A simulation-based neighbourhood search algorithm to schedule multi-category patients at a multi-facility health care diagnostic centre

Varun Jain and Usha Mohan

Department of Management Studies, Indian Institute of Technology Madras, Chennai, India

ABSTRACT

A key operational decision faced by a multi-facility health care diagnostic centre serving different patient categories (for example: Health Check-up Patient (*HCP*), Out-Patient (*OP*), Emergency Patient (*EP*), or In-Patient) is whom to serve next at a particular facility. In this paper, we model random arrival of these patients belonging to different categories and priorities at multiple diagnostic facilities over a finite planning horizon. We formulate a mathematical model for sequential decision-making under uncertainty using Markov Decision Process (MDP) with the objective of maximising net revenue and use dynamic programming (DP) to solve it. To address dimensionality and scalability issue of MDP, we provide a decentralised MDP (D_MDP) formulation. We develop simulation-based neighbourhood search algorithm to improve DP solution for D_MDP. We compare these solutions with three other rule-based heuristics using simulation.

ARTICLE HISTORY

Received 21 April 2017
Revised 22 August 2017
Accepted 18 October 2017

KEYWORDS

Health care; scheduling;
Markov decision process;
heuristics; simulation

1. Introduction

For many years, the objective in health care operations was to maximise the number of patients served and minimise patient waiting time. However in recent times, with increasing competition and scarcity of resources, economic parameters such as maximising revenue has become a major objective. Hence, there has been a distinct paradigm shift from health care delivery to preventive health check-up with renewed objectives. Health check-up plans have emerged as a primary source of revenue in many diagnostic centres.

The motivation for this research comes from the recent trend observed in the health care delivery system (henceforth, we use centre/diagnostic centre/system interchangeably to mean a multi-facility diagnostic centre) in India which we believe applies to many other emerging economies also. The need for preventive health care has forced diagnostic centres to respond by offering health check-up packages. Many multi-facility health care diagnostic centres in the recent past have come up with different packages for a preventive health check-up, such as Platinum health check-up, Executive health check-up, etc. The number of facilities (tests) needed by each category is usually different. Depending on the type of package, a diagnostic centre forms subsets of various facilities. For example, health check-up patients (*HCPs*) may require fasting blood sugar (F1), ultrasound scan (F2), electrocardiogram (ECG) (F3), treadmill Test (TMT) (F4), and X-Ray (F5). The sequence of services in this example is F1, F2, F3, F4, F5 and are usually fixed. The *OP* and *EP* category of pa-

tients usually come for only one test and not a sequence of tests. An important operational issue faced by such diagnostic centres that cater to multiple categories of patients (patients with different priorities and patients who need either all services or a subset of services) is scheduling of the patients while managing their scarce resources.

The main motivation for the work is based on our observations at two diagnostics centres: (1) A leading private hospital in Nagpur, a city in Central India. The hospital is a 118-bedded hospital that has various health check-up packages and caters to different categories of patients. The in-house diagnostic facilities cater to inpatients, outpatients as well as *HCPs*. (2) A multi-facility health check-up centre in Chennai, a city from Southern India. The diagnostic centre also offers different health check-up packages along with serving other categories of patients.

One of the significant challenges faced in this scheduling arises due to the limitations forced by the patient's category as well as limitations on medical resources such as availability of facility, nurse, supporting staff, and physician. In most diagnostic centres, this problem is addressed by specialised personnel using *ad-hoc* trial and error methods and rules of thumb (Gocgun, Bresnahan, Ghate, & Gunn, 2011). The resultant schedules very often lead to long waiting times for patients and thus affects the revenue generated. The complexity of the problem necessitates the need for good mathematical models to address the problem and aid in decision-making.

When there is uncertainty in the system, the Markov Decision Process (MDP) framework is used to model the system dynamics. In this paper, we look at a stochastic environment and capture the system dynamics with MDP. Dynamic programming (DP) is one of the traditional approaches for solving the appointment scheduling MDP problem. However, it suffers from the curse of dimensionality as the state space grows exponentially. Hence, researchers have developed useful heuristic tools. To overcome the scalability and dimensionality problem, we factor the MDP model to form a decentralised MDP (D_MDP), and we develop a simulation-based neighbourhood search algorithm (SBNSA) to solve it. One of the most important aspects of the decentralised approach proposed in the paper is the “linkage” between the L single-facility MDP models and that these L facilities are not completely independent of each other as far as the state transition is concerned (the *HCP* queues are linked in a sequential manner). Another key contribution of this paper is to present an efficient heuristic to schedule patients in a multi-facility health care diagnostic centre.

2. Literature review

Hulshof, Kortbeek, Boucherie, Hans, and Bakker (2012) provide the structured taxonomy to aid decision-making for both the OR/MS (Operations Research/Management Sciences) and health care community. The majority of literature in patient scheduling has focused on appointment scheduling with varying objectives. Rais and Viana (2010) survey many applications of Operations Research in the field of health care. In particular, they review critical optimisation issues such as service planning, resource scheduling, logistics, medical therapeutics, disease diagnosis, and preventive care along with solution techniques to solve the optimisation problems. According to their survey, the most commonly used tool in this area is simulation. Gupta and Denton (2008) review the literature on health care appointment scheduling. They discuss three environments which require appointment scheduling: primary care, specialty care, and elective surgery. They further explain the four complexities of the health care system: (1) arrival process, (2) service process, (3) patient and provider preferences and (4) incentives and performance. Based on these factors, they categorise the existing literature and pose a few open challenges.

The problem of scheduling health check up patients (referred as health examination patients) has been tackled using binary integer programming (BIP) techniques. The BIP models are largely deterministic and cannot capture the uncertainty in patient arrival. Chern, Chien, and Chen (2008) propose a health examination scheduling algorithm (HESA) to handle more patients and examination procedures than possible with the traditional

BIP model. Their objectives are to minimise patient and doctor waiting time taking sequencing of examination procedures and continuity into consideration. They tested this heuristic on the real time data of two hospitals and showed that HESA performs better in terms of computation time and ability to handle larger size problems than BIP technique.

Vlah, Lukač, and Pacheco (2008) study a multi-priority resident patient scheduling at multiple centres for a given day with known demand. They develop an elaborate BIP model for addressing equipment scheduling and physician allocation simultaneously, with the objective of maximising throughput of the highest priority patient. Due to the issue of scalability in BIP model, they propose a reduced variable neighbourhood search (VNS) metaheuristic. They justify the improved performance of VNS over BIP model on real-time and experimental data.

In recent times, one of the most used methods in patient scheduling when uncertainty exists is to model resource utilisation as a MDP. The earliest paper that used MDP to schedule diagnostic resources is by Green, Savin, and Wang (2006). They extend previous studies by considering multiple classes of patients with a probabilistic arrival in a finite planning horizon with the objective of maximising revenue. They develop a MDP model and study the structural properties of the policies along with different appointment policies to build faster heuristics rules for efficient utilisation of a diagnostic facility. They also provide guidelines on how different policies should be selected under various scenarios. Kolisch and Sickinger (2008) extend this work to include two parallel CT-scanners (Computed Tomography scanners). They generalised the problem by allowing more than one scheduled outpatient in one time period. They observed that health care systems are more willing to accept simple decision-based rules as compared to complex computer-based decision rules. Hence, they studied the effect of different simple decision rules for various appointment schedules and scenarios.

Patrick, Puterman, and Queyranne (2008) attempt to attain wait-time targets while minimising penalty cost for patients waiting longer than the wait-time targets. They develop a dynamic scheduling MDP model for a diagnostic resource with patients with multiple priorities. They use approximate dynamic programming (ADP) for solving the MDP model as the dimension of the problem was too large for a direct solution. They apply the model and solution methodology on real data and report the quality of policy through simulation.

Sauré, Patrick, Tyldesley, and Puterman (2012) extend the work of Patrick et al. (2008) by including treatments that span multiple days, multiple appointment requests, stochastic service time, and letting portions of appointment be served using overtime. They formulate

it as a discounted infinite-horizon MDP for identifying good policies for allocating valuable resources in a cost effective manner and with a reduction in waiting time. One of the methods proposed in the literature to mitigate no-shows and hence avoid wasted capacity, is “Open Access (OA)”. [Patrick \(2012\)](#) refer to the essence of “Open Access (OA)” as “do today’s demand today”. They provide a MDP approach for the dynamic clinic scheduling problem when no-shows depend on wait times. With the help of a simulation study, they show that allowing wait-time-dependent no-show rates and short booking windows along with overbooking performs better than OA scheduling policy.

[Vermeulen, Bohte, Elkhuisen, Lameris, Bakker, and Poutre \(2009\)](#) develop an adaptive approach to automating the scheduling of resource calendars. They classify patients into different groups as per medical constraint and urgency to allocate them to different time slots. Their performance metric was to achieve a specific service level. They also tried to optimise the utilisation of available resources by shifting resource as per requirements of the patient group. [Schütz and Kolisch \(2012\)](#) consider the problem of booking multiple categories of patients for multiple types of services at the same facility in advance. They model it using a continuous-time MDP with the objective of maximising profits. They extend the previous work by considering stochastic service time and unpunctual patients along with no-shows, cancellation, and overbooking. They solve it using simulation-based ADP with discrete event simulation for service time.

[Gocgun et al. \(2011\)](#) develop a mathematical model for multi-category patient scheduling for CT in a diagnostic facility using MDP to study economic and operational trade-offs. They model the problem as a finite-horizon MDP with expected net CT revenue as the objective. They solve the model using DP to get the optimal policy. They propose five heuristics and run experiments using real-time data from an urban hospital. While comparing results of the MDP policies with the heuristics, they also consider patient-throughput and service quality metrics. Their analysis shows that the optimal policy is as competitive as other heuristics and can be used in future for more real-time scheduling. However, they model only one facility at a centre.

Our work is closest in spirit to the work by [Kolisch and Sickinger \(2008\)](#) and [Gocgun et al. \(2011\)](#). We extend their work to include multiple facilities at a centre; where each facility provides a different unique service. Similar to their work, we have included multiple category patients. At least one category of patient (in this paper the *HCPs* category), has to be served in more than two facilities in a predefined sequence. The objective is to maximise expected net revenue. We model the problem as a MDP and use DP to solve it.

To overcome the challenge of scalability, the current MDP model may be factored into small sub-problems to form multiple D_MDP problems. This factorisation helps us to overcome the curse of dimensionality faced by the MDP formulation. Each subproblem has a decision-making agent. In the past, considerable work has been done on the use of D_MDP. However, there are very few articles about D_MDP models in health care domain. [Hosseini, Hoey, and Cohen \(2014\)](#) used coordinated D_MDP for resource allocation. They modelled each patient as an agent and proposed an auction-based mechanism for allocation to minimise the expected regret for not obtaining the required resource. They compared their algorithm with simulation-based algorithm and other allocation approaches and showed that the proposed approach could be used with a lower cost of computation.

To improve the policy obtained and hence the performance of the D_MDP, we propose a heuristic, simulation-based neighbourhood search Algorithm (SBNSA). Through numerical experiments, we show that SBNSA outperforms all the other queuing rule-based heuristics. The efficient SBNSA is a key contribution to the body of knowledge.

3. Model description

We consider a health care system with L diagnostic facilities at a centre, each having a different dedicated resource (r_l , $l = 1, \dots, L$) to cater to the demand. We assume three categories of patients (i): (1) Health Check-up Patient (*HCPs*), (2) Out-Patient (*OP*), (3) Emergency Patient (*EP*). The demand of a *HCPs* is known apriori. All *HCPs* come at the beginning of the planning horizon and require service at each of the facilities in a predefined order. We can also extend the model to include subcategories of *HCPs* who might require only a subset of facilities. Revenue from the *HCPs* (o_{il}) is realised after they are served at all the facilities. *OP* and *EP* arrive randomly at any facility l over the planning horizon and leave the system after receiving service from that particular facility. We describe the flow of a patient in the system in Figure 1.

For example, consider a multi-facility diagnostic centre with two facilities: (1) TMT and (2) ECG in the mentioned order. *OP* and *EP* are referred to this diagnostic centre from another centre, hospital or doctor for availing of either the TMT or ECG facility. After receiving service from the respective facility, they depart from the diagnostic centre immediately. *HCPs* receives service from both the facilities in the same order and then leave the diagnostic centre. This patient flow is similar to the model in [Chern et al. \(2008\)](#).

We assume that the finite planning horizon is divided into discrete equal time slots ($n = 1, \dots, N$).

Figure 1. Patient flow in the system.

We further assume that the service time of each resource in their respective facility to be same. The current status of the system can be defined based on the previous state, the decision taken, and the random arrival of patients, independent of the states prior to the previous state. Hence, MDP is best suited for providing a good formulation methodology under such circumstances. In this section, we develop a MDP formulation to serve the requested demand over the planning horizon. We aim to identify the optimal policies with the objective of maximising the expected net revenue generated in a cost-effective manner.

We define the MDP as a tuple of five components $\mathcal{M} = [\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}]$ i.e., decision epoch (stage), state, action, transition probabilities, and immediate reward, as given in Puterman (1994). The decision-maker takes action at each decision epoch for each of the facilities to obtain an optimal policy to maximise the reward at the end of the planning horizon. Based on the decision taken in the current state and the random arrival of patients, the system evolves to a new state based on the transition probabilities. Each action generates a reward which drives the model towards better decisions. The value function incorporates future gains or losses, rather than taking myopic decisions based on current status of the system. In the following sections, we discuss this model in detail.

We assume that the demand of HCPs is known a priori and all HCPs enters the system before the first stage. Arrival of OP and EP follows Bernoulli distribution with predefined arrival probability $p_n(\text{new}_{il})$ for categories $i = 2, 3$ and facilities $l = 1, \dots, L$. We further assume that OP and EP arrive independently at all the facilities. We consider fixed service time of equal duration for a given resource at each facility l ($l = 1, \dots, L$) and assume that not more than one request

from OP and EP category can arrive for service at facility l between two decision epoch.

We restrict the length of OP queue (len) by putting a bound only on the number of OP allowed at each facility to keep a check on the problem size. If the maximum len is attained at facility l and an OP is being served at that facility, then the model allows for another OP to join the queue at that facility. If the maximum len is attained and a HCP or EP is being served at a facility, then the model rejects an OP who is trying to join the queue at that facility and incurs rejection penalty cost (c_{2l}). If the maximum len is attained and an OP is being served at a facility, there will be no rejection of HCP who is trying to join the queue as there is no bound on the number of HCP allowed at each facility.

EP is given non-preemptive priority over HCP and OP. As there is no influence of revenue generated by EP on the decision-making, we have safely neglected it in our net revenue calculation for the system (Gocgun et al., 2011). We assume that HCP and OP have the same priority level. The system incurs a small penalty (c_{2l}) if OP is not served at the end of the planning horizon and OP leaves the system permanently. We assume a very high penalty (c_{1l}) for declining service to HCP at the end of the day as HCP demand is scheduled a priori and alternative arrangements have to be made for unmet demand. A summary of all the notations used in the manuscript is given in Appendix 1.

3.1. Global state representation (GSR) MDP model

In this model, the whole system status is captured by a single global state. This global state represents the number of patients of different category waiting at each facility for a given time slot. The action to be taken is the decision of whom to serve next at each facility based on the current system state where the objective is to

maximise the expected net revenue based on the reward function. This action is determined by a single decision-making agent. This formulation helps in an easy communication of state change from one-time decision epoch to next. Figure 2 illustrates the dynamics of the GSR MDP model (for 2 facilities). The components of the model are explained below.

Decision epoch/Stage ($\mathcal{N}$): We assume a work day is divided into N slots of equal duration. For each diagnostic facility, the decision-maker decides on which patient needs to be served by a resource in the next time slot.

State ($\mathcal{S}$): This describes the current status of the system, and is given by

$$\mathcal{S} : s = (w_{11}, w_{21}, w_{31}, w_{12}, w_{22}, w_{32}, \dots, w_{1L}, w_{2L}, w_{3L}) \quad (1)$$

where w_{il} is the number of patients of category i waiting at facility l for service. Here $i = 1$ (HCP), 2 (OP), 3 (EP) and $l = 1, 2, \dots, L$. w_{3l} is 1 if an emergency patient arrives during a previous stage at facility l , 0 otherwise.

Action set ($\mathcal{A}$): The action for each category (i) and facility (l) to be taken at each decision epoch is given by

$$\mathcal{A} : a = (ap_{11}, ap_{21}, ap_{31}, ap_{12}, ap_{22}, ap_{32}, \dots, ap_{1L}, ap_{2L}, ap_{3L}) \quad (2)$$

where ap_{il} ($i = 1, 2, 3$ and $l = 1, 2, \dots, L$) is the number of patients of category i who are selected for service in the next time slot at facility l . Any action taken should satisfy the following constraints:

$$ap_{1l} + ap_{2l} + ap_{3l} \leq r_l, \quad \forall l = 1, 2, \dots, L \quad (3)$$

$$ap_{il} \leq w_{il}, \quad \forall i = 1, 2, 3; l = 1, 2, \dots, L \quad (4)$$

Equation (3) ensures that the number of patients selected for service at facility l does not exceed the number of available resources, r_l , (In emerging economies, many of diagnostic centres catering to multiple categories of patients have only one expensive medical equipment, and this forms the bottleneck. Hence we restrict ourselves to one resource at each facility ($r_l = 1$). This can be easily extended to include multiple resources.) Equation (4) ensures that the number of patients of category i selected for service at facility l does not exceed the number of patients waiting at facility l .

Transition probabilities ($\mathcal{P}$): Based on the action taken at stage n and state s , the system moves to state s'

in stage $n + 1$, where

$$\begin{aligned} s' = & (w_{11} - ap_{11}, w_{21} + new_{21} - ap_{21}, w_{31} \\ & + new_{31} - ap_{31}, w_{12} + ap_{11} - ap_{12}, w_{22} \\ & + new_{22} - ap_{22}, w_{32} + v_{32} - ap_{32}, \dots, w_{1L} \\ & + ap_{1,L-1} - ap_{1L}, w_{2L} + new_{2L} \\ & - ap_{2L}, w_{3L} + new_{3L} - ap_{3L}) \end{aligned} \quad (5)$$

where $new_{il} = 1$ if a new patient of category i arrives at facility l between stage n and $n + 1$ and 0 otherwise. Also HCP patient served at a previous facility is transferred to the next facility. The transition probability for this change of state is given by the conditional probabilities, as follows

$$\mathcal{P} : P_n(s'|s, a) = \prod_{i=1}^3 \prod_{l=1}^L pr_{il} \quad (6)$$

where pr_{il} is the conditional probability of state change for category i at facility l . Detailed computation of these probabilities is shown in the Appendix 2.

Rewards ($\mathcal{R}$): The reward obtained by selecting action a in state s that results in a new state s' is denoted by $o(s, a, s')$, and is given by

$$\begin{aligned} \mathcal{R} : o(s, a, s') &= \sum_i \sum_l o_{il} \cdot ap_{il} - \sum_i \sum_l (w_{il} - ap_{il}) wc_{il} \\ &\quad - \sum_l (new_{2l} \cdot c_{2l} |_{ap_{2l}=0 \text{ and } w_{2l}=len}) \end{aligned} \quad (7)$$

where o_{il} is the revenue obtained from patient of category i at facility l , wc_{il} is the cost of waiting for patient of category i at facility l per time slot ($i = 1, 2$ and $l = 1, 2, \dots, L$). We impose a penalty for a rejection of a request by an OP due to capacity constraints at that facility (that is, bumping due to a constraint on the queue length len).

Value function/The Bellman equation ($\mathcal{V}$): The value function $V_n(s)$ specifies the maximum expected revenue generated over the planning horizon from stage n onwards. Let $A(s)$ be set of all the feasible actions for state s . The cost incurred for not serving patients at the end of planning horizon is given by

$$V_{N+1}(s) = - \sum_{i=1}^2 \sum_{l=1}^L (c_{il} \cdot w_{il}) \quad (8)$$

where c_{il} is the cost incurred for not serving patient of category i at facility l at the end of planning horizon ($i = 1, 2$ and $l = 1, 2, \dots, L$). The Bellman recursive equation is given by

Figure 2. Dynamics of the GSR MDP model (for 2 facilities).

$$\mathcal{V} : V_n(s) = \max_{a \in A(s)} \left\{ o(s, a, s') + \sum_{s' \in \mathcal{S}} P_n(s'|s, a) V_{n+1}(s') \right\} \quad (9)$$

3.2. Structure of optimal policy

To understand the structure of the optimal policy for the GSR MDP model, we performed a numerical experiment. Due to scalability issue with GSR MDP model, we limited the experimental study to two facilities with dedicated resources. We divided the planning horizon of 9 am to 12:30 pm into 7 equal half hour time slots ($N = 7$). Seven *HCP* patients were accepted for service during the start of the planning horizon. The length of *OP* (len) was limited to seven patients. Only one *EP* was allowed. As *EP* has non-preemptive priority over the other patient categories, the queue length of *EP* is always less than or equal to one. The base parameters are given in Table 1. We selected parameters similar to Gocgun et al. (2011). We used parameter values based on general observations from the pilot study we conducted at two diagnostics centres in India and based on expert advice suitable for Indian scenario. One diagnostic centre was a mid-sized hospital in Nagpur, India, and the other diagnostic centre was a health check-up centre in Chennai, India. We use backward induction algorithm of dynamic programming (DP) to solve the Bellman recursive equations and obtain the structure of optimal policy for finite horizon discrete time MDP model.

Figure 3 illustrates the structure of the optimal policy we obtained for the two facility, three category scenario with base case parameters. When $w_{12} \geq 1$, $w_{22} \geq 1$, and $w_{32} = 0$, we observe that *HCP* was always preferred for service at facility two over *OP* in the absence

Table 1. Base case parameters.

Base case parameters	Patient categories		
	<i>HCP</i>	<i>OP</i>	<i>EP</i>
Allowed number of patients	7	7	1
Rejection penalty (Rs.) (c_{ij})	1200	100	0
Waiting cost (Rs.) ($w_{c_{ij}}$)	0	50	0
Arrival probability ($p_n(\text{new}_{ij})$)	–	0.7	0.1
Reward (Rs.) (o_{i1}, o_{i2})	{0, 700}	{350, 400}	{0, 0}

of *EP*. Also we notice a switching pattern at facility one which was influenced by the number of *HCP* waiting at facility two (Figure 3). Switching occurs when the number of *HCP* at facility two was equivalent or more than the time slots remaining for service and so *OP* was served instead of *HCP*. In such scenario, serving *HCP* at facility one was not a good option, as *HCP* would not have left the system even if they were transferred to facility two. Instead *OP* was served, for generating more revenue and for penalty cost incurred for not serving *OP*. The policy we obtained was highly sensitive to all the parameters of the system. Stage of the switch between the actions shifts based on these parameters. We summarise the pattern followed at facility one for all the decision epochs n :

$$ap_{31} = \begin{cases} 1, & \text{if } w_{31} > 0. \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

$$ap_{21} = \begin{cases} 1, & \text{if } w_{31} = 0, w_{11} = 0, w_{21} > 0 \\ & \text{or if } w_{31} = 0, w_{11} > 0, w_{21} > 0, \\ & w_{12} \geq N - n. \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

$$ap_{11} = \begin{cases} 1, & \text{if } w_{31} = 0, w_{21} = 0, w_{11} > 0 \\ & \text{or if } w_{31} = 0, w_{11} > 0, w_{21} > 0, \\ & w_{12} < N - n. \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

No. of HCP at facility 2 $\uparrow$	7	-	-	-	-	-	-	-
	6	-	-	-	-	-	-	OP
	5	-	-	-	-	-	OP	OP
	4	-	-	-	-	OP	OP	OP
	3	-	-	-	OP	OP	OP	OP
	2	-	-	HCP	HCP	OP	OP	OP
	1	-	HCP	HCP	HCP	HCP	OP	OP
	0	1	2	3	4	5	6	7
			Stages $\rightarrow$					

Figure 3. Structure of optimal policy.

The GSR MDP model as discussed suffers from the exponential increase in the size of state space $((HCP \times OP \times EP + 3)^L)$, limiting its application to only problems of small size and a limited number of facilities (suitable for up to 3 facilities). To overcome this limitation, we have developed a decentralised MDP formulation which can handle multiple facilities and larger size problems. We use this switching pattern observed here to improve the policy further as discussed later in detail.

4. Decentralised Markov decision process (D_MDP) model

It is not always possible to have a complete representation of the whole system by one global state as different facilities have their dynamics (different revenue, queue length, etc.). In this section, we present a decentralised representation of the system. Unlike Hosseini et al. (2014) where they considered each patient as an agent, we have modelled each facility as a decision-making agent. This helps us overcome the issues of scalability of the GSR MDP model as it reduces state space to $(L \times (HCP \times OP \times EP + 3))$ and allows for easy extension to multiple facilities. In D_MDP, we factor the GSR MDP model into L semi-independent decentralised MDP models, each maintaining their own state space, actions, and rewards. Each agent at his facility selects a patient to serve to maximise his own expected net revenue. There are two different types of actions executed by each agent: (1) affects only the local state of that agent and 2) joint action - affects the local state of that agent as well as another agent.

One of the major challenges to model D_MDP is to link different agents to communicate the change in state and actions (Bernstein, Givan, Immerman, & Zilberstein, 2002). To link two subsequent facilities, here we assume easy sharing of information regarding joint actions across the facilities. To incorporate the joint action of serving HCP at the previous facility, two different transition probability matrices are maintained which helps the agent arrive at a decision at the current facility. The objective is to maximise the

system expected net revenue. The underlying MDP model remains the same with some modification in the way it is represented. This is explained below and Figure 4 illustrates the dynamics of D_MDP.

Let S_l be the state space for facility l where $S_l : s_l = (w_{1l}, w_{2l}, w_{3l})$ indicates number of patients of each category waiting at facility l . Then the global state is given by

$$S : S_1 \times S_2 \times \dots \times S_L \quad (13)$$

Similarly the global action space is given by

$$A : A_1 \times A_2 \times \dots \times A_L \quad (14)$$

where $A_l : a_l = (ap_{1l}, ap_{2l}, ap_{3l})$, specifies the action taken by an agent for his facility l . Constraints specified for GSR MDP model are applicable for this model also.

For facility l , let the system move from state s_l to s'_l based on the action of the current as well as previous facility ($l = 2, \dots, L$) along with independent random arrival of OP and EP. For the first facility ($l = 1$), there is no new entry/transfer of HCP from previous facility. Thus s'_1 is obtained as follows

$$s'_1 = (w_{11} - ap_{11}, w_{21} + new_{21} - ap_{21}, w_{31} + new_{31} - ap_{31}) \quad (15)$$

For the remaining facilities ($l = 2, \dots, L$) HCP may transfer from $l - 1$ depending upon $ap_{1, l-1}$. Thus s'_l for remaining facilities ($l = 2, \dots, L$) is given by

$$s'_l = (w_{1l} + ap_{1, l-1} - ap_{1l}, w_{2l} + new_{2l} - ap_{2l}, w_{3l} + new_{3l} - ap_{3l}) \quad (16)$$

The *transition probability* is defined for each facility and is based on the current state, action taken in the current facility as well as at the previous facility, and arrival of OP and EP. The L facilities maintain their states and transition probabilities independent of each other. However, they are not completely independent of each other as far as the state transition is concerned because HCP is transferred from facility $l - 1$ to l . The decentralised approach proposed in the paper links the L single-facility MDP models by sequentially linking the HCP queues. We achieve this as explained below.

Let T_{1l} be the transfer function to capture the joint action for HCP which is defined as follows

$$T_{1l} = \begin{cases} 1 & \text{if } ap_{1l} \geq 1 \\ 0 & \text{if } ap_{1l} = 0 \end{cases} \quad (17)$$

T_{1l} takes value 1 if action at facility l is to serve HCP, 0 otherwise.

If a HCP is served at facility $l - 1$ ($ap_{1, l-1} \geq 1$), the patients are transferred to facility l . That causes a change in the state of both $l - 1$ and l facility. We assume that the decision at facility $l - 1$ will be taken

Figure 4. Dynamics of D_MDP model.

just before the decision at facility l for same stage n . To capture this dynamics for D_MDP, we maintain two different Transition Probability Matrices (TPM) to communicate the effect of action to serve HCP ($ap_{1,l-1}$) at the previous facility ($l-1$) based on the state of facility l using the above-defined transition function. They are indicated as $TPM1$ and $TPM2$. If the action taken at the facility $l-1$ is to serve HCP ($T_{1,l-1} = 1$), then the matrix $TPM2$ is used causing w_{1l} to be increased by $ap_{1,l-1}$. If the action taken at facility $l-1$ is to not serve HCP ($T_{1,l-1} = 0$), then the matrix $TPM1$ is used leaving w_{1l} unchanged by previous facility action. We maintain two separate policies corresponding to different TPM and select the appropriate policy from two available sets depending on $ap_{1,l-1}$. The transition probability is given by

$$P_{nl}(s'_l | s_l, a_l) = (pr_{1l}^1 \times (1 - T_{1,l-1}|_{l>1}) + pr_{1l}^2 \times T_{1,l-1}|_{l>1}) \times \prod_{i=2}^3 pr_{il} \quad \text{for } l = 2, \dots, L \quad (18)$$

where pr_{il} is as defined before and pr_{il}^1 and pr_{il}^2 are the probabilities for HCP corresponding to $TPM1$ and $TPM2$, respectively. The derivation of these probabilities is given in Appendix 2.

Tables 2 and 3 illustrates the possible states accessible and corresponding transition probabilities ($TPM1$ and $TPM2$, respectively) that are calculated depending upon the action to serve HCP at previous facility ($ap_{1,l-1}$). Suppose the current state at facility $l-1$ has 1 HCP, 1 OP and 0 EP in the queue, $s_{l-1} = (1, 1, 0)$, and decision is to serve 1 HCP at facility 2 ($ap_{1,l-1} = 1$). Assume that the arrival probability of OP and EP are 0.7 and 0.1, respectively ($p_n(\text{new}_{2l}) = 0.7$ and $p_n(\text{new}_{3l}) = 0.1$). Table 2 shows the transition probabilities ($TPM1$) when the action is not to serve HCP at facility $l-1$

($T_{1,l-1} = 0$). Table 3 shows transition probabilities ($TPM2$) when action is to serve HCP at facility $l-1$ ($T_{1,l-1} = 1$).

The rewards for facility l is given by

$$\mathcal{R}_l : o_l(s_l, a_l, s'_l) = \sum_i o_{il} \cdot ap_{il} - \sum_i (w_{il} - ap_{il})wc_{il} - (\text{new}_{2l} \cdot c_{2l} |_{ap_{2l}=0 \text{ and } w_{2l}=len}) \quad \forall l = 1, 2, \dots, L \quad (19)$$

where $o_l(s_l, a_l, s'_l)$ is the immediate reward obtained at facility l in state s_l and transitioning to state s'_l due to action set a_l .

The end of day penalty for facility l is given by

$$V_{N+1,l}(s_l) = - \sum_{i=1}^2 (c_{il} \cdot w_{il}) \quad \forall l = 1, 2, \dots, L \quad (20)$$

where $V_{N+1,l}(s_l)$ is the total rejection penalty for not fulfilling patient requests at facility l at the end of the planning horizon.

The value function for D_MDP is given by

$$\mathcal{V}_l : V_{nl}(s_l) = \max_{a_l \in A(s_l)} \{o_l(s_l, a_l, s'_l) + \sum_{s'_l \in \mathcal{S}_l} P_{nl}(s'_l | s_l, a_l) V_{n+1,l}(s'_l)\} \quad \forall l = 1, 2, \dots, L \quad (21)$$

Equation (21) is the Bellman recursive equation where $V_{nl}(s_l)$ is the value of state s_l at each facility l and stage n .

5. Solution methodology and Heuristic rules

We use DP algorithm to get the initial solution for GSR MDP and D_MDP model. We use the results from the

Table 2. TPM1: Action at facility $l - 1$ is not to serve HCP.

$T_{1,l-1} = 0$		Possible states						
State	Action	(0, 2, 0)	(0, 1, 0)	(0, 2, 1)	(0, 1, 1)	(1, 2, 0)	(1, 1, 0)	(1, 2, 1)
(1, 1, 0)	(1, 0, 0)	1x0.7x0.9	1x0.3x0.9	1x0.7x0.1	1x0.3x0.1	0x0.7x0.9	0x0.3x0.9	0x0.7x0.1

Table 3. TPM2: Action at facility $l - 1$ is to serve HCP.

$T_{1,l-1} = 1$		Possible states						
State	Action	(0, 2, 0)	(0, 1, 0)	(0, 2, 1)	(0, 1, 1)	(1, 2, 0)	(1, 1, 0)	(1, 2, 1)
(1, 1, 0)	(1, 0, 0)	0x0.7x0.9	0x0.3x0.9	0x0.7x0.1	0x0.3x0.1	1x0.7x0.9	1x0.3x0.9	1x0.7x0.1

policy structure of GSR MDP explained earlier to improve the solution obtained for D_MDP by enforcing the following actions in absence of EP ,

$$\begin{aligned}
 ap_{1L} &= 1 \quad \text{if } w_{1L} > 0 \\
 ap_{2l} &= 1 \quad \text{if } w_{2l} > 0, \\
 &\quad \text{for } n = N \text{ and } l = 1, 2, \dots, L - 1
 \end{aligned}$$

Since the agent at each facility tries to maximise his average net revenue in D_MDP model, it results in a suboptimal policy for the system as a whole. Using the insights from the policy structure, we develop a simulation-based neighbourhood search algorithm (SBNSA) to improve the policy obtained by DP algorithm. In next section, we discuss the algorithm in detail.

5.1. Simulation-based neighbourhood search algorithm (SBNSA)

The initial policy from DP is first modified by incorporating learning from the structure of optimal policy. This is sent as an input to the simulator to generate the average reward. The output from the above step is subjected to neighbourhood search, where we first generate a random number to decide whether to enter the current stage to change the actions in that particular stage. If the decision is to enter, then we again generate a string of random numbers to select which state's action should be changed given that no EP is waiting ($w_{3l} = 0$). We decrease the probability of entering stages towards the end of the planning horizon to maintain effective policy structure. We use low probability to change the action to restrict the search in the neighbourhood of current policy. A high level description of the algorithm is given in Figure 5, the detailed steps of SBNSA is given in Algorithm 1 and an explanation of the pseudocode is given in Appendix 3. We refer to the heuristic SBNSA applied to D_MDP problem as S_D_MDP.

Note that in step 14 of Algorithm 1, we change action ap_{1l} (or ap_{2l}) with ap_{2l} (or ap_{1l}) for the selected state.

Figure 5. SBNSA flow chart.

5.2. Other heuristics

For comparative analysis of the policies obtained by the earlier mentioned methods, following are the other heuristics based on queuing rules and improved by SBNSA.

- (1) *Priority HCP SBNSA* (S_P_HCP): A HCP patient is selected from w_{1l} at facility l for service in the absence of EP . It is similar to first-in-first-Out (FIFO) rule, as all the HCP request arrive first in the beginning of planning horizon at the system. The initial policy generated by Priority HCP is used in SBNSA to improve the initially obtained policy.
- (2) *Priority OP SBNSA* (S_P_OP): An OP patient is selected from w_{2l} at facility l for service in the absence of EP . The initial policy generated by Priority OP is used in SBNSA to improve the initially obtained policy.

Table 4. Experimental Study 1: setup for system with 3 facilities.

	Category	Facilities		
		1	2	3
Reward (Rs.) (o_{ij})	OP	300	350	400
	HCP	0	0	(900, 700)
Rejection penalty (Rs.) (c_{ij})	OP	(100, 200)	(100, 200)	(100, 200)
	HCP	(1200, 1500)	(1200, 1500)	(1200, 1500)
Waiting cost (Rs.) ($w_{c_{ij}}$)	OP	(0, 50)	(0, 50)	(0, 50)
	HCP	0	0	0
Arrival probability ($p_n(new_{2l})$)	OP	(0.7, 0.5)	(0.7, 0.5)	(0.7, 0.5)

Algorithm 1 Simulation-Based Neighbourhood Search Algorithm (SBNSA)

```

1: main NS_ALGO
2:   Global:  $Sim\_iter = X$  days;  $NS\_rounds = Y$  iterations;
    $Best(Revenue) = 0$   $enter\_initial\_1$ ;  $ch\_action$ 
3:   Obtain Initial( $\pi$ ) by DP Algorithm and modify
   it using learning from structure of optimal policy
4:    $Change(\pi, Revenue, HCP\_rej, OP\_rej) \leftarrow$ 
   ( $\pi, SIMULATOR(Sim\_len, Initial(\pi))$ )
5:   for all  $NS\_rounds$  do
6:     if  $Change(Revenue) \geq Best(Revenue)$  then
7:        $Best(\pi, Revenue, HCP\_rej, OP\_rej) \leftarrow Change(\pi, Revenue,$ 
        $HCP\_rej, OP\_rej)$ 
8:     end if
9:      $Change(\pi) \leftarrow Best(\pi)$ 
10:     $enter\_initial \leftarrow enter\_initial\_1$ 
11:    for all  $L$  do
12:       $size\_state\_change_l \leftarrow$  count of states and
      their index for each  $l$  which satisfy : ( $w_{1l} \geq 1 \ \&\& \ w_{2l} \geq 1$ ) ||  $w_{3l} ==$ 
      0
13:      for all  $N - 1$  do
14:        if  $enter\_initial \geq rand()$  then
15:          for all  $size\_state\_change_l$  do
16:            if  $ch\_action \geq rand()$  then
17:              Update  $Change(\pi): ap_{1l} \setminus ap_{2l} \leftarrow ap_{2l} \setminus ap_{1l}$ 
18:            end if
19:          end for
20:           $enter\_initial \leftarrow enter\_initial\_1 \times enter\_initial$ 
21:        end if
22:      end for
23:    end for
24:     $Change(\pi, Revenue, HCP\_rej, OP\_rej) \leftarrow$ 
    ( $\pi, SIMULATOR(Change(\pi))$ )
25:  end for
26: end main
27: function  $SIMULATOR(Sim\_iter, \pi)$ 
28:   return ( $Revenue, HCP\_rej, OP\_rej$ )
29: end function

```

- (3) *Random Selection SBNSA (S_Rnd)*: One patient is randomly selected from w_{1l} or w_{2l} at facility l , when no EP is waiting for service. The initial policy generated by the random selection is used in SBNSA to improve the initially obtained policy.

6. Experimental design and scenarios tested

We perform experimental studies to validate our formulation and check the robustness of the model

described. This section details the experimental setup.

6.1. Experimental Study 1

The base parameters used in the numerical experiments are according to the observations we had made during our pilot studies at two multi-facility diagnostic centres in India that caters to different categories of patients. With GSR MDP as the base model, we create several data-sets wherein certain parameters observed at these two diagnostic centres were systematically changed to enable comparative analysis. To restrict the state space and to ensure faster computation, we restrict the problem size by limiting the following: the system operating time between 10 am to 12 am (4 stages); the number of facilities $L = 3$ where each facility on an average has the same service time; and the number of HCP appointments to 2. With a similar experimental set-up for obtaining the structure of optimal policy, we also varied the number of OP (len). We observed that after a threshold there is no change in the expected net revenue due to variation in len . Hence, we have set len to this threshold value of 2. Also by changing five parameters (specifically, rejection penalty for HCP and OP , OP arrival probability, HCP revenue and OP waiting cost), we conduct a full factorial experiment with 32 (2^5) different data-sets (instances). The instances used are given in Table D1(a) of Appendix 4. The value of the remaining parameters is the same as listed in Table 4 (We indicate the two levels of parameter values to create instances inside parenthesis. For example, two levels for Reward for serving HCP are Rs. 900 and Rs. 700).

We use Matlab 2014a to code for all the above heuristics. Each instance of N time slots was simulated for 1000 days and 100 rounds of SBNSA. We recorded computation time, average net revenue, average number of HCP rejected, and the average number of OP rejected for each instance. The percentage gap in the average net revenue is given by,

$$= \frac{\text{average net revenue for the base model} - \text{average net revenue for the other policies model}}{\text{average net revenue for the base model}} \times 100$$

Table 5. Experimental Study 2: setup for system with 5 facilities.

		Facilities				
	Category	1	2	3	4	5
Reward (Rs.) (o_{ij})	OP	300	350	400	450	250
	HCP	0	0	0	0	(1750, 1250)
Rejection penalty (Rs.) (c_{ij})	OP	(100, 200)	(100, 200)	(100, 200)	(100, 200)	(100, 200)
	HCP	(2500, 2700)	(2500, 2700)	(2500, 2700)	(2500, 2700)	(2500, 2700)
Waiting cost (Rs.) ($w_{c_{ij}}$)	OP	(0, 50)	(0, 50)	(0, 50)	(0, 50)	(0, 50)
	HCP	0	0	0	0	0
Arrival probability ($p_n(\text{new}_{ij})$)	OP	(0.7, 0.5)	(0.7, 0.5)	(0.7, 0.5)	(0.7, 0.5)	(0.7, 0.5)

Figure 6. Comparison of average net revenue for different policies of 32 instances for Experimental Study 1 (3 facilities).

Similar performance metrics have been used in Gocgun et al. (2011). We also compute p -values using paired t -test to compare the average net revenue of GSR MDP model and S_D_MDP obtained from simulation of each instance for 1000 days. We use a two tail paired t -test to establish that there is no significance difference between the mean of GSR MDP and S_D_MDP for each instance with the hypothesis:

$$H_0 : \mu_{\text{GSRMDP}} - \mu_{\text{S_D_MDP}} = 0$$

$$H_a : \mu_{\text{GSRMDP}} - \mu_{\text{S_D_MDP}} \neq 0$$

We tabulate the average percentage difference, p -values and average number of HCP and OP rejected from the results of the Experimental Study 1. These results are summarised in Tables D2 and D3 in Appendix 4.

In all these instances, we fail to reject null hypothesis as there does not exist significant differences between the performance measures for both models at 5% significance level. This led to the inference that S_D_MDP performs as good as GSR MDP. Also, the computation

time for S_D_MDP was approximately 1 second for each round of SBNSA, hence around 110 seconds for each instance. Whereas GSR MDP took 3508 seconds, which was 32 times slower than S_D_MDP.

6.2. Experimental Study 2

We extend the previous set-up for D_MDP model to five facilities. The parameters value are in Table 5. We further increased the size of the problem by allowing 6 HCP and 3 OP in the system between 9 am to 2 pm (10 stages). Similar to Experimental Study 1, we systematically changed the parameters to create 32 instances (Table D1(b) in Appendix 4).

As noted earlier, there is no significant difference between the optimal value of the GSR MDP and the one obtained by S_D_MDP. Hence in this study, we use S_D_MDP as the base model for comparison. Using one-tailed paired t -test at 5% significance level, we compared S_D_MDP with second best heuristics based on the average net revenue for each instance simulated

Graph is shown continuous to demonstrate dominance of particular policy

Figure 7. Comparison of average net revenue for different policies of 32 instances for Experimental Study 2 (5 facilities).

for 1000 days. The hypothesis tested in this study is:

$$H_0 : \mu_{S_D_MDP} - \mu_{\text{SecondBest}} \leq 0$$

$$H_a : \mu_{S_D_MDP} - \mu_{\text{SecondBest}} > 0$$

The average percentage difference, p -values and average number of HCP and OP rejected from the results of the Experimental Study 2 are given in Tables D4 and D5 in Appendix 4.

6.3. Numerical results: key insights

First, we discuss the performance of S_D_MDP . As expected, the GSR MDP model performs better than all other policies. However, due to scalability issues and prohibitive run times, the GSR MDP model could not be extended beyond three facilities. The SBNSA algorithm applied to the decentralised MDP (S_D_MDP) improves the solution significantly with an average net revenue close to that of the GSR MDP policy. The performance of S_D_MDP was compared with that of GSR MDP using a two-tailed paired t -test and the results from Table D2 show that there is no significant difference in their relative performance with respect to average net revenue. This is further validated by Figure 6, which illustrates the relative performance of all heuristics with respect to average net revenue. The computation time for GSR MDP (3508 seconds) is 32 times slower than that of S_D_MDP (110 seconds).

Having established the efficiency of S_D_MDP , we conducted Experimental Study 2 with five facilities. Using S_D_MDP as a benchmark, we compared against

other heuristics. Using a one-tailed paired t -test, we observe that S_D_MDP performs significantly better than all the other heuristics and the results are shown in Table D4. Figure 7 demonstrates the dominance of S_D_MDP with respect to other heuristics.

Another key insight from the numerical experiments is that the number of HCP at the preceding facilities decides the preference for serving OP at preceding facilities. When number of HCP waiting at last facility is more than the remaining time slots, OP is preferred for service at preceding facilities, because even if the HCP is served and transferred to next facility, HCP will not leave the system. This is consistent with the switching behaviour observed in Figure 3.

7. Conclusions and future scope

To summarise, we modify the classical MDP framework as the GSR MDP model to allow for dynamic scheduling of patients arriving at a multi-facility diagnostic centre. Due to the explosion in the state space and scalability issues in the GSR MDP model, we further factor the GSR MDP model to obtain the D_MDP model. We develop the SBNSA heuristic to improve the policies obtained by D_MDP and propose other rule-based heuristics. Our main contribution is the S_D_MDP heuristic which outperforms all the other queuing rule-based heuristics that are currently used by decision-makers.

This work can be extended in many directions. One of the limitations of the current study is the assumption that the service time is the same at each facility. While this holds true for a subset of tests, in general it does

not hold. A natural extension of this work is to check the results if this assumption were relaxed. We assume health care patients to follow a particular sequence to different facilities. In reality, continuity of examination procedures is required only for a subset of procedures. It would be interesting to add this dimension and analyse the system. Our objective has been to maximise net revenue. It would be interesting to model the system with other objectives used by diagnostics centres such as minimising patient waiting time, meeting certain service level targets, and optimal utilisation of resources. A sensitivity analysis for identifying the number of HCP that should be accepted into the system would give further managerial insights to the problem. While the underlying mathematical model can be scaled up to capture many categories and facilities, the greatest challenge is in the implementation of solution methodologies as the MDP model poses computational challenges. One way to overcome computational challenges and prohibitive run times might be to explore solution methods like Reinforcement Learning.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

- Bernstein, D. S., Givan, R., Immerman, N., & Zilberstein, S. (2002). The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research*, 27, 819–840.
- Chern, C. C., Chien, P. S., & Chen, S. Y. (2008). A heuristic algorithm for the hospital health examination scheduling problem. *European Journal of Operational Research*, 186(3), 1137–1157.
- Gocgun, Y., Bresnahan, B. W., Ghate, A., & 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.
- Green, L. V., Savin, S., & Wang, B. (2006). Managing patient service in a diagnostic medical facility. *Operations Research*, 54(1), 11–25.
- Gupta, D., & Denton, B. (2008). Appointment scheduling in health care: Challenges and opportunities. *IIE Transactions*, 40(9), 800–819.
- Hosseini, H., Hoey, J., & Cohen, R. (2014). *A coordinated MDP approach to multi-agent planning for resource allocation, with applications to healthcare*. arXiv preprint arXiv:1407.1584.
- Hulshof, P. J., Kortbeek, N., Boucherie, R. J., Hans, E. W., & Bakker, P. J. (2012). Taxonomic classification of planning decisions in health care: A structured review of the state of the art in OR/MS. *Health Systems*, 1(2), 129–175.
- Kolisch, R., & Sickinger, S. (2008). Providing radiology health care services to stochastic demand of different customer classes. *OR Spectrum*, 30(2), 375–395.
- Patrick, J. (2012). A Markov decision model for determining optimal outpatient scheduling. *Health Care Management Science*, 15(2), 91–102.
- Patrick, J., Puterman, M. L., & Queyranne, M. (2008). Dynamic multipriority patient scheduling for a diagnostic resource. *Operations Research*, 56(6), 1507–1525.
- Puterman, M. L. (1994). *Markov decision processes: Discrete stochastic dynamic programming*. New York, NY: John Wiley.
- Rais, A., & Viana, A. (2010). Operations research in healthcare: A survey. *International Transactions in Operational Research*, 18(1), 1–31.
- Sauré, A., Patrick, J., Tyldesley, S., & Puterman, M. L. (2012). Dynamic multi-appointment patient scheduling for radiation therapy. *European Journal of Operational Research*, 223(2), 573–584.
- Schütz, H. J., & Kolisch, R. (2012). Approximate dynamic programming for capacity allocation in the service industry. *European Journal of Operational Research*, 218(1), 239–250.
- Vermeulen, I. B., Bohte, S. M., Elkhuisen, S. G., Lameris, H., Bakker, P. J., & Poutré, H. L. (2009). Adaptive resource allocation for efficient patient scheduling. *Artificial Intelligence in Medicine*, 46(1), 67–80.
- Vlah, S., Lukač, Z., & Pacheco, J. (2011). Use of VNS heuristics for scheduling of patients in hospital. *Journal of the Operational Research Society*, 62(7), 1227–1238.

Appendix 1

Notations and parameters

Notation

$i = 1(HCP), 2(OP), 3(EP)$	Patient categories
$l = 1, \dots, L$	Types of diagnostic facilities
$n = 1, \dots, N$	Time slot (stages)
j	Superscripted index for different TPM
len	Maximum number of OP allowed at any facility

Model parameters

L	Number of types of diagnostic facilities
$A(s)$	Feasible action space
I	Number of patient categories
$o(s, a, s')$	Reward function for state s transitioning to s' due to action a
$V_n(s)$	Value function for state s and stage n
w_{il}	Number of patients of category i waiting at facility l
ap_{il}	Number of category i patients selected for service at facility l
new_{il}	Binary arrival, 1 if patient of category i arrives at facility l between n and $n + 1$, 0 otherwise
pr_{il}	Probability of state change corresponding to the action to serve HCP ($i = 1$) or due to random arrival/no arrival of OP ($i = 2$) or EP ($i = 3$) at facility l
pr_{il}^j	Probability of state change corresponding to the action to serve HCP ($i = 1$) or due to random arrival/no arrival of OP ($i = 2$) or EP ($i = 3$) at facility l for TPM j dynamics
T_{il}	Binary variable, 1 if the action is to serve HCP at facility l ($ap_{il} \geq 1$), 0 otherwise
o_{il}	Reward for serving patient of category i at facility l
c_{il}	Penalty cost per patient for not serving patient of category i at facility l at the end of planning horizon
wc_{il}	Waiting cost per patient of category i at facility l

Appendix 2

Computing transition probabilities for TPM

Probability for HCP at facility 1 (GSR MDP and D_MDP).

$$pr_{11}|_{l=1} = pr_{11}^1 = pr_{11}^2 = \begin{cases} 1 & \text{if } w'_{11} = w_{11} - ap_{11} \\ 0 & \text{if } w'_{11} \neq w_{11} - ap_{11} \end{cases} \quad (B1)$$

Probability for HCP at facility 2 to L (GSR MDP).

$$pr_{1l}|_{l=2 \dots L} = \begin{cases} 1 & \text{if } w'_{1l} = w_{1l} - ap_{1l} + ap_{1,l-1} \\ 0 & \text{if } w'_{1l} \neq w_{1l} - ap_{1l} + ap_{1,l-1} \end{cases} \quad (B2)$$

Probability for OP at each facility l (GSR MDP and D_MDP).

$$pr_{2l} = \begin{cases} 1 - p_n(new_{2l}) & \text{if } w'_{2l} = w_{2l} - ap_{2l} \\ p_n(new_{2l}) & \text{if } w'_{2l} = w_{2l} - ap_{2l} + 1 \text{ and } w_{2l} \neq len \\ 1 & \text{if } w'_{2l} = w_{2l} - ap_{2l}; w_{2l} = len \text{ and } ap_{2l} = 0 \\ 0 & \text{otherwise.} \end{cases} \quad (B3)$$

Probability for EP at each facility l (GSR MDP and D_MDP).

$$pr_{3l} = \begin{cases} 1 - p_n(new_{3l}) & \text{if } w'_{3l} = w_{3l} - ap_{3l} \\ p_n(new_{3l}) & \text{if } w'_{3l} = w_{3l} - ap_{3l} + 1 \\ 0 & \text{otherwise.} \end{cases} \quad (B4)$$

Probability for HCP corresponding to TPM1 (pr_{1l}^1) at facility 2 to L (D_MDP).

$$pr_{1l}^1|_{l=2 \dots L} = \begin{cases} 1 & \text{if } w'_{1l} = w_{1l} - ap_{1l} \\ 0 & \text{if } w'_{1l} \neq w_{1l} - ap_{1l} \end{cases} \quad (B5)$$

Probability for HCP corresponding to TPM2 (pr_{1l}^2) at facility 2 to L (D_MDP).

$$pr_{1l}^2|_{l=2 \dots L} = \begin{cases} 1 & \text{if } w'_{1l} = w_{1l} - ap_{1l} + ap_{1,l-1} \\ 0 & \text{if } w'_{1l} \neq w_{1l} - ap_{1l} + ap_{1,l-1} \end{cases} \quad (B6)$$

Appendix 3

Step by step explanation of SBNSA pseudocode

- (1) Starting the main function body named NS_ALGO which contains the main logic for SBNSA
- (2) Define global parameters: System is simulated for X days (Sim_iter); policy is subjected to Y iteration of Neighborhood search (NS_rounds); probability to choose any particular stage to change the action ($enter_initial_1$); a small probability to change the action of the current state (ch_action)
- (3) Initial policy ($Initial(\pi)$) is obtained using DP algorithm and modified by using the structure of optimal policy
- (4) Initial policy ($Initial(\pi)$) is simulated for X days by calling simulation function (SIMULATOR) and obtained average of Revenue, number of HCP rejected (HCP_rej) and number of OP rejected (OP_rej) is stored as *Change* solution along with Initial policy (π)
- (5) Start For loop for Neighborhood search is initiated for Y iterations
- (6) New average revenue ($Change(Revenue)$) obtained from (SIMULATOR) is compared with Best average revenue ($Best(Revenue)$) stored and If $Change(Revenue)$ is equal or better than the $Best(Revenue)$, then go to next step
- (7) Store the *Change* solution along with $Change(\pi)$ in *Best* solution
- (8) End the If statement for updating *Best* solution
- (9) Copy the Best policy ($Best(\pi)$) in policy ($Change(\pi)$) which will be subjected to Neighborhood search
- (10) Update the probability of choosing a stage ($enter_initial$) by the default choosing probability ($enter_initial_1$)
- (11) Start For loop for all the L facilities
- (12) To update the policy, only states which has at least one patient of both HCP ($w_{1l} \geq 1$) and OP ($w_{2l} \geq 1$) and no patient of EP ($w_{3l} == 0$) is subjected to Neighborhood search. Count of such states and their index for each facility l , is stored in variable $size_state_change_l$ and $index_state_change_l$, respectively
- (13) Start For loop for $N - 1$ stages
- (14) Generate a uniform random number between $[0,1]$ using $rand()$ function and If it is less than equal to $enter_initial$ then go to next step else next stage
- (15) Start For loop for $size_state_change_l$ states obtained in step 9 corresponding to facility l
- (16) Generate a uniform random number between $[0,1]$ using $rand()$ function and If it is less than equal to ch_action then go to next step else next state
- (17) Change action for HCP (ap_{1l}) with action for OP (ap_{2l}) or action for OP (ap_{2l}) with action for HCP (ap_{1l}) for the selected state from $index_state_change_l$ of facility l and update $Change(\pi)$
- (18) End the If statement of $ch_actions$
- (19) End the For loop of states
- (20) From initial observation, policy for end stages were more in line with the optimal policy. Hence the probability of choosing end stages is decreased $enter_initial$ to maintain effective policy structure. Update the $enter_initial = enter_initial_1 \times enter_initial$
- (21) End the If statement for $enter_initial$
- (22) End the For loop of stages
- (23) End the For loop of facilities
- (24) Newly updated policy ($Change(\pi)$) is simulated for X days by calling simulation function (SIMULATOR) and obtained average of Revenue, number of HCP rejected (HCP_rej) and number of OP rejected (OP_rej) is stored as *Change* solution along with $Change(\pi)$
- (25) End the For loop of Neighborhood search
- (26) End the main Neighborhood Search Algorithm (NS_ALGO) body
- (27) Function (SIMULATOR) to perform simulation with input as Sim_iter and policy (π)
- (28) Function performs the simulation for X number of days and returns the average of Revenue, HCP_rej and OP_rej
- (29) End the SIMULATOR Function

Appendix 4

Numerical results

Table D1. Instances for experimental studies.

No.	(a) Experimental Study 1					(b) Experimental Study 2				
	c_{1l}	wc_{2l}	o_{1l}	c_{2l}	new_{2l}	c_{1l}	wc_{2l}	o_{1l}	c_{2l}	new_{2l}
1	1200	0	900	100	0.7	2500	0	1750	100	0.7
2	1500	0	900	100	0.7	2700	0	1750	100	0.7
3	1200	50	900	100	0.7	2500	50	1750	100	0.7
4	1500	50	900	100	0.7	2700	50	1750	100	0.7
5	1200	0	700	100	0.7	2500	0	1250	100	0.7
6	1500	0	700	100	0.7	2700	0	1250	100	0.7
7	1200	50	700	100	0.7	2500	50	1250	100	0.7
8	1500	50	700	100	0.7	2700	50	1250	100	0.7
9	1200	0	900	200	0.7	2500	0	1750	200	0.7
10	1500	0	900	200	0.7	2700	0	1750	200	0.7
11	1200	50	900	200	0.7	2500	50	1750	200	0.7
12	1500	50	900	200	0.7	2700	50	1750	200	0.7
13	1200	0	700	200	0.7	2500	0	1250	200	0.7
14	1500	0	700	200	0.7	2700	0	1250	200	0.7
15	1200	50	700	200	0.7	2500	50	1250	200	0.7
16	1500	50	700	200	0.7	2700	50	1250	200	0.7
17	1200	0	900	100	0.5	2500	0	1750	100	0.5
18	1500	0	900	100	0.5	2700	0	1750	100	0.5
19	1200	50	900	100	0.5	2500	50	1750	100	0.5
20	1500	50	900	100	0.5	2700	50	1750	100	0.5
21	1200	0	700	100	0.5	2500	0	1250	100	0.5
22	1500	0	700	100	0.5	2700	0	1250	100	0.5
23	1200	50	700	100	0.5	2500	50	1250	100	0.5
24	1500	50	700	100	0.5	2700	50	1250	100	0.5
25	1200	0	900	200	0.5	2500	0	1750	200	0.5
26	1500	0	900	200	0.5	2700	0	1750	200	0.5
27	1200	50	900	200	0.5	2500	50	1750	200	0.5
28	1500	50	900	200	0.5	2700	50	1750	200	0.5
29	1200	0	700	200	0.5	2500	0	1250	200	0.5
30	1500	0	700	200	0.5	2700	0	1250	200	0.5
31	1200	50	700	200	0.5	2500	50	1250	200	0.5
32	1500	50	700	200	0.5	2700	50	1250	200	0.5

Table D2. Percentage difference between average net revenue for each instance obtained using GSR MDP against other heuristics from Experimental Study 1 (3 facilities) simulated over 1000 days.

No.	D_MDP	S_D_MDP	S_P_HCP	S_P_OP	S_Rnd	<i>p</i> - value
1	35.90	2.43	5.62	23.08	7.59	0.24
2	46.76	2.61	6.10	30.24	32.78	0.28
3	40.27	3.74	8.32	34.68	33.56	0.19
4	54.68	7.96	9.22	50.04	47.16	0.05
5	36.16	3.12	6.54	30.02	25.09	0.19
6	49.01	3.08	7.20	31.58	34.24	0.26
7	41.55	6.05	10.10	33.02	32.85	0.08
8	59.66	5.93	11.46	52.40	50.15	0.16
9	39.36	4.01	8.25	30.33	27.10	0.16
10	52.95	4.10	9.08	44.71	36.75	0.22
11	45.77	7.38	12.14	32.88	32.61	0.06
12	65.57	7.00	14.26	54.05	54.36	0.15
13	40.39	5.59	9.94	28.18	29.00	0.09
14	57.25	4.72	11.17	45.72	39.54	0.21
15	49.19	3.33	15.23	49.97	29.39	0.30
16	76.17	8.04	18.56	57.77	57.63	0.18
17	32.46	2.50	5.77	6.43	9.23	0.27
18	42.10	2.12	6.35	37.23	11.65	0.34
19	35.20	2.72	7.53	7.88	25.07	0.29
20	47.17	3.03	8.94	9.93	14.26	0.31
21	33.80	2.98	6.53	6.95	9.84	0.25
22	45.61	3.31	7.67	8.90	12.84	0.28
23	37.49	3.36	8.88	8.86	8.11	0.27
24	52.97	5.70	10.85	11.60	16.46	0.21
25	34.32	4.61	7.28	7.38	24.50	0.15
26	45.49	2.69	8.58	9.25	13.13	0.32
27	37.89	6.75	9.64	9.26	8.63	0.10
28	52.28	9.33	11.57	11.85	36.67	0.08
29	36.27	2.96	8.48	8.19	7.36	0.28
30	50.45	3.60	10.26	10.63	17.02	0.30
31	41.01	4.34	10.24	8.71	9.89	0.26
32	61.08	9.07	14.78	14.52	13.32	0.14

Table D3. Average number of rejection of *HCP* and *OP* (Base Case) from Experimental Study 1.

3 facilities	GSRMDP	D_MDP	S_D_MDP	S_P_HCP	S_P_OP	S_Rnd
Average <i>HCP_rej</i>	0.53	0.97	0.55	0.54	0.85	0.82
Average <i>OP_rej</i>	2.76	2.26	2.82	2.97	2.15	2.29

Table D4. Percentage difference between average net revenue for each instance obtained using S_D_MDP against other heuristics from Experimental Study 2 (5 facilities) simulated over 1000 days.

No.	D_MDP	S_P_HCP	S_P_OP	S_P_rnd	<i>p</i> -value
1	122.75	41.86	180.91	178.06	0.00
2	175.98	49.58	273.31	151.89	0.00
3	357.32	150.74	298.51	167.07	0.00
4	280.84	118.87	293.88	156.61	0.00
5	282.00	121.75	371.26	233.49	0.00
6	1251.08	456.21	1725.68	951.15	0.00
7	177.69	79.95	110.74	93.37	0.00
8	83.77	29.06	54.84	29.17	0.00
9	302.37	146.92	355.73	205.84	0.00
10	3089.81	1215.81	3965.75	2401.32	0.00
11	148.59	29.63	49.76	38.31	0.00
12	125.19	47.99	75.78	59.29	0.00
13	489.33	223.32	430.05	281.62	0.00
14	180.26	83.44	178.77	109.96	0.00
15	112.49	22.50	24.51	30.55	0.00
16	39.85	0.77	0.82	0.37	0.39*
17	24.27	40.09	230.24	136.81	0.01
18	37.24	55.71	333.55	199.10	0.00
19	325.94	292.51	954.31	506.34	0.00
20	121.36	99.83	382.19	213.96	0.00
21	88.52	175.94	789.55	488.45	0.00
22	685.00	1259.61	6282.33	3911.51	0.01
23	64.65	70.98	140.92	71.39	0.00
24	46.15	47.00	115.00	61.67	0.00
25	43.00	93.63	363.90	187.16	0.01
26	93.02	188.68	811.66	500.16	0.01
27	177.75	193.00	330.15	195.36	0.00
28	81.75	85.55	181.80	94.00	0.00
29	720.63	1154.79	2445.65	1498.41	0.00
30	48.45	108.36	338.45	171.75	0.00
31	42.09	37.98	61.34	31.30	0.00
32	40.01	49.14	66.98	37.00	0.00

Table D5. Average number of rejection of HCP and OP in 5 facilities (Base case) from Experimental Study 2.

5 facilities	D_MDP	S_D_MDP	S_P_HCP	S_P_OP	S_Rnd
Average HCP_rej	3.00	3.52	3.09	5.03	4.23
Average OP_rej	11.38	10.98	12.75	1.99	5.75