Section 5 briefly highlights the contributions of this study and discusses future research directions.

## 2 Literature Review

Adan and Vissers [1] categorize patient admissions into two groups: non-scheduled and scheduled. Non-scheduled admissions are unplanned and might be due to emergency cases whereas scheduled patients are planned and selected from a waiting list created as the service is being requested. In most of the studies (e.g., Bowers and Mould [8], Cayirli et al. [9], Pham and Klinkert [31]), scheduled (elective) cases are considered in order to reduce potential uncertainties associated with patient attributes (i.e., patient type, financial gain, resource allocation per patient, etc.). Some other studies assume that the treatment time for an outpatient is constant for the sake of simplicity [12]. In the context of this study, we assume that proton therapy patients are elective outpatients who do not need to spend the night at the facility.

Conforti et al. [13] compare the two most common types of daily basis radiation therapy patient admission planning strategies: (i) block system and (ii) non-block system. In a block system, a workday is composed of a fixed number of blocks/time slots with the same duration. On the other hand, in the non-block scheduling, different time intervals are reserved for different patients based on the type of the treatments. Furthermore, patients in the non-block system are usually admitted on a first-come first-served basis unless the individual cases are not restricted by strict earliest and/or latest start constraints. In the same study, after highlighting the fact that the block scheduling strategy is more commonly used than its counterpart in many radiation therapy centers, a patient scheduling model which accounts for non-block scheduling rules is developed for a short planning period (a week). In reality, non-block appointment system is conceptually superior to the block appointment system since the former is able to represent the whole workload. This is because, during a workday, accumulation of the idle time leftover from a time slot can be used to treat other patients who have not been admitted yet.

Despite of its advantages, the non-block scheduling appointment strategy is not considered very often in the literature due to the extra limitations/constraints it adds to the treatment planning problem. On the other hand, block scheduling is commonly encountered [12, 23, 19, 28, 40], since it simplifies most of the concepts that are hard to be taken into account in a radiation therapy patient scheduling problem. Although proton therapy treatment session durations may vary for different patient groups, we assume that the strategic level proton therapy patient admission planning problem has a block appointment system since our objective is to investigate the impacts of mix restrictions on patient admission planning decisions from a strategic point of view. In addition, the block appointment system simplifies patient admission decisions to a certain extent in an environment *with stochastic arrivals*. In such a system, a group of patients can be assigned to time slots in such a way that each of them can concurrently be receiving treatments. Figure 1 demonstrates a basic block patient allocation schema for a configuration of three gantries and three time slots in a given day. For each time slot, the number of patients assigned to a time slot cannot be greater than the number of gantries available in that time slot in the system. Note that the daily treatment times of all gantries are identically distributed to time slots of a day as demonstrated in Figure 1. Therefore, the total number of patients that can be treated in all gantries in a day would be equivalent to,  $Q$ , the total

number of time slots allocated to all gantries in a given day (i.e.,  $Q = 9$  in Figure 1).

A more recent contribution that is directly related to proton therapy patient admission planning corresponds to the work of Nunes et al. [28]. They present an MDP model in order to control the patient admissions from different specialties on a periodic basis. Patient flows from  $m$  distinct specialties are assumed to be continuous. In order to measure the resource consumption, they adopt the *treatment pattern* methodology, which is first proposed by Kapadia et al. [22]. It is assumed that a patient could follow  $n$  different treatment patterns until he is discharged. In this study, state space of the hospital is defined as the number of patients from all specialties following different treatment patterns. The number of patients to be admitted to the hospital in each specialty for the next decision period is considered as the action space. Finally, the stochastic dynamics of the study is described as the probability of one patient's transition from a treatment pattern to all others.

One of the first MDP approaches developed for hospital admission planning is proposed by Kolesar [24] who transform the MDP model into a linear program to obtain results. Another important finding of this study is that simultaneous reservations for different services are modeled over a planning horizon. Gocgun et al. [19] develop a finite horizon MDP to model a patient scheduling problem that is commonly seen in providing computed tomography (CT) service. They assume that there are only four patient types who demand the CT service during a work day period. The MDP model in their study aims to maximize the total profit obtained by scheduling different patient types for the available time slot of a CT machine during the work day. Different from most of the patient planning and scheduling models, they adopt each available time slot for each CT machine as a *stage*. Similar to the study conducted by Nunes et al. [28], the number of patients from different types constitutes the state space and the actions are denoted by the number of admitted patients from different types for the next service slot. They compare the optimal solutions obtained from their model with simple heuristic rules (first-come first-served (FCFS), randomized, etc.) typically employed to admit patients in real life. Even though their model finds better values in terms of total profit, FCFS heuristic provides better values in terms of the average number of patients not scanned by the end of the day.

### 3 Solution Methodology

#### 3.1 Markov Decision Process (MDP) Model

In light of the insights developed by Nunes et al. [28], Kapadia et al. [22], Gocgun et al. [19], we model our strategic patient mix optimization model as an MDP. Based on the block appointment schema in Figure 1, the finite state and action space of our problem can be modeled as in equations (1) and (2).  $s_{t,q}^k$  represents the number of patients being treated from category  $k$  in service slot  $q$  between decision instants  $t - 1$  and  $t$ . Thus, the state space  $S$  comprises all possible states. Furthermore,  $a_q^k$  represents the number of patients in each category that will be admitted for treatment in service slot  $q$  in the next period.

$$S = \left( \begin{array}{c} s_{t,q=1}^1 \\ s_{t,q=1}^2 \\ \vdots \\ s_{t,q=1}^{|K|} \end{array} \right), \left( \begin{array}{c} s_{t,q=2}^1 \\ s_{t,q=2}^2 \\ \vdots \\ s_{t,q=2}^{|K|} \end{array} \right), \dots, \left( \begin{array}{c} s_{t,q=Q}^1 \\ s_{t,q=Q}^2 \\ \vdots \\ s_{t,q=Q}^{|K|} \end{array} \right) \quad (1)$$

$$A_{s_t} = \left( \begin{array}{c} a_{q=1}^1 \\ a_{q=1}^2 \\ \vdots \\ a_{q=1}^{|K|} \end{array} \right), \left( \begin{array}{c} a_{q=2}^1 \\ a_{q=2}^2 \\ \vdots \\ a_{q=2}^{|K|} \end{array} \right), \dots, \left( \begin{array}{c} a_{q=Q}^1 \\ a_{q=Q}^2 \\ \vdots \\ a_{q=Q}^{|K|} \end{array} \right) \quad (2)$$

In addition to the number of current patients receiving treatments and newly accepted patients from each category, we also need to account for the ones whose treatment ends on each day. From each patient type  $k$  on day  $t$ , it is known that there will be  $a_{t-n_k}^k$  number of discharges, which is the number of patients from type  $k$  accepted on day  $t - n_k$  given that a patient from category  $k$  must receive fractions during  $n_k$  consecutive days. Incorporating the discharged patients into the state definitions would increase the complexity and degrade the tractability of the model. Therefore, a useful assumption is made in order to avoid this foreseen difficulty. Accordingly, discharges are assumed to be handled after accepting new patients to each category. Hence, transition from one state to another includes only the newly accepted patients and the ones that are receiving their treatments.

In the case of identical gantries, we can modify the gantry-time slot representation of a single day as in Figure 2. Accordingly, state space and action space can be defined in a more compact way as in equations (3) and (4). Based on the representation in Figure 2, both time slots and the gantries are assumed to be identical on a given day. Moreover, for the same proton therapy patient planning problem with deterministic patient arrivals, Gedik [18] has identified that if the daily gantry

Day 1			
	Time Slot 1	Time Slot 2	Time Slot 3
Gantry 1	≤ 1	≤ 1	≤ 1
Gantry 2	≤ 1	≤ 1	≤ 1
Gantry 3	≤ 1	≤ 1	≤ 1

Fig. 1: Block appointment system

capacity depends only on gantry type not time period, an optimal solution exists in which they start and treat the same number of patients in category  $k$  on each day  $t$  and in each gantry  $g$ . In other words, this implies that the optimal solution of the patient admission planning problem *enforces patient types to hold the desired patient mix restrictions on daily basis*. In order to represent this,  $s^k$  is defined as the total number of type  $k$  patients receiving treatments in all time slots and  $a^k$  is defined as the accepted number of patients from category  $k$  to all time slots.

$$S = \begin{pmatrix} s_{q=1}^1 + \dots + s_{q=Q}^1 \\ s_{q=1}^2 + \dots + s_{q=Q}^2 \\ \vdots \\ s_{q=1}^{|K|} + \dots + s_{q=Q}^{|K|} \end{pmatrix} = \begin{pmatrix} s^1 \\ s^2 \\ \vdots \\ s^{|K|} \end{pmatrix} \quad (3)$$

$$A_s = \begin{pmatrix} a_{q=1}^1 + \dots + a_{q=Q}^1 \\ a_{q=1}^2 + \dots + a_{q=Q}^2 \\ \vdots \\ a_{q=1}^{|K|} + \dots + a_{q=Q}^{|K|} \end{pmatrix} = \begin{pmatrix} a^1 \\ a^2 \\ \vdots \\ a^{|K|} \end{pmatrix} \quad (4)$$

Due to the problem definition, patient mix ratios ( $d_k$ ) for each category  $k \in K$  must be satisfied and the maximum number of fractions must be treated over a planning period. In order to assess the impacts of these constraints, we represent patient mix preferences in the optimality equations. One procedure is to assign a penalty for those patient types that fail to conform to the desired patient mix levels. To implement this procedure, we must keep track of the total number of patients treated (after the actions are taken), and then compare both the actual and desired patient mix levels. Let  $\mathbb{J}$  be the total number of patients from all types being treated in the facility.

$$\mathbb{J} = \sum_{k \in K} \{s^k + a^k\} \quad (5)$$

Then, the desired ( $D_k$ ) and actual ( $H_k$ ) patient mix of patient type  $k$  can be calculated as follows.

$$D_k = d_k \mathbb{J} \quad (6)$$

$$H_k = s^k + a^k \quad (7)$$

Hence, deviation from patient mix constraint for each category  $k$  is  $|H_k - D_k|$ . Then, a reward function ( $r(s, a)$ ) for each state and action can be constructed as follows:

$$r(s, a) = \sum_{k \in K} a^k n_k f_k - \sum_{k \in K} (w_k |H_k - D_k|) \quad a \in A_s \quad (8)$$

where  $f_k$  is the number of fractions required by patient type  $k$  per day and  $n_k$  is the consecutive number of days a patient type  $k$  needs to receive treatment. Therefore, the first term in the reward function is the total number of fractions obtained by starting  $a^k$  patients from each patient type  $k \in K$ . The second term penalizes the deviation of each type  $k$  from the desired patient mix level by  $w_k$ . It is important to note that the deviation is measured in terms of number of patients, but the first term is in terms of fractions. Finally, it is clear to see that the objective seeks to maximize the total reward.

Let  $v(s)$  be the total expected reward obtained for state  $s$ . Assuming that the state and action spaces are finite and the horizon length is infinite, a value iteration algorithm finds a stationary  $\epsilon$ -optimal policy and approximation to value function (9) for each state  $s$  with a discount factor of  $\beta$ .

$$v(s) = \max_{a \in A_s} \{r(s, a) + \beta \sum_{j \in S} p_{j|s,a} v(j)\} \quad (9)$$

We note that  $p_{j|s,a}$  is the transition probability from state  $s$  to  $j$  under action  $a$ . The stochastic dynamic of the problem, patient arrivals, can be modeled as Poisson arrivals. Hence, let  $N(k)$  be the number of patient types  $k$  request treatment on a specific day in the proton therapy facility. Then,  $N(k) \sim \text{Poisson}(\alpha_k)$ . Assuming that the arrivals are independent from each other,

Day 1			
	Time Slot 1	Time Slot 2	Time Slot 3
3 Gantries	≤ 3	≤ 3	≤ 3

Fig. 2: Modified block appointment system for identical gantries

we can estimate the probability of patient arrivals as follows:

$$\begin{aligned}
 P(N(1) = x_1, N(2) = x_2, \dots, N(|K|) = x_{|K|}) \\
 = \frac{e^{-\alpha_1} \alpha_1^{x_1}}{x_1!} \frac{e^{-\alpha_2} \alpha_2^{x_2}}{x_2!}, \dots, \frac{e^{-\alpha_{|K|}} \alpha_{|K|}^{x_{|K|}}}{x_{|K|}!}
 \end{aligned} \tag{10}$$

where  $x_k$  as the number of patients of type  $k$  requesting treatment.

#### A Numerical Example

This section demonstrates the performance of the MDP model over a numerical example in terms of best patient admission policy and convergence graphs for each state. Moreover, since the best patient admission policies may not always be optimal due to some other operational restrictions, second and third best actions are also reported. Before explaining these findings, the transition probability matrix, action and state space are illustrated in Figure 3 and 4. For this illustration the total patient capacity ( $Q$ ) is equal to 4.

The reason behind the limited and decreasing order action space as the number of patients in the facility increases is due to the assumption that discharges are performed after accepting new patients. When the capacity (the absorbing “FULL” state) is reached, we assume that the MDP model is terminated. First column in Figure 3 lists all possible states for this problem. Note that the first number in each state represents the number of patients in the facility from the first category and the second number stands for the number of patients in the facility from the second category. Actions are the possible combinations of accepting patients to the facility from these two types. The last action of each state in Figure 3 groups all combinations that might lead to the full state under a single action. Similarly, the probability of reaching the full state is calculated by subtracting the probabilities of all other arrival combinations from one.

Figure 4 demonstrates the relationships between the new arrivals and actions in more detail. The transition probabilities only depend on the total patients in the

facility (current and admitted by actions) prior to receipt of new arrivals. They are the same for any  $s$  and  $a$  which have the same value for  $s+a$  and are functions of  $s+a$  only. Baseline input parameters are seen in Table 1. For the sake of simplicity, the desired mix percentages ( $d_k$ ) and the number of treatment sessions ( $f_k$ ) are assumed to be identical for the two patient types. The number of consecutive days required for type  $k$  over the planning period ( $n_k$ ) and number of daily fractions required for type  $k$  ( $f_k$ ) are the actual numbers of the two categories obtained from the University of Florida Proton Therapy Institute (UFPTI). Arrival rates ( $\alpha_k$ ) and penalty terms ( $w_k$ ) are selected arbitrarily.

Figure 5 demonstrates the convergence of value function for each state obtained by solving the optimality equations 9 with the input parameters in Table 1 via a value iteration algorithm. This figure depicts that the value function of the state (0,0) dominates the value function of all other states. This is because, the fewer the number of patients in the facility, the more spots that can be filled with new patients which ultimately leads to a larger total expected fractions. Transitioning to the full state is the best action for the last five states as seen in Figure 5. Acceptance preferences (best, second and third best actions) change as the number of accepted patients directly impacts the actual and desired total mix ratios. This is illustrated for each state in Table 2. For instance, the best action is to accept two and one patient from the first and second category, respectively, for the state (0,0). On the other hand, accepting one patient from the first category becomes more affordable and the best action as the penalty terms decrease for the state (0,2). Similar conclusions can be made for each state.

### 3.2 An Aggregate MDP Model

The MDP model in the previous section provides strategic level optimal patient admission policies when the time horizon is assumed to be in steady state. Despite the useful practical insights it provides for the decision makers, it becomes intractable as the problem parameters increase. The state space of the MDP model grows exponentially in the order of  $Q$  and  $|K|$  (see Sec-

States	Actions
(0,0)	[(0,0), (0,1), (1,0), (0,2), (1,1), (2,0), (0,3), (1,2), (2,1), (3,0), (>4)]
(0,1)	[(0,0), (0,1), (1,0), (0,2), (1,1), (2,0), (>3)]
(1,0)	[(0,0), (0,1), (1,0), (0,2), (1,1), (2,0), (>3)]
(0,2)	[(0,0), (0,1), (1,0), (>2)]
(1,1)	[(0,0), (0,1), (1,0), (>2)]
(2,0)	[(0,0), (0,1), (1,0), (>2)]
(0,3)	[(0,0), (>1)]
(1,2)	[(0,0), (>1)]
(2,1)	[(0,0), (>1)]
(3,0)	[(0,0), (>1)]
Full	[(>0)]

Fig. 3: State space and action space when  $Q = 4$

		Probability Transition Matrix (p(j s,a))										
		j										
		(0,0)	(0,1)	(1,0)	(0,2)	(1,1)	(2,0)	(0,3)	(1,2)	(2,1)	(3,0)	Full
(s+a)	(0,0)	0.41	0.16	0.2	0.03	0.08	0.05	0	0.02	0.02	0.01	0.01
	(0,1)	0	0.41	0	0.16	0.2	0	0.03	0.08	0.05	0	0.06
	(1,0)	0	0	0.41	0	0.16	0.2	0	0.03	0.08	0.05	0.06
	(0,2)	0	0	0	0.41	0	0	0.16	0.2	0	0	0.23
	(1,1)	0	0	0	0	0.41	0	0	0.16	0.2	0	0.23
	(2,0)	0	0	0	0	0	0.41	0	0	0.16	0.2	0.23
	(0,3)	0	0	0	0	0	0	0.41	0	0	0	0.59
	(1,2)	0	0	0	0	0	0	0	0.41	0	0	0.59
	(2,1)	0	0	0	0	0	0	0	0	0.41	0	0.59
	(3,0)	0	0	0	0	0	0	0	0	0	0.41	0.59
	Full	0	0	0	0	0	0	0	0	0	0	1

Fig. 4: Probability transition matrix when  $Q = 4$

Table 1: Input parameters

	$n_k$	$f_k$	$w_k$	$d_k$	$\alpha_k$
k=1	40	1	8	50%	0.5
k=2	30	1	6	50%	0.4

Table 2: Best, second best and third best actions

Action	State										
	(0,0)	(0,1)	(1,0)	(0,2)	(1,1)	(2,0)	(0,3)	(1,2)	(2,1)	(3,0)	Full
Best	(2,1)	(2,0)	(1,1)	(1,0)	(1,0)	(0,1)	>1	>1	>1	>1	>0
Second Best	(3,0)	(1,1)	(2,0)	>2	(0,1)	(1,0)	(0,0)	(0,0)	(0,0)	(0,0)	>0
Third Best	(1,2)	(1,0)	(0,2)	(0,1)	>2	>2	(0,0)	(0,0)	(0,0)	(0,0)	>0

tion 4). Thus, this section introduces a state aggregation technique that is proposed to overcome the exponen-

tially growing states and actions (*curse of dimensionality*) associated with the MDP formulation.



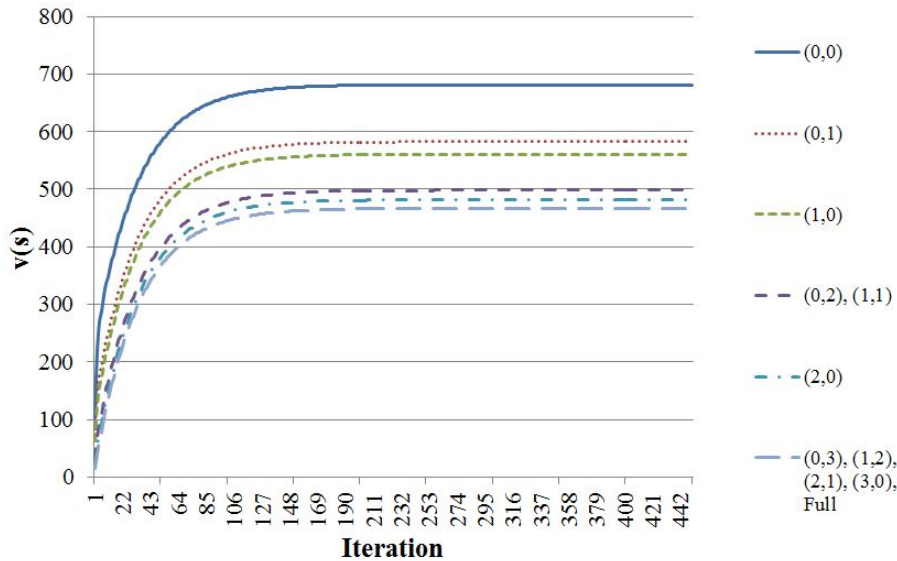


Fig. 5:  $v(s)$  convergence

In order to reduce the complexity associated with a large state space, researchers have focused on applying aggregation techniques in which the main idea is clustering original states into aggregate subgroups. These aggregate subgroups are treated as newly created states in the aggregate MDP. As a consequence, the reduced number of states will directly decrease the size of the probability transition matrix since the number of entries in this matrix is the square of the number of states. Using this approach, the original MDP can be represented by a potentially exponentially smaller *approximate* MDP. Since the ultimate approximate model preserves the Markovian property, any MDP algorithm can be used to solve the *aggregate* MDP.

State aggregation is referred to as state abstraction in the studies within the artificial intelligence domain. Dearden and Boutilier [15] provide a very concise discussion regarding the differences between state abstraction (or aggregation) and other compact MDP representation techniques. They propose an MDP model for the decision-theoretic planning (DTP) problem, another version of optimal stochastic control problem, and develop an abstract MDP model with exponentially fewer states which allow them to obtain approximate optimal solutions. The key idea behind their abstraction policy is to use action and reward structure to judge whether a variable (state) should be included in the aggregate MDP or not. The selection criteria of a state is based on the eligibility of capturing important behaviors of the original MDP (i.e., set of possible actions, amount of reward). Although there may be several states removed from the original MDP, the ultimate policy from their aggregate MDP remains executable for the orig-

inal MDP. In other words, the optimal policy for the aggregate problem is the approximate optimal solution for the original problem. This is a very important feature of a good practical state aggregation algorithm and one of the main objectives of this study is to generate approximate optimal patient admission policies.

Dean et al. [14] introduce a  $\epsilon$ -homogeneity technique that is employed to partition the very large state space of MDPs. They specifically study approximate optimal solutions for bounded MDPs in which upper and lower bounds on the transition probabilities and rewards are given as input parameters. Li et al. [25] propose a unified treatment of state abstraction for MDPs with large state space by analyzing five different state aggregation techniques and assess their functionalities in planning and learning problems. Moreover, an insightful case study is offered to measure the tradeoff between minimizing curse of dimensionality through different state abstraction and minimizing information loss. Other applications of state abstraction techniques in reinforcement learning can be found in Dietterich [16], Sutton et al. [41] and Andre and Russell [5].

Despite its popularity and dominance in artificial intelligence field, especially reinforcement learning application area, state aggregation has a very strong competitor in operation research field. Researchers in operations research have recently started exploring a different version of tackling curse of dimensionality through approximate dynamic programming (ADP) techniques. As opposed to clustering states in state aggregation, ADP focuses on stepping forward in time and selecting states with good approximate dynamic value functions to obtain approximately optimal policies [33]. This tech-

nique has been successfully implemented on numerous challenging problems including multidimensional knapsack [7], transportation & logistics [35], resource allocation [34] and capacity allocation [38, 37]. For further details on different aspects of ADP techniques, we refer the reader to Powell [32].

### 3.3 State Aggregation

Recall that  $S = \{s^1, s^2, \dots, s^{|K|}\}$  is the state space of the original MDP model where  $s^k$  is defined as the total number of type  $k$  patients within the facility. We let  $\{S_1, S_2, \dots, S_{|K|}, S_{|K|+1}, S_{|K|+2}\}$  partition  $S$  where  $\bigcup_{k=1}^{|K|+2} S_k = S$  and  $S_k \cap S_m = \emptyset$  if  $k \neq m$  and  $k, m \in \mathbb{K} = \{1, 2, \dots, |K|, |K| + 1, |K| + 2\}$ .  $\mathbb{K}$  is the set of states in the aggregate MDP.  $S_{|K|+2}$  represents the full state in the aggregate MDP. If  $s^1 = s^2 = \dots = s^{|K|}$  holds, that is the number of patients from all patients are the same, in an original MDP state, then it is assigned to the  $S_{|K|+1}$  aggregate state. We assign an original MDP state  $\{s^1, s^2, \dots, s^{|K|}\}$  to aggregate state  $S_k$  if  $s^k > s^m$  for all  $m \in K$  and  $m \neq k$  (patient type  $k$  has the unique largest number of patients). If there is more than one patient types which have the identical maximum amount of patients, then we randomly pick the aggregate state among those states. For example, if  $s^k = s^m > s^n$  for all  $n \in K$  and  $n \neq k \neq m$  in  $\{s^1, s^2, \dots, s^{|K|}\}$ , then we randomly assign this original state to either  $S_k$  or  $S_m$  with equal probability.

For each  $k \in \mathbb{K}$ , we define  $\mathbb{A}_k$  as  $\bigcup_{s \in S_k} A_s$ . That is,  $\mathbb{A}_k$  is the set of actions available from state  $k \in \mathbb{K}$  in the aggregate MDP. We use the *fixed-weight aggregation* technique that is proposed by Heyman and Sobel [21]. Accordingly, for each  $k, m \in \mathbb{K}$  and  $a \in \mathbb{A}_k$ , we let  $\gamma_{km}$  and  $\rho_k^a$  be a transition probability and single-state immediate reward in the aggregate MDP as defined in equations (11) and (12), respectively,

$$\gamma_{km} = \sum_{s \in S_k} \lambda_s^k \sum_{j \in S_m} p_{sj} \quad (11)$$

$$\rho_k^a = \sum_{s \in S_k} \lambda_s^k r(s, a) \quad (12)$$

where  $\lambda_s^k \geq 0$  and  $\sum_{s \in S_k} \lambda_s^k = 1$  for each  $k \in \mathbb{K}$  and  $s \in S_k$ . In (11),  $\sum_{j \in S_m} p_{sj}$  is the aggregate probability of transitioning from state  $s$  to aggregate state  $S_m$  which makes  $\gamma_{km}$  the weighted average of these transition probabilities where  $\lambda_s^k$  is the weights for  $s \in S_k$ . Intuitively,  $\lambda_s^k$  assesses the contribution of state  $s$  to the aggregate state  $S_k$  and can be any valid weighting function [25]. It is easy to see that  $\gamma_{km}$  are the transi-

tion probabilities in the aggregate MDP:

$$\begin{aligned} \sum_{m \in \mathbb{K}} \gamma_{km} &= \sum_{m \in \mathbb{K}} \sum_{s \in S_k} \lambda_s^k \sum_{j \in S_m} p_{sj} \\ &= \sum_{s \in S_k} \lambda_s^k \sum_{m \in \mathbb{K}} \sum_{j \in S_m} p_{sj} \\ &= \sum_{s \in S_k} \lambda_s^k \sum_{j \in S} p_{sj} = 1 \end{aligned}$$

Similarly,  $\rho_k^a$  is the weighted average of  $r(s, a)$  as defined in (8).

### 3.4 Fixed Policy Evaluation

Heyman and Sobel [21] prove that the output of the fixed aggregation transformation technique is indeed an MDP model with fewer states. This enables us to use the same value iteration algorithm to solve the aggregate model. The policy obtained by solving the aggregate model is executable for the original MDP model since the set of aggregate actions for each aggregate group is inherited from the original states ( $\mathbb{A}_k = \bigcup_{s \in S_k} A_s$ ). Of course, solutions for both original and aggregate MDP may not be identical. Thus, we need a fixed policy evaluation technique to assess the quality of the aggregate MDP policy with respect to the optimal policy. In other words, a disaggregating method is required to substitute the policy for the aggregate MDP model into the original MDP.

The optimal policy for infinite horizon MDP model is always stationary. That is, the optimal stationary policy ( $\pi^*$ ) for an infinite horizon discounted MDP model does not change with time and remains identical once it is obtained. Given a policy  $\pi$ , we define  $V_\pi(s)$  as the value of policy  $\pi$  at state  $s$ ,  $R_\pi(s)$  as the immediate reward and  $P_\pi(s, j)$  as the probability transition matrix. Let  $V_\pi$  and  $R_\pi$  be  $n$  dimensional column vectors and  $P$  be an  $n \times n$  matrix. Note that for any given policy  $\pi$ ,  $R_\pi$  and  $P$  will be known which will make the solution of the following set of linear equations (13) possible with respect to the only unknown variable  $V_\pi$ .

$$\begin{aligned} V_\pi &= R_\pi + \beta P_\pi V_\pi \\ R_\pi &= (I - \beta P)V_\pi \\ V_\pi &= (I - \beta P)^{-1} R_\pi \end{aligned} \quad (13)$$

After solving the aggregate MDP model by the value iteration algorithm, we obtain the approximate optimal policy  $\bar{\pi}$ . We create another feasible policy  $\delta$  for the original MDP model based on  $\bar{\pi}$  such that

$$\delta(s) = a \text{ if } s \in S_k \text{ and } \bar{\pi}(k) = a \forall k \in \mathbb{K}. \quad (14)$$

The policy disaggregation (14) suggests using the same optimal action for the aggregate subgroup  $S_k$  for

all the original states assigned to it ( $s \in S_k$ ). The quality of the aggregate policy  $\delta$  and optimal policy  $\pi^*$  is measured by solving the set of linear equations in (13).

## 4 Computational Results

### 4.1 Problem Parameters and Experimental Design

Several different problem instances are used to test the tractability and efficiency of the aggregate MDP model. The problem parameters in Table 3 except for daily arrival rates ( $\alpha_k$ ) were provided by the UFPTI and they include patient mix levels ( $d_k$ ), number of consecutive treatment days ( $n_k$ ) and required daily treatment sessions ( $f_k$ ) for ten patient categories. The arrival rates were calculated by using the expected number of new incidence of each patient category provided by the American Cancer Society [4]. These values are first adjusted based on the population of Florida and then normalized to fit into  $[0, 1]$ . Finally, the arrival rate of each patient is obtained by multiplying the normalized value by four to make the total arrival rate of the patients equal to the minimum number of time slots available in the facility.

Although there are multiple treatment regimens for some type of patients, receiving treatments over a number of consecutive days is the common property of the patient groups in the problem instances of this study. Men [26] and Men et al. [27] also report proton therapy treatment problem instances with patients who require treatment sessions over consecutive days.

Different combinations of the number of patient categories ( $|K| = 5, 6, \dots, 10$ ) and available time slots ( $Q = 4, 5, 6$ ) are used to vary the size of the state space. Therefore, we group multiple patient types as shown in Table 4-8 to meet different number of patient categories in problem instances with  $|K| = 9, 8, \dots, 5$ , respectively. In addition, three different penalty terms are adopted as follows:

1.  $w_k = d_k \times 20 \quad \forall k \in K$
2.  $w_k = \frac{n_k \times f_k}{4} \quad \forall k \in K$
3.  $w_k = n_k \times f_k \times d_k \quad \forall k \in K$

The first penalty type is used to represent the impact of penalizing the deviations from the desired patient mix levels in proportional to the desired percentages whereas the total number of fractions (treatment sessions) during the entire treatment period is adopted as the unit of the second penalty type. The influence of these two factors is combined in the third penalty type where each unit of deviation is penalized by the weighted number of treatment sessions. Lastly, state

aggregation weights are calculated as in equation (15).

$$\lambda_s^k = \frac{1}{|S_k|} \quad \forall k \in \mathbb{K}, s \in S_k \quad (15)$$

### 4.2 MDP vs. Aggregate MDP Model

The patient admission planning problem formulations in Section 3 are modeled in JAVA programming language. We assess the quality of the aggregate policy ( $\delta$ ) and optimal policy ( $\pi^*$ ) as described in Section 3.4. Recall that these policies provide the best action for each state and the total expected rewards obtained for each state are calculated by solving (13). In order to compare the values and best actions obtained by the MDP and aggregate MDP models, we define two performance indicators: (i) average percent difference (APD) and (ii) matched action percentage (MAP).

$$\text{APD} = 100 \times \frac{\sum_{s \in S} \left( \frac{|V_{\pi^*}(s) - V_{\delta}(s)|}{\max\{V_{\pi^*}(s) - V_{\delta}(s)\}} \right)}{|S|} \quad (16)$$

$$\text{MAP} = 100 \times \frac{\text{Number of same actions across all states in } S}{\text{Total number of states } (|S|)} \quad (17)$$

APD demonstrates how much, on average, the total expected reward values obtained by the MDP and aggregate MDP model deviate from each other across all states, whereas MAP illustrates the average precision of the aggregate MDP model in terms of locating the same actions with the MDP model across all states. These two measures for the instances with penalty type 1, 2 and 3 are illustrated in Figures 6-8, Figures 9-11 and Figures 12-14, respectively. When the penalty type 2 is used, that is each unit of deviation from the desired patient mix level for category  $k$  is penalized by  $\frac{n_k f_k}{4}$ , APD and MAP levels are consistently decreasing and increasing, respectively, as the models are exposed to larger  $|K|$  values. This leads us to conclude that behaviors of the MDP and aggregate MDP models become similar as  $|K|$ , therefore, the number of states increases when the penalty 2 is used. We can see the same pattern in APD and MAP values when each unit deviation from the desired mix level for patient type  $k$  is charged with  $n_k f_k d_k$  (penalty type 3). The only instance that does not comply with this trend under penalty type 3 is when  $|K| = 6$  and  $Q = 6$ . Even though the APD and MAP values improve with larger  $|K|$  when penalty type 1 is in use and  $Q = 4$ , we cannot observe the same

Table 3: (Aggregate) MDP problem parameters when  $|K| = 10$ 

	Patient Category ( $k \in K$ )									
	1	2	3	4	5	6	7	8	9	10
$d_k$	65%	15%	7%	3%	3%	2%	1%	1%	2%	1%
$n_k$	40	40	31	31	30	30	30	30	42	12
$f_k$	1	1	2	2	1	1	1	1	1	1
$\alpha_k$	1.65	0.38	0.92	0.19	0.12	0.08	0.15	0.15	0.19	0.15

Table 4: (Aggregate) MDP problem parameters when  $|K| = 9$ 

	Patient Category ( $k \in K$ )								
	1	2	3 & 10	4	5	6	7	8	9
$d_k$	65%	15%	8%	3%	3%	2%	1%	1%	2%
$n_k$	40	40	31	31	30	30	30	30	42
$f_k$	1	1	2	2	1	1	1	1	1
$\alpha_k$	1.65	0.38	1.07	0.19	0.12	0.08	0.15	0.15	0.19

Table 5: (Aggregate) MDP problem parameters when  $|K| = 8$ 

	Patient Category ( $k \in K$ )							
	1	2	3 & 10	4	5	6	7 & 8	9
$d_k$	65%	15%	8%	3%	3%	2%	2%	2%
$n_k$	40	40	31	31	30	30	30	42
$f_k$	1	1	2	2	1	1	1	1
$\alpha_k$	1.65	0.38	1.07	0.19	0.12	0.08	0.30	0.19

Table 6: (Aggregate) MDP problem parameters when  $|K| = 7$ 

	Patient Category ( $k \in K$ )						
	1	2	3 & 10	4	5 & 6	7 & 8	9
$d_k$	65%	15%	8%	3%	5%	2%	2%
$n_k$	40	40	31	31	30	30	42
$f_k$	1	1	2	2	1	1	1
$\alpha_k$	1.65	0.38	1.07	0.19	0.21	0.30	0.19

Table 7: (Aggregate) MDP problem parameters when  $|K| = 6$ 

	Patient Category ( $k \in K$ )					
	1	2	3 & 10	4	5,6,7 & 8	9
$d_k$	65%	15%	8%	3%	7%	2%
$n_k$	40	40	31	31	30	42
$f_k$	1	1	2	2	1	1
$\alpha_k$	1.65	0.38	1.07	0.19	0.51	0.19

Table 8: (Aggregate) MDP problem parameters when  $|K| = 5$ 

	Patient Category ( $k \in K$ )				
	1 & 2	3 & 10	4	5, 6, 7 & 8	9
$d_k$	80%	8%	3%	7%	2%
$n_k$	40	31	31	30	42
$f_k$	1	2	2	1	1
$a_k$	2.03	1.07	0.19	0.51	0.19

behavior under the same penalty type with  $Q = 5$  and  $Q = 6$ .

We observe that the majority of the contributions made to MAP levels come from the states in which the best action is to transition to the full state in both MDP models. Aggregate MDP model is very good at matching with these type of actions in general. It is not very competitive in anticipating the best action for the states that have several good quality actions available. We believe that part of the reason behind having better APD and MAP values with penalty types 2 and 3 is because they apply penalty levels directly proportional to the immediate reward coefficients ( $n_k f_k$ ) which in turn restricts the set of rewarding actions for states. This ultimately produces higher chance for the aggregate MDP to identify the best action taken by the MDP and approximate the total expected rewards.

Table 10 illustrates the optimal patient admission policies for three selected states ( $\bar{s}_1, \bar{s}_2$  and  $\bar{s}_3$ ) out of 3004 with  $|K| = 10$  and  $Q = 6$ . These states are selected arbitrarily in order to clarify how the proposed models can support managerial decision making processes. In the original MDP model, recall that  $\bar{s}_1$  corresponds to the empty state of the proton therapy facility when there is no patient from any category in the system. On the other hand,  $\bar{s}_2$  represents the case when only one category 5 and two category 7 patients are being treated in the facility. In the aggregate MDP model,  $\bar{s}_1, \bar{s}_2$  and  $\bar{s}_3$  are members of clustered states  $S_{|K|+1=11}, S_7$  and  $S_5$ , respectively. The necessary characteristics of these patient types can be seen in Table 3. Both the original and aggregate MDP models are able to report the top three best actions for each state based on the values obtained from the value iteration algorithm. For example, when the facility is operating in state  $\bar{s}_1$  under penalty type 1, admitting one patient from category 3 and four patients from category 4 is the best action obtained by the original MDP model. On the other hand, the aggregate MDP model reports this strategy as the second best admission plan and proposes admitting five patients only from category 4 as the best action. As a reminder, the top three actions reported under the aggregate MDP model corresponds to a cluster of original states and obtained by the policy decomposition step in equation (14). This method suggests using the same optimal action for all the original states assigned to the aggregate state ( $s \in S_k$ ). As an example, when the facility is operating in empty state ( $\bar{s}_1$ ) under penalty type 1, admitting five patients from category 4 is provided as a best action by the aggregate MDP model. This is also the best action for all other original states that are members of  $S_{11}$ . Even though the aggregate MDP model cannot achieve the same top

three best actions produced by the original MDP model for these three states, Figure 17, 18 and 19 demonstrate that the ultimate value functions generated by these two models are nearly identical especially for  $\bar{s}_2$  and  $\bar{s}_3$  under penalty type 1, 2 and 3, respectively. In other words, the best actions found via the aggregate MDP model provide good quality solutions.

We report problem setup times and value iteration algorithm solution times for both MDP and aggregate MDP models in Table 11. We ran these test problems on a Core 2 Quad 2.93 GHz, 4 GB RAM computer. We see that solving the MDP model takes significantly more time than solving the aggregate MDP model in all instances. This is a direct result of including a larger number of states ( $|S|$ ) in the MDP compared to the one ( $|K| + 2$ ) in the aggregate MDP. However, as the number of states increases, it takes dramatically more computation time to set up the aggregate MDP problem components. We also tested the computational performance of the two models on the problem instances with very large number of states as seen in Table 9. The aggregate MDP model is computationally more efficient than the original MDP model although the setup time for the larger problems with the aggregate MDP model is relatively high. These larger experiments are performed on a computer with two Intel six-core Xeon X5670 2.93 GHz processors and 24GB of memory.

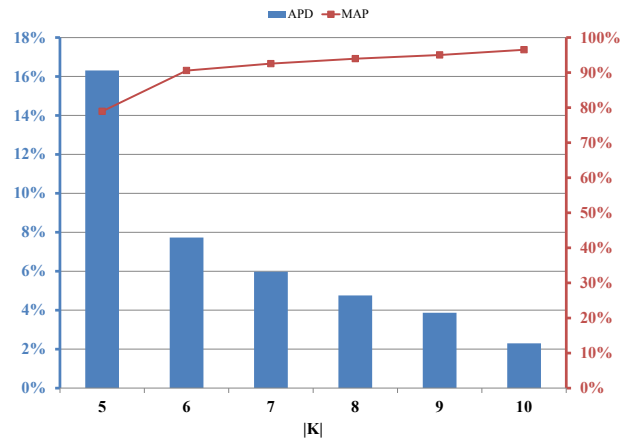


Fig. 6: APD and MAP with penalty type 1 and  $Q = 4$

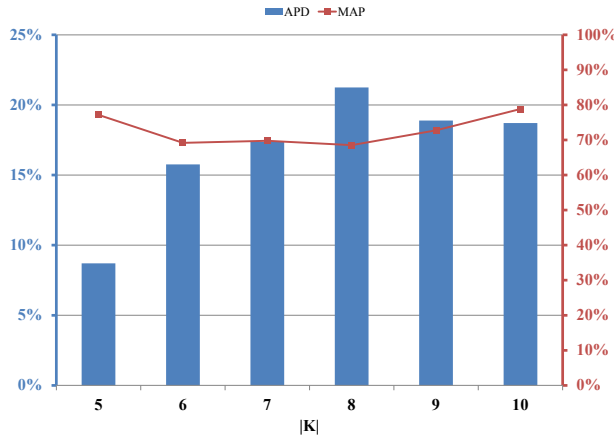


Fig. 7: APD and MAP with penalty type 1 and  $Q = 5$

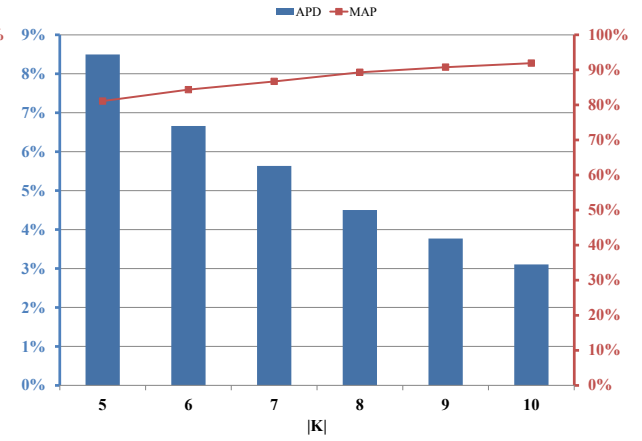


Fig. 10: APD and MAP with penalty type 2 and  $Q = 5$

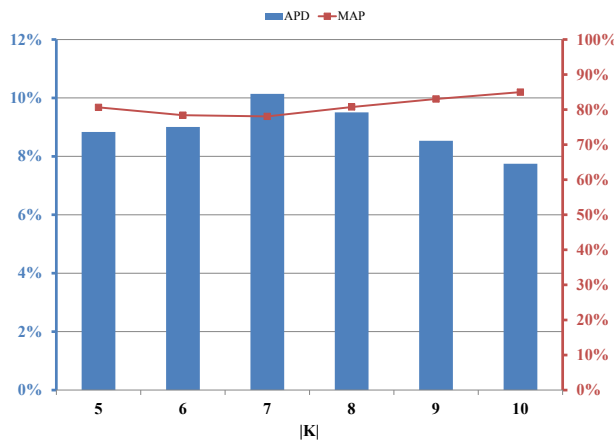


Fig. 8: APD and MAP with penalty type 1 and  $Q = 6$

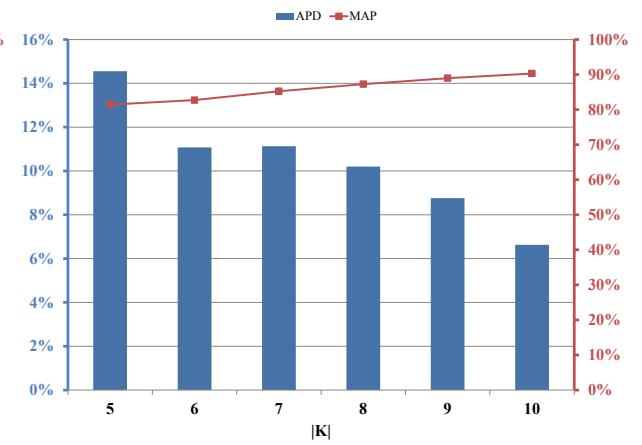


Fig. 11: APD and MAP with penalty type 2 and  $Q = 6$

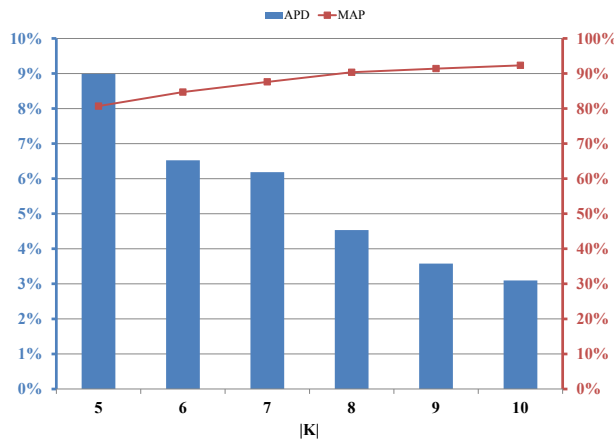


Fig. 9: APD and MAP with penalty type 2 and  $Q = 4$

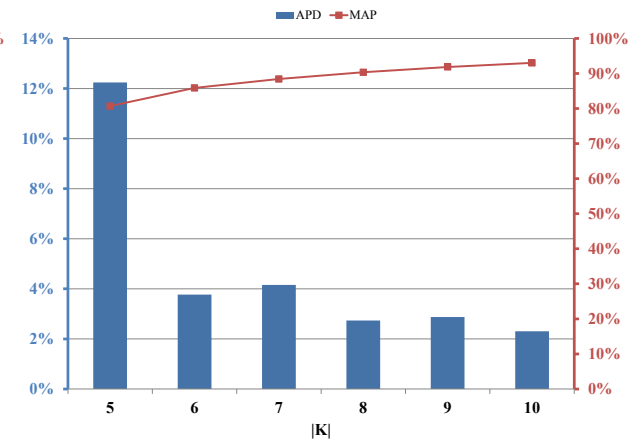


Fig. 12: APD and MAP with penalty type 3 and  $Q = 4$

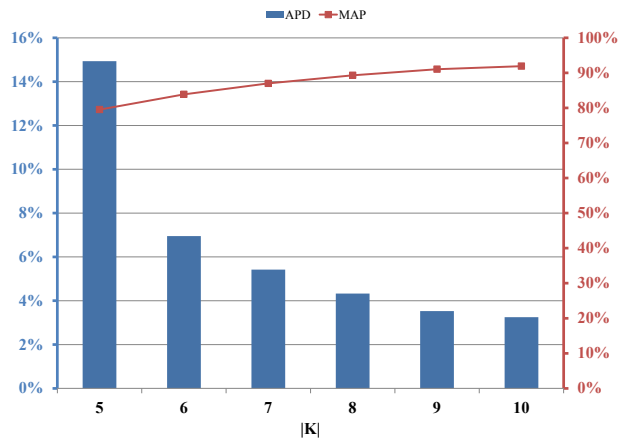


Fig. 13: APD and MAP with penalty type 3 and  $Q = 5$

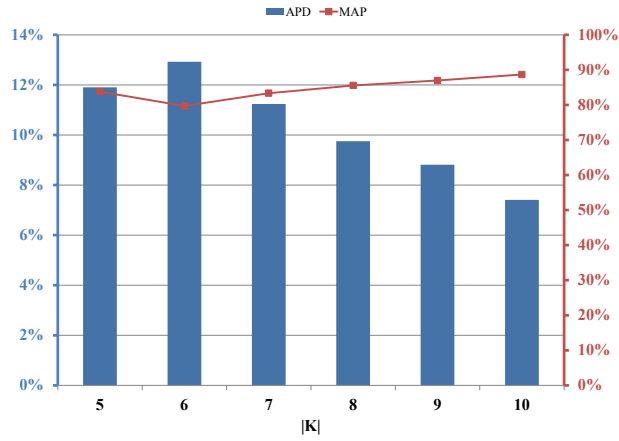


Fig. 14: APD and MAP with penalty type 3 and  $Q = 6$

Table 9: Performance of (aggregate) MDP models on large problem instances

$Q$	$ K $	Number of states	MDP		Aggregation	
			Setup (sec.)	Algorithm (sec.)	Setup (sec.)	Algorithm (sec.)
7	10	8009	56.930	19111.621	128.065	4.709
15	5	11629	-	-	3900.989	7.430
8	10	19449	-	-	865.857	12.243
17	5	20350	-	-	3900.989	7.430
20	5	42505	-	-	30607.492	16.096
10	10	92379	-	-	32868.907	65.671

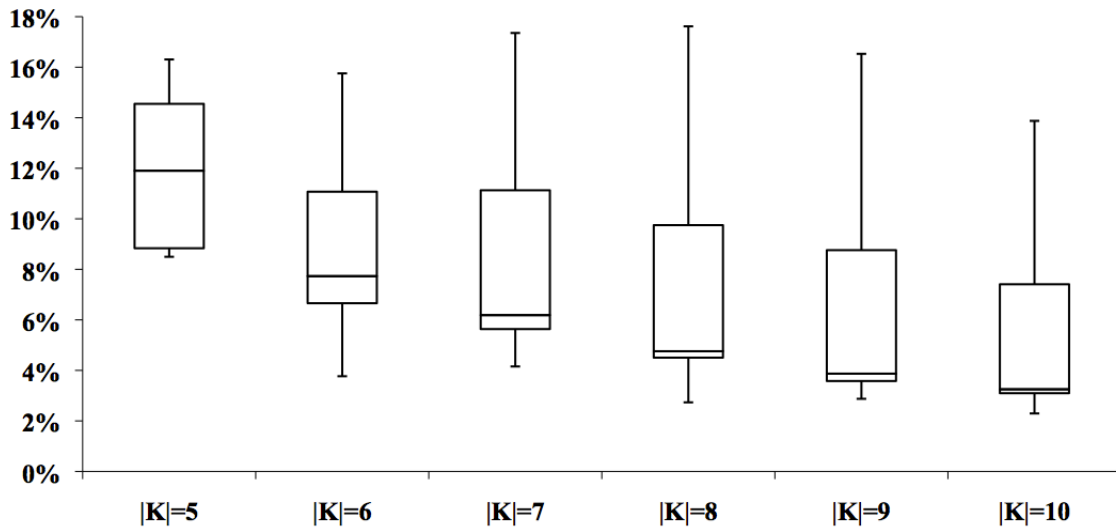


Fig. 15: APD Box Plot



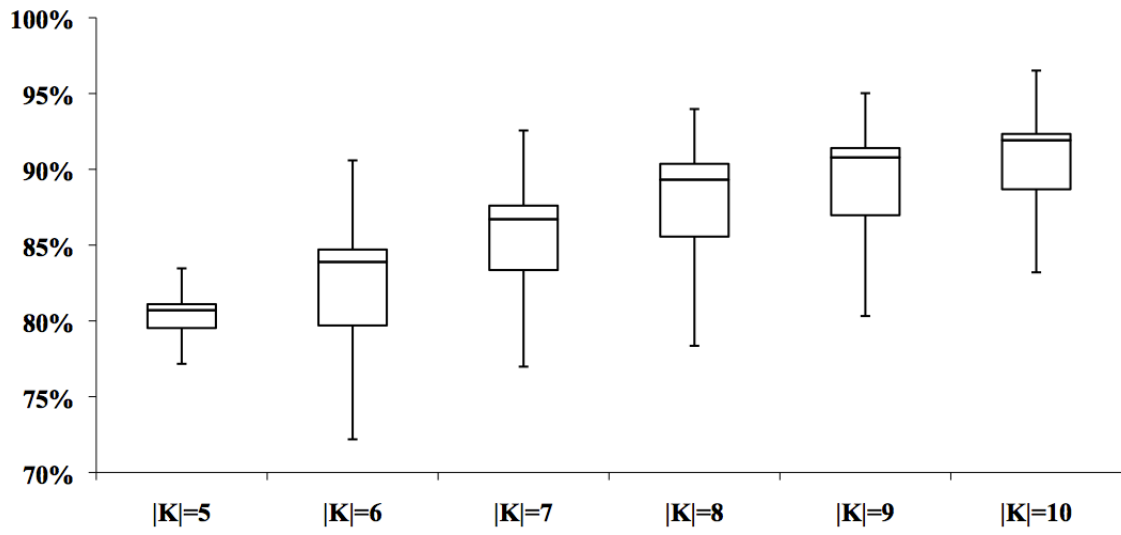


Fig. 16: MAP Box Plot

Table 10: Add caption

Penalty Type	Action	State					
		$\bar{s}_1=(0,0,0,0,0,0,0,0,0)$		$\bar{s}_2=(0,0,0,0,1,0,2,0,0)$		$\bar{s}_3=(0,0,0,0,2,0,0,1,1,0)$	
		MDP	Agg. MDP	MDP	Agg. MDP	MDP	Agg. MDP
1	Best	(0,0,1,4,0,0,0,0,0)	(0,0,0,5,0,0,0,0,0)	(0,0,1,1,0,0,0,0,0)	(0,0,0,2,0,0,0,0,0)	>2	(0,0,0,2,0,0,0,0,0)
	Second Best	(0,0,0,5,0,0,0,0,0)	(0,0,1,4,0,0,0,0,0)	(0,0,0,2,0,0,0,0,0)	(0,0,1,1,0,0,0,0,0)	(0,0,0,1,0,0,0,0,0)	(0,0,0,2,0,0,0,0,0)
	Third Best	(0,0,2,3,0,0,0,0,0)	(0,0,2,3,0,0,0,0,0)	(0,0,0,2,0,0,0,0,0)	(0,0,2,0,0,0,0,0,0)	(0,0,1,0,0,0,0,0,0)	(0,0,2,0,0,0,0,0,0)
2	Best	(3,0,1,1,0,0,0,0,0)	(0,0,0,5,0,0,0,0,0)	(0,0,1,1,0,0,0,0,0)	(0,0,0,2,0,0,0,0,0)	>2	(0,0,0,2,0,0,0,0,0)
	Second Best	(2,0,1,2,0,0,0,0,0)	(0,0,1,4,0,0,0,0,0)	(1,0,1,0,0,0,0,0,0)	(0,0,1,1,0,0,0,0,0)	(0,0,1,0,0,0,0,0,0)	(0,0,1,1,0,0,0,0,0)
	Third Best	(2,0,2,1,0,0,0,0,0)	(0,0,2,3,0,0,0,0,0)	(0,0,2,0,0,0,0,0,0)	(0,0,2,0,0,0,0,0,0)	(0,0,0,1,0,0,0,0,0)	(0,0,2,0,0,0,0,0,0)
3	Best	(3,0,1,1,0,0,0,0,0)	(0,0,0,5,0,0,0,0,0)	(2,0,0,0,0,0,0,0,0)	(0,0,0,2,0,0,0,0,0)	>2	(0,0,0,3,0,0,0,0,0)
	Second Best	(3,0,0,2,0,0,0,0,0)	(0,0,1,4,0,0,0,0,0)	(1,0,0,1,0,0,0,0,0)	(0,0,1,1,0,0,0,0,0)	(1,0,0,0,0,0,0,0,0)	(0,0,1,2,0,0,0,0,0)
	Third Best	(3,0,2,0,0,0,0,0,0)	(0,0,2,3,0,0,0,0,0)	(1,0,1,0,0,0,0,0,0)	(0,0,2,0,0,0,0,0,0)	(0,0,0,1,0,0,0,0,0)	(0,0,2,1,0,0,0,0,0)

## 5 Conclusions and Future Research Directions

The MDP model proposed in this study is powerful decision mechanisms in the case of an unexpected opening in the facility (i.e., no-shows, appointment cancellations, etc.). It aims to find the best patient admission decision with the maximum total expected reward while conforming to the patient mix and capacity restrictions. Numerical experiments demonstrate that it becomes intractable due to exponentially growing state space as problem parameters increase. Thus, approximate optimal patient admission policies are targeted via an aggregate MDP model developed by fixed-weight aggregation technique. Despite its ease of application, it is capable of providing very good executable approximate policies for the original MDP model which is also evidenced by small APD and large MAP values. Moreover, we observe that as the number of patient categories ( $|K|$ ) increases, the aggregate MDP model produces more realistic patient admission policies in terms of MAP and ADP values. This fact is highlighted in Figure 15 and 16 where the basic statistical properties of the aggregate MDP model in terms of APD and MAP are represented for the entire problem instances clustered by  $|K|$ . We also note that the worst case APD (MAP) values are 21% (69%), 14.1% (79%) and 14.4% (80%) for the penalty type 1, 2, and 3, respectively. Overall, the findings show that the aggregate MDP approach is a viable approach for proposing the desirable patient admission strategies. Thus, the proposed approach can be embedded in a decision support system in order to perform sensitivity analysis, or to address the changes in future plans in the case of any parameter changes (i.e., patient mix ratio updates). The final version of this study will be presented to the UFPTI in order to start clinical testing.

There are several future directions regarding this study. Note that the gantry capacity constraints are the only factor that limits the acceptance of new patients in the MDP model. One important extension of this work would be incorporating other operational restrictions such as sequencing of patients, anesthesia surveillance team availability and assessing the impacts of these on patient admission policies. For instance, in order to account for the treatments of patients who need other resources (i.e., technicians or nurses) during their treatment sessions, we need to differentiate the time slots based on the availability of such resources in gantries. This would dramatically increase the dimension of both the state and action spaces, which in turn ultimately impact the complexity of the problem. Another potential extension of this work would be accounting for the waiting patients for this treatment. Since this is another

layer of information that describes the state of the facility, it should be incorporated in the state space definition. Finally, a more competitive aggregation technique that can mimic the behaviors of the MDP model with several other operational constraints would be another possible research direction.

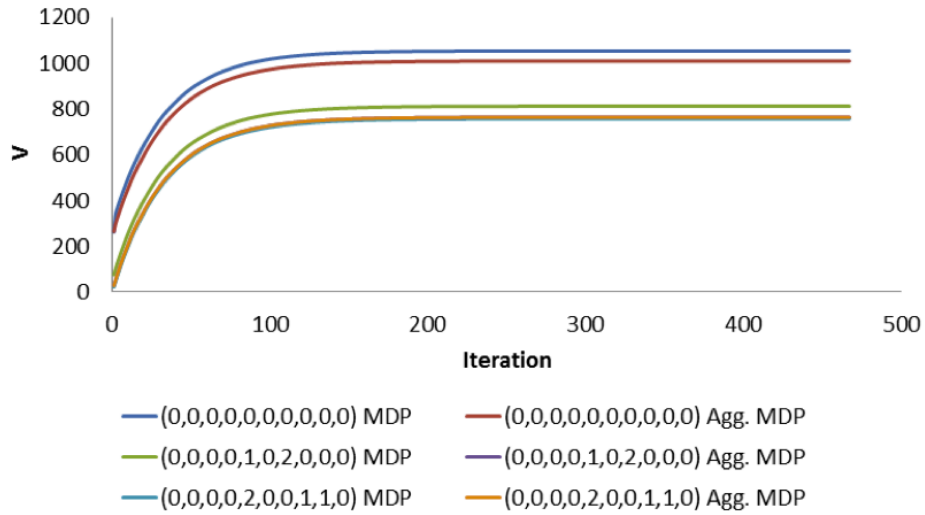


Fig. 17: Value function convergence graph for each selected state with penalty type 1,  $|K| = 10$  and  $Q = 6$

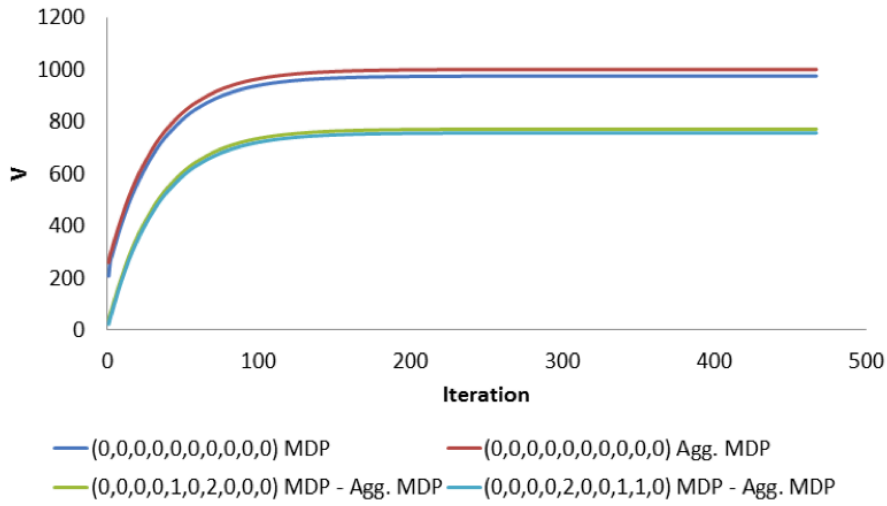


Fig. 18: Value function convergence graph for each selected state with penalty type 2,  $|K| = 10$  and  $Q = 6$

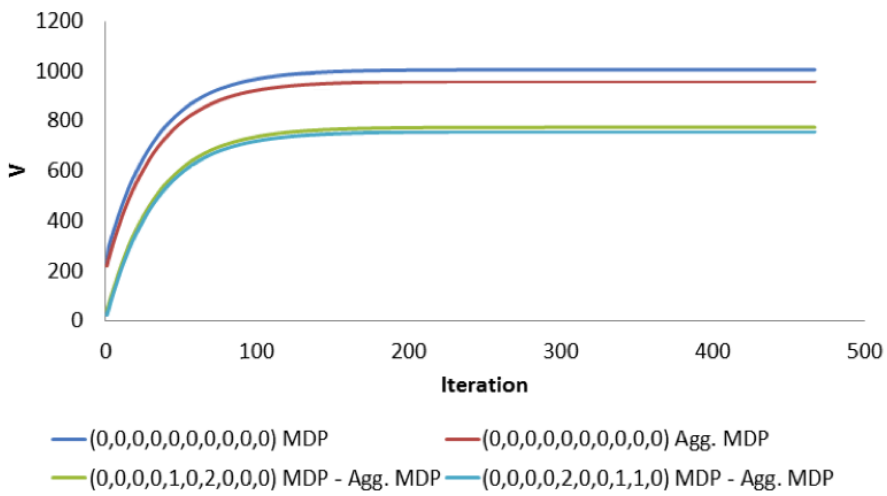


Fig. 19: Value function convergence graph for each selected state with penalty type 3,  $|K| = 10$  and  $Q = 6$

Table 11: Computational Performance of MDP and Aggregate MDP models

Penalty Type	MDP				Aggregation				Penalty Type	MDP				Aggregation			
	$Q$	$ K $	$ S $	Setup (sec.)	Algorithm (sec.)	Setup (sec.)	Algorithm (sec.)	$Q$		$ K $	$ S $	Setup (sec.)	Algorithm (sec.)	Setup (sec.)	Algorithm (sec.)		
1	4	5	57	0.026	1.899	0.097	0.282	1	4	8	166	0.175	6.800	0.403	0.482		
2	4	5	57	0.004	1.493	0.068	0.200	2	4	8	166	0.105	7.268	0.340	0.311		
3	4	5	57	0.003	1.423	0.046	0.143	3	4	8	166	0.041	7.174	0.269	0.330		
1	5	5	127	0.008	5.236	0.160	0.212	1	5	8	496	0.175	64.324	1.224	0.616		
2	5	5	127	0.012	5.273	0.144	0.182	2	5	8	496	0.181	64.245	1.292	0.621		
3	5	5	127	0.006	5.037	0.122	0.219	3	5	8	496	0.181	64.141	1.265	0.615		
1	6	5	253	0.023	22.049	0.277	0.288	1	6	8	1288	1.045	580.842	5.519	1.278		
2	6	5	253	0.022	22.465	0.314	0.243	2	6	8	1288	1.082	598.189	5.859	1.304		
3	6	5	253	0.022	22.593	0.305	0.268	3	6	8	1288	1.071	577.072	5.730	1.341		
1	4	6	85	0.048	3.140	0.167	0.247	1	4	9	221	0.271	11.172	0.683	0.475		
2	4	6	85	0.006	2.673	0.075	0.158	2	4	9	221	0.190	11.460	0.309	0.324		
3	4	6	85	0.004	2.486	0.089	0.163	3	4	9	221	0.115	11.620	0.357	0.377		
1	5	6	211	0.021	12.605	0.257	0.249	1	5	9	716	0.630	130.615	2.172	0.858		
2	5	6	211	0.019	12.511	0.247	0.303	2	5	9	716	0.610	129.702	2.199	0.842		
3	5	6	211	0.019	12.289	0.282	0.256	3	5	9	716	0.596	131.751	2.181	0.813		
1	6	6	463	0.091	69.813	0.952	0.455	1	6	9	2003	3.843	1427.650	11.937	2.154		
2	6	6	463	0.091	69.963	0.916	0.452	2	6	9	2003	3.758	1444.031	12.299	2.157		
3	6	6	463	0.089	69.728	0.955	0.419	3	6	9	2003	3.658	1446.783	11.780	2.141		
1	4	7	121	0.100	4.664	0.246	0.220	1	4	10	287	0.884	18.449	1.082	0.498		
2	4	7	121	0.140	4.249	0.184	0.250	2	4	10	287	0.386	18.473	1.060	0.423		
3	4	7	121	0.109	4.123	0.146	0.234	3	4	10	287	0.415	18.351	1.197	0.465		
1	5	7	331	0.060	26.009	0.535	0.362	1	5	10	1002	2.333	251.040	4.091	1.140		
2	5	7	331	0.057	25.929	0.568	0.385	2	5	10	1002	2.358	252.854	4.018	1.211		
3	5	7	331	0.058	25.789	0.610	0.426	3	5	10	1002	2.355	253.151	3.911	1.229		
1	6	7	793	0.321	207.925	2.287	0.736	1	6	10	3004	13.449	3272.205	26.170	3.473		
2	6	7	793	0.321	200.939	2.264	0.721	2	6	10	3004	13.191	3273.545	27.323	3.575		
3	6	7	793	0.321	200.400	2.321	0.715	3	6	10	3004	15.575	3214.341	33.887	3.608		

## References

1. Adan, I. and Vissers, J. (2002). Patient mix optimisation in hospital admission planning: a case study. International Journal of Operations & Production Management, 22(4):445–461.
2. Agnetis, A., Coppi, A., Corsini, M., Dellino, G., Meloni, C., and Pranzo, M. (2012). Long term evaluation of operating theater planning policies. Operations Research for Health Care, 1(4):95 – 104.
3. Agnetis, A., Coppi, A., Corsini, M., Dellino, G., Meloni, C., and Pranzo, M. (2014). A decomposition approach for the combined master surgical schedule and surgical case assignment problems. Health Care Management Science, 17(1):49–59.
4. American Cancer Society (2014). Cancer Facts and Figures 2014. <http://www.cancer.org/acs/groups/content/@research/documents/webcontent/acspc-042151.pdf>. [Online; accessed 30-June-2014].
5. Andre, D. and Russell, S. J. (2002). State abstraction for programmable reinforcement learning agents. In AAAI/IAAI, pages 119–125.
6. Averill, R. F., Muldoon, J. H., Vertrees, J. C., Goldfield, N. I., Mullin, R. L., Fineran, E. C., Zhang, M. Z., Steinbeck, B., and Grant, T. (1993). The evolution of casemix measurement using diagnosis related groups (drgs ). Wallingford 3M Health Information Systems.
7. Bertsimas, D. and Demir, R. (2002). An approximate dynamic programming approach to multidimensional knapsack problems. Management Science, 48(4):550–565.
8. Bowers, J. and Mould, G. (2005). Ambulatory care and orthopaedic capacity planning. Health Care Management Science, 8:41–47.
9. Cayirli, T., Veral, E., and Rosen, H. (2006). Designing appointment scheduling systems for ambulatory care services. Health Care Management Science, 9:47–58.
10. Ciavotta, M., Dellino, G., Meloni, C., and Pranzo, M. (2010). A rollout algorithmic approach for complex parallel machine scheduling in healthcare operations. In Operations research for patient: centered health care delivery: proceedings of the XXXVI International ORAHS Conference, volume 5, pages 316–324.
11. Ciavotta, M., Meloni, C., and Pranzo, M. (2009). Scheduling dispensing and counting in secondary pharmaceutical manufacturing. AIChE Journal, 55(5):1161–1170.
12. Conforti, D., Guerriero, F., and Guido, R. (2008). Optimization models for radiotherapy patient scheduling. 4OR, 6:263–278.
13. Conforti, D., Guerriero, F., and Guido, R. (2010). Non-block scheduling with priority for radiotherapy treatments. European Journal of Operation Research, 201:289–296.
14. Dean, T., Givan, R., and Leach, S. (1997). Model reduction techniques for computing approximately optimal solutions for markov decision processes. In Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence, pages 124–131. Morgan Kaufmann Publishers Inc.
15. Dearden, R. and Boutilier, C. (1997). Abstraction and approximate decision-theoretic planning. Artificial Intelligence, 89(1):219–283.
16. Dietterich, T. G. (2000). Hierarchical reinforcement learning with the maxq value function decomposition. Journal of Artificial Intelligence Research, 13(1):227–303.
17. Gedik, R. (2011). Evaluating the Capacity of a Proton Therapy Facility. University of Arkansas, Fayetteville.
18. Gedik, R. (2014). Large-Scale Solution Approaches for Healthcare and Supply Chain Scheduling. Phd thesis, University of Arkansas.
19. Gocgun, Y., Bresnahan, B. W., Ghate, A., and Gunn, M. L. (2011). A markov decision process approach to multi-category patient scheduling in a diagnostic facility. Artificial Intelligence in Medicine, 53(2):73 – 81.
20. Goitein, M. and Jermann, M. (2003). The relative costs of proton and x-ray radiation therapy. Clinical Oncology, 15:S37–S50.
21. Heyman, D. P. and Sobel, M. J. (2003). Stochastic Models in Operations Research: Stochastic Optimization, volume 2. Courier Dover Publications.
22. Kapadia, A. S., Vineberg, S. E., and Rossi, C. D. (1985). Predicting course of treatment in a rehabilitation hospital: a markovian model. Computers & OR, 12(5):459–469.
23. Kapamara, T., Sheibani, K., Haas, O., Reeves, C., and Petrovic, D. (2006). A review of scheduling problems in radiotherapy. Proceedings of the Eighteenth International Conference on Systems Engineering (ICSE2006), Coventry University, UK, pages 201–207.
24. Kolesar, P. (1970). A markovian model for hospital admission scheduling. Management Science, 16(6):pp. B384–B396.
25. Li, L., Walsh, T. J., and Littman, M. L. (2006). Towards a unified theory of state abstraction for mdps. In ISAIM.

26. Men, C. (2009). Optimization models for radiation therapy: treatment planning and patient scheduling. University of Florida.
27. Men, C., Salari, E., Romeijn, H. E., Klein, S. L., Li, Z., Mendenhall, N. P., and Palta, J. R. (2008). Report of the strategic model. Technical report, University of Florida.
28. Nunes, L. G. N., de Carvalho, S. V., and Rodrigues, R. d. C. M. (2009). Markov decision process applied to the control of hospital elective admissions. Artificial intelligence in medicine, 47(2):159–171.
29. Petrovic, D., Castro, E., Petrovic, S., and Kapamara, T. (2013). Radiotherapy scheduling. In Uyar, A. S., Ozcan, E., and Urquhart, N., editors, Automated Scheduling and Planning, volume 505 of Studies in Computational Intelligence, pages 155–189. Springer Berlin Heidelberg.
30. Petrovic, D., Morshed, M., and Petrovic, S. (2009). Genetic algorithm based scheduling of radiotherapy treatments for cancer patients. In Combi, C., Shahar, Y., and Abu-Hanna, A., editors, Artificial Intelligence in Medicine, volume 5651 of Lecture Notes in Computer Science, pages 101–105. Springer Berlin Heidelberg.
31. Pham, D. N. and Klinkert, A. (2008). Surgical case scheduling as a generalized job shop scheduling problem. European Journal of Operation Research, 185:1011–1025.
32. Powell, W. B. (2007). Approximate Dynamic Programming: Solving the curses of dimensionality. John Wiley & Sons.
33. Powell, W. B. (2009). What you should know about approximate dynamic programming. Naval Research Logistics (NRL), 56(3):239–249.
34. Powell, W. B., George, A., Bouzaiene-Ayari, B., and Simao, H. P. (2003). Approximate dynamic programming for high dimensional resource allocation problems. In Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on, volume 5, pages 2989–2994. IEEE.
35. Powell, W. B., Simao, H. P., and Bouzaiene-Ayari, B. (2012). Approximate dynamic programming in transportation and logistics: a unified framework. EURO Journal on Transportation and Logistics, 1(3):237–284.
36. Salari, E., Men, C., Li, Z., Mendenhall, N., Palta, J., and Romeijn, H. (2009). Su-ff-t-467: A new patient scheduling approach for proton therapy treatment delivery. Medical Physics, 36(6):2630–2630.
37. Sauré, A., Patrick, J., Tyldesley, S., and Puterman, M. L. (2012). Dynamic multi-appointment patient scheduling for radiation therapy. European Journal of Operational Research, 223(2):573–584.
38. Schütz, H.-J. and Kolisch, R. (2012). Approximate dynamic programming for capacity allocation in the service industry. European Journal of Operational Research, 218(1):239–250.
39. Shelley, W., Brundage, M., Hayter, C., Paszat, L., Zhou, S., and Mackillop, W. (2000). A shorter fractionation schedule for postlumpectomy breast cancer patients. International Journal of Radiation Oncology, Biology, Physics, 47:1219–1228.
40. Sickinger, S. and Kolisch, R. (2009). The performance of a generalized baileywelch rule for outpatient appointment scheduling under inpatient and emergency demand. Health Care Management Science, 12(4):408–419.
41. Sutton, R. S., Precup, D., and Singh, S. (1999). Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. Artificial intelligence, 112(1):181–211.
42. Yamada, Y., Ackerman, I., Franssen, E., Mackenzie, R. G., and Thomas, G. (1999). Does the dose fractionation schedule influence local control of adjuvant radiotherapy for early stage breast cancer. International Journal of Radiation Oncology, Biology, Physics, 44:99–104.