Proceedings of the 2022 Winter Simulation Conference
B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C.G. Corlu, L.H. Lee, E.P. Chew, T. Roeder, and P. Lendermann, eds.

A HIERARCHICAL DEEP REINFORCEMENT LEARNING APPROACH FOR OUTPATIENT PRIMARY CARE SCHEDULING

Mona Issabakhsh

Lombardi Comprehensive Cancer Center
Georgetown University
2115 Wisconsin Ave, NW
Washington DC, 2007, USA

Seokgi Lee

Rayen School of Engineering
Youngstown State University
One University Plaza
Youngstown, OH, 44555, USA

ABSTRACT

Primary care clinics suffer from high patient no-shows and late cancellation rates. Admitting walk-in patients to primary care setting helps improving clinic's utilization rates and accessibility, therefore, following an efficient walk-in patient admission policy is highly prominent. This research applies a learning-based outpatient management system investigating patient admission and assignment policies to improve the operational efficiency in a general outpatient clinic with high no-show and cancellation rates and daily walk-in requests. Contrary to the general outpatient literature, our results show that only 30% of the walk-in requests should be admitted to minimize the wait time of already admitted patients and providers' over time. Our results also suggest assigning more than 50% of the available slots of a clinic session to punctual patients who have an appointment, to minimize long-run costs. The model and the results, however, are generated based on specific data and parameters, and cannot be directly generalized to other clinics.

1 INTRODUCTION

Although the outpatient appointment scheduling literature is quite rich, most studies are focused on outpatient clinics that do not accept walk-in patients (Qu et al. 2015; Li et al. 2019). Among studies that consider the appointment scheduling systems of the clinics accepting walk-in patients, it is mainly assumed that all walk-in patients must be served, which is not applicable to the clinics admitting walk-in patients selectively. Rather than walk-in patient admission, most studies focus on walk-in and scheduled patients appointment allocation (Li et al. 2021; Fan et al. 2019; Zacharias and Yunes 2020). To the best of our knowledge, only a single study has considered walk-in patient admission problem in a primary care setting (Qu et al. 2015). Qu et al. (2015) developed a finite-horizon Markov decision process (MDP) model to optimize the walk-in patients' admission policy in a primary-care clinic. They developed an MDP model to admit and assign walk-in patients to a single server (doctor), considering patients' wait time and doctors' idle and overtime. They classified the state space into different subsets to solve this problem and investigated each subset's optimal walk-in patient admission policies. Similar to the proposed scheme by Qu et al. (2015), patient admission policies are investigated in this study to improve the operational efficiency in a general outpatient clinic with high no-show and cancellation rates and many walk-in patients. To bridge the research gap, a patient's admission and assignment problem for an outpatient setting with different treatment types is proposed, which is more aligned with the actual function of outpatient clinics. The probability of patient appointment cancellation is considered in the proposed model as a part of state definition, which is not included in Qu et al. (2015) model. The probability of patient appointment cancellation is an essential indicator of possible future empty appointment slots, which should be considered while making patients' admission and assignment decisions. The objective of walk-in patients admission by general outpatient clinics is to reduce the negative impact of patient no-shows and cancellations and improve the clinic's

utilization and accessibility. This study measures clinic utilization by doctors' idle time and overtime. The accessibility to the clinic is indicated and measured by patient wait time and the dissatisfaction of not admitted walk-in patients, which is again not covered by Qu et al. (2015). Lastly, the patient admission and assignment policy is explored in this research considering the entire state and action space using a learning-based general outpatient management system (LGOM), in which a hierarchical deep Q-network algorithm (HDQN) (Issabakhsh 2021) acts as the decision-making core; compared with the solving approach of Qu et al. (2015) in which subsetting the state space is necessary.

LGOM framework is introduced in this research, which can produce online general outpatient management policies. LGOM operates online; that is, it can modify the policy for general outpatient management daily. LGOM is also simulated and trained before implementation to the live service to obtain an intermediate off-line policy. In other words, LGOM is trained with simulation data, and then implemented to the live situation to update the policy further and reflect the real feedback. LGOM is a hierarchical system with two levels, in which the Q-values of each level are updated using a feedforward neural network to find the best policy for complex problems with a high number of states and actions. Hierarchical learning methods break down a long-horizon reinforcement learning (RL) problem into a hierarchy of subproblems, resulting in temporal abstraction and efficient credit assignment over longer timescales. This property makes HDQN-based LGOM a promising approach to scale RL to a long horizon problem, like general outpatient management in which several important decisions with long-run effects should be made in each clinic session (Pateria et al. 2021; Nachum et al. 2019).

2 PROBLEM DEFINITION

In this study, patients are classified into two categories of appointment (app-patients) and walk-in patients. App-patients are scheduled in the current clinic session and arrive on time (before/by) their appointment. In line with the literature and primary care practices, punctual app-patients should have a higher priority for service and must be served in the clinic session that they scheduled an appointment. Therefore, all punctual app-patients waiting at the end of a clinic session must be visited during overtime. Walk-in patients who show up to the current clinic session without an appointment and unpunctual app-patients who miss their appointment slots and show up later in the clinic session, are placed within the second group of patients. Walk-in and unpunctual app-patients have a lower priority for services in primary care clinics. They may be rejected for service in the current clinic session and be asked to make a future appointment (Qu et al. 2015). In this study, walk-in patients and unpunctual app-patients are categorized as walk-in patients. Walk-in patients are either admitted to waiting for services available in the current clinic session or rejected, and all admitted patients should be served during the current clinic session. The rejected walk-in patients are either scheduled with later clinic session appointments or referred to other outpatient clinics (Qu et al. 2015). The flow of the two groups of patients in a general outpatient clinic is illustrated in Figure 1. During the clinic's operating hours, patients arrive for their appointments or cancel their appointments, and walk-in patients arrive randomly. Each clinic session is divided into M equal-length appointment slots, and the walk-in patient admission decisions are assumed to be made at the beginning of each appointment slot. A clinic session in a typical primary care clinic is four hours. The decisions to be made include admitting the walk-in patient who arrived during the previous slot to the walk-in wait queue or not and whether a walk-in patient should be visited in the next slot or an app-patient.

Patients arrive at their appointments and cancel their appointments randomly and independently. The service time per patient is assumed to be constant for each type of treatment since the variation of service times among patients is limited, and most primary care providers try to keep a consistent service time for each patient (Qu et al. 2015; LaGanga and Lawrence 2007; Gupta and Denton 2008). Multiple treatment types are provided to the patients in a clinic, and a single doctor is available in for each treatment type per clinic session. A common assumption in most outpatient clinics, especially primary care clinics, is that providers only see their patients and new patients of the same type, which ensures continuity and quality of care (Cayirli and Veral 2003; Gupta and Denton 2008). Based on this assumption, the walk-in patient

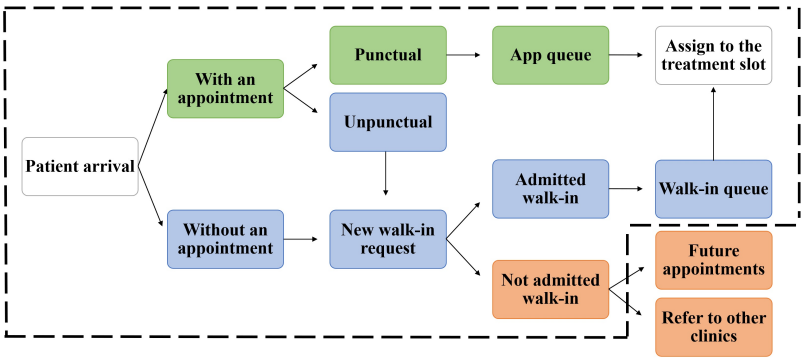


Figure 1: The patient flow in a general outpatient clinic.

admission to one provider is independent of the admission to other providers. Therefore each doctor has a separate queue of the app and walk-in patients in each appointment slot, waiting to be visited.

2.1 System Definition

The LGOM system is explicitly focused on managing patients' admission and allocation during the day. Decisions regarding walk-in patients admissions are made at the beginning of each appointment slot. The time duration between two consecutive decisions is equal, and therefore an MDP model is developed to capture the decision process of admitting patients into the general outpatient clinic.

App and Walk-in Patients often arrive at the system. Assuming that LGOM knows the operating information of the clinic, such as capacity, resources availability, and scheduling congestion level, as well as information on the currently admitted patients, every time a new walk-in patient arrives, LGOM decides whether to accept the request and add the patient to the queue of a specific treatment type or reject (divert) them and ask them to make a future appointment. Due to having a limited number of providers and appointment slots, assigning an available slot to a punctual app-patient or a walk-in patient should also be decided. LGOM may conclude that accepting the walk-in patient request into the system is not profitable for the system (e.g., increasing the congestion, wait time of patients, or overtime of the providers). In that case, it may reject the patient's request, increasing patient dissatisfaction. Such decisions are made based on maximizing the long-run profitability of the system. The sets, parameters, and variables of the model are shown in Table 1.

The MDP model in our problem is composed of M decision stages. Each clinic session is divided into equal-length appointment slots, and decisions are made at the beginning of each appointment slot. $i \in \{1,2,...,M\}$, $m \in \{1,2,...,M\}$ and $j \in \{1,2,...,K\}$ show the index of appointment slot, the decision stage of the clinic session, and the treatment type, respectively. Two variables are defined for decision stage m:

- x_{ij}^m, which is equal to 1, if a patient with an appointment in slot i for treatment type j has arrived at the clinic on time and is ready to be visited in decision stage m and is zero otherwise. $\mathbf{x}_j^m : \{x_{1j}^m, x_{2j}^m, ..., x_{Mj}^m\}$ is the vector showing the queue of app-patients waiting for treatment type j, at the decision slot m.
- y_{ij}^m, which is equal to 1 if a walk-in patient arrived and admitted in slot i for treatment type j is waiting to be visited in decision stage m, and is zero otherwise. $\mathbf{y}_j^m : \{y_{1j}^m, y_{2j}^m, ..., y_{Mj}^m\}$ is the vector showing the queue of walk-in patients waiting for treatment type j in the decision slot m.

The four-hour clinic session is divided into eight 30-min treatment slots, aligned with the literature. Arrays x and y show the app and walk-in patients waiting in the queue to be visited by a provider. The clinic starts admitting walk-in patients from slot 1, and therefore there is no admitted walk-in patient waiting in the

Table 1: Sets, parameters and variables of LGOM.

Item	Description
$V = \{1,...,m,...,M\}$	Set of appointment slots
$N = \{1,...,K\}$	Set of providers
M	The total number of slots
K	The total number of providers
P_j^m	The cancellation probability of future appointments type j in slot m
r_{aj}	The unit revenue of accepting a type j app-patient
r_{wj}	The unit revenue of accepting a type j walk-in patient
c_a	The unit wait cost per slot of app-patients
c_w	The unit wait cost per slot of walk-in patients
c_l	The unit idle cost per slot of providers
c_d	The unit dissatisfaction cost of a walk-in patient
c_o	The unit overtime cost per slot of providers
$\bar{o_j}$	The max overtime slots of treatment type j treatment type
x_{ij}^m	App-patients binary variable
y_{ij}^m	Walk-in patients binary variable
d_{ad}^{mj}	A binary variable which equals to 1, if a type j walk-in patient is admitted in slot m, and is zero otherwise
d_s^{mj}	A binary variable which equals to 1, if a type j app-patient is assigned to slot m, and is zero otherwise
d_w^{mj}	A binary variable which equals to 1, if a type j walk-in patient is assigned to slot m, and is zero otherwise
o_j^m	A non-negative variable which shows the expected number of overtime in decision stage m for type j treatment
r_a^m	A non-negative variable which shows the revenue made by accepting the app-patients in slot m
r_w^m	A non-negative variable which shows the revenue made by accepting the walk-in patients in slot m
c_a^m	A non-negative variable which shows the the wait cost of app-patients in slot m
c_w^m	A non-negative variable which shows the the wait cost of walk-in patients in slot m
c_l^m	A non-negative variable which shows the idle cost of providers in slot m
c_d^m	A non-negative variable which shows the dissatisfaction cost of not admitted walk-in patients in slot m
c_o^m	A non-negative variable which shows the overtime cost of providers in slot m

queue in slot 1. Also, based on the available data, patients arrive for their appointment not earlier than one slot ahead. In line with the literature, a classifier is applied for patients' appointment cancellation prediction, based on the historical data and demographic information of the patients that reserved an appointment (Masselink et al. 2012). We trained the predictive models using patient appointment data from an academic hospital in Central Virginia (Issabakhsh, Lee, and Kang 2021), with mean cancellation rate at 20% (slightly imbalanced classification). Variables such as patients' age, gender, treatment type, health records, and the history of appointment cancellations were applied to develop the classification model. Different classifiers such as k-nearest neighborhood, logistic regression, decision tree, random forest, and gradient boosting were trained, tested, and compared based on the area under the receiver operating characteristic curve (ROC AUC) criteria. Random forest was the best classifier, with the highest ROC AUC in both train and test data, compared with other classifiers. The classifier labels patients as "predicted show" (0) and "predicted no-show" (1). At decision stage m , P_j^m is calculated as the number of patients who are labeled as 1 and scheduled an appointment type j after slot m, who have not arrived at the clinic or have not called to cancel their appointments, divided by the total number of patients that are scheduled an appointment type j after slot m.

The state of the problem in decision stage m is defined as:

$$\mathbf{s}^m(\mathbf{x}_j^m, \mathbf{y}_j^m, P_j^m), \forall m = 1,2,...,M, \forall j = 1,2,...,K$$

Random events in decision stage m including the arrival of the app and walk-in patients, appointment cancellation calls by app-patients, or when app-patients are late to their appointments, result in $\mathbf{s}^m$ to transit to a new state in the following decision stage $(m+1)$. Two types of decisions should be made based on the state of the problem in decision stage m. The first decision is admitting a walk-in patient who arrives

at the clinic between decision stages $m-1$ and m for type j treatment. Binary variable d_{ad}^{mj} takes one if a walk-in patient arrived between decision stages $m-1$ and m is admitted for type j treatment, and zero otherwise:

$$d_{ad}^{mj} = \begin{cases} 1 & \text{if a walk-in patient is admitted in slot } m \\ 0 & \text{otherwise} \end{cases} \quad \forall j \in \{1,2,..,k\}$$

The second decision is about the assignment of slot m to the app-patients and walk-in patients currently waiting in the clinic for type j treatment. Binary variable d_s^{mj} takes 1 if an app patient is assigned to slot m and zero otherwise; d_w^m is also a binary variable that is equal to 1 if a walk-in patient is assigned to slot m and zero otherwise:

$$d_s^{mj} = \begin{cases} 1 & \text{if an app-patient is assigned to slot } m \\ 0 & \text{otherwise} \end{cases} \quad \forall j \in \{1,2,..,k\}$$

$$d_w^{mj} = \begin{cases} 1 & \text{if a walk-in patient is assigned to slot } m \\ 0 & \text{otherwise} \end{cases} \quad \forall j \in \{1,2,..,k\}$$

A provider type j can only visit a single patient per slot, since the duration of an appointment slot is equal to the constant service time per patient, therefore:

$$d_s^{mj} + d_w^{mj} \leq 1, \forall m \in \{1,2,...,M\}, \forall j \in \{1,2,..,k\}$$

The number of overtime slots after each clinic session is limited, for each type of treatment. The expected number of overtime slots in each decision stage can be calculated as follows for each treatment type:

$$o_j^m = (\sum_{i=1}^{m} x_{ij}^m + y_{ij}^m - d_s^{mj} - d_w^{mj}) - P_j^m(M-m), \forall j \in \{1,2,..,k\}$$

The clinic stops admitting walk-in patients in decision stage m if the expected number of overtime slots is larger than the max overtime slot of each treatment type; therefore, the criteria for admitting walk-in patients is $o_j^m \leq \bar{o_j}$.

2.2 Value of a Decision

The reward of each action is composed of the revenue from visiting an app or walk-in patient and the cost associated with an action in a given state, including patients' wait and dissatisfaction costs, and doctors' idle and overtime costs. A linear relationship is considered between the reward and the number of patients visited, in line with the outpatient appointment scheduling literature (LaGanga and Lawrence 2007; Qu et al. 2015). A linear relationship is also considered between patient wait cost and patient wait time, doctors' idle cost and idle time, and the clinic's overtime cost and overtime (Muthuraman and Lawley 2008; Qu et al. 2015; Cayirli and Veral 2003).The immediate reward of a state (decision stage) and action pair is composed of the following items:

- The revenue made by accepting app-patients: $r_a^m = \sum_{j=1}^{k} r_{aj} d_s^{mj}$
- The revenue made by accepting walk-in patients: $r_w^m = \sum_{j=1}^{k} r_{wj} d_w^{mj}$

- The wait cost of app-patients: $c_a^m = c_a(\sum_{j=1}^{k}\sum_{i=1}^{m} x_{ij}^m - d_s^{mj})$
- The wait cost of walk-in patients: $c_w^m = c_w(\sum_{j=1}^{k}\sum_{i=1}^{m} y_{ij}^m - d_w^{mj})$
- The idle cost of the doctors: $c_l^m = c_l(K - (\sum_{j=1}^{k} d_s^{mj} + d_w^{mj}))$
- The dissatisfaction cost of not admitted walk-in patients: $c_d^m = c_d(\sum_{j=1}^{k}(1 - d_{ad}^{mj}))$
- The overtime cost: $c_o^m = max(0, \sum_{j=1}^{k} c_o^j[(\sum_{i=1}^{m} x_{ij}^m + y_{ij}^m - d_s^{mj} - d_w^{mj}) - ((M-m)P_j^m])$

The immediate reward of a state and action pair, therefore, can be calculated as:

$$R^m(s^m, a^m) = r_a^m + r_w^m - (c_a^m + c_w^m + c_l^m + c_d^m + c_o^m)$$

The cost of waiting before the patient's appointment time is ignored in the immediate reward since waiting due to early arrival is voluntarily.

2.3 LGOM Algorithm

LGOM is a general outpatient management system that applies a two-level HDQN core. LGOM can operate in an online manner; that is, it can modify the policy for general outpatient management daily. Besides, it can also be simulated before implementing it to the live service to obtain an intermediate solution. The steps for simulating LGOM to learn the efficient general outpatient management policies are shown in Algorithm I.

Algorithm I:	Simulation of LGOM
Input:	maximum subgoal horizons H_0, H_1, learning rate α
Output:	two trained levels (level 0 and level 1)
1:	**for** N iterations
2:	**Initialize** replay memory D_0 to capacity N_0, and replay memory D_1 to capacity N_1
3:	**Initialize** action-value functions Q_0 and Q_1 with random weights θ_0 and θ_1
4:	**Initialize** target action-value functions $\hat{Q}_0$ and $\hat{Q}_1$ with weights $\bar{\theta}_0$ and $\bar{\theta}_1$
5:	**Select** an arbitrary state (s_1) and an arbitrary goal (g_1) for the highest level (level 1)
6:	**for** M episodes
7:	**for** H_1 attempts or until g_1 is achieved
8:	Sample action a_1 using ε-greedy policy π_1
9:	$s_0 \leftarrow s_1$, $g_0 \leftarrow a_1$
10:	**for** H_0 attempts or until g_0 is achieved
11:	Sample primitive action a_0 using ε-greedy policy π_0
12:	**Update** r_0 using Equation (7)
13:	**if** $g_0 \neq s'$
14:	$r_0 = r_0 - penalty$
15:	**else**
16:	$\gamma_0 = 0$
17:	**if** $g_1 \neq s'$
18:	$r_1 = penalty$
19:	**else**
20:	$r_1 = 0, \gamma_1 = 0$
21:	**Update** the state information
22:	**Store** level 0 transition in D_0, and level 1 transition in D_1
23:	**Sample** random minibatches of transitions from D_0 and D_1
24:	**Set** $y_0 = r_0 + \gamma_0 \max_a \hat{Q}_0(s', g_0, a_0; \bar{\theta}_0)$, and $y_1 = r_1 + \gamma_1 \max_a \hat{Q}_1(s', g_1, a_1; \bar{\theta}_1)$
25:	**Perform** a gradient descent step on $(y_0 - Q(s_0, g_0, a_0; \theta_0))^2$, and $(y_1 - Q(s_1, g_1, a_1; \theta_1))^2$ w.r.t θ_0 and θ_1
26:	**Reset** $\hat{Q}_0$ to Q_0, and $\hat{Q}_1$ to Q_1 every C step

The core of LGOM is a two-level HDQN algorithm in which the Q-values of each level are updated using a feedforward neural network (Issabakhsh 2021). The summary of the steps of LGOM algorithm is as follows:

1. The algorithm starts from the highest level (level 1), and an arbitrary state and goal are selected: s_1, g_1
2. Using ε-greedy policy π_1 action a_1 is sampled and set as the goal for level 0: $g_0 = a_1$

3. Level 0 starts from the current state ($s_0 = s_1$), and its goal ($g_0 = a_1$)
4. Using ε-greedy policy π_0 action a_0 is sampled for level 0 and executed
5. The initial immediate reward (r_0) and the next state (s') are observed
6. The immediate rewards of both levels are updated as follows:

$$r_0 = \begin{cases} r_0 & \text{if } \ s' = g_0 \\ r_0 - penalty & \text{otherwise} \end{cases}$$

$$r_1 = \begin{cases} 0 & \text{if } \ s' = g_1 \\ r_1 - penalty & \text{otherwise} \end{cases}$$

7. The target networks of both levels are updated as follows:

$$y_0 = r_0 + \gamma_0 \max_a \hat{Q}_0(s', g_0, a_0; \bar{\theta}_0)$$

$$y_1 = r_1 + \gamma_1 \max_a \hat{Q}_1(s', g_1, a_1; \bar{\theta}_1)$$

8. A gradient descent step is then performed on the following equations with respect to w.r.t θ_0 and θ_1, respectively:

$$(y_0 - Q(s_0, g_0, a_0; \theta_0))^2$$

$$(y_1 - Q(s_1, g_1, a_1; \theta_1))^2$$

9. When each level either runs out of attempts or achieves its goal state, execution at that level ceases and the level above outputs another subgoal

3 COMPUTATIONAL EXPERIMENTS

A fully connected artificial neural network (ANN) architecture is employed for designing the deep Q-network (DQN) of LGOM in which the total hidden layers are 4. The number of the nodes for each layer is 64, 32, 32, and 2, with a learning rate of 10^{-3}, and the discount factor, the memory size, and the batch size of 0.99, 1500, and 100 respectively. Hyper-parameters of DQN were adjusted by experimenting to achieve both convergence and stability.

Simulation instances are produced using patients information data from an academic hospital in Central Virginia (Issabakhsh, Lee, and Kang 2021), and rates and parameters shown in Table 2. A 4-h clinic session is divided into 15 or 30-minute appointment slots in most primary care outpatient clinics (Green et al. 2006; Qu et al. 2015). Similar to Qu et al. (2015) we consider a 4-hour clinic session, divided into eight 30-min slots. Each provider takes 25–30 minutes to see a patient and use the few remaining minutes of the appointment slot to do the paperwork. Therefore, 4-hour clinic slots, divided into eight 30-minute appointment slots, are considered in this research. Patients who scheduled an appointment may cancel or not show up for their appointments or arrive before or after their appointment time. The no-show and late-cancellation rates in an outpatient clinic can reach 50–55%, based on the literature, and the rate of patients who arrive earlier than their appointment is higher than those who come later (Qu et al. 2015; Lee et al. 2005). Qu et al. (2015) obtained the no-show rate, the late cancellation rate, and the walk-in patient arrival rate of a local primary care clinic based on 6-month historical data of appointments scheduled and patient visits. The arrival pattern of patients with appointments were also approximated based on one-month patient arrival data collected by time studies in the clinic and was classified nto three

groups: early (one-time slot earlier than the corresponding appointment time), on-time (within the time slot before the corresponding appointment time), and late arrivals (after the corresponding appointment time). The percentage ranges of early arrivals, on-time, and late arrivals were 4.84– 5.76%, 70.49–72.58%, and 20.96–24.59%, respectively, among all patients who showed up for their appointments. As shown in Table 2, the average no-show rate is 5.87%, the average late-cancellation rate is 10.59%, and the walk-in patient arrival rate is 1.57 patients per clinic session for one provider. Therefore, the maximum number of walk-in patients between two consecutive appointment slots is considered two, and the maximum number of admitted walk-in patients in each slot is considered one.

In this research, the expected total net reward for patients admission and allocation policy depend on the reward from seeing an app-patient or a walk-in patient (r_a^m and r_w^m), provider's idle cost, and overtime cost per appointment slot (c_d^m and c_o^m), patients wait for costs per appointment slot (c_a^m and c_w^m), and not admitted walk-in patients dissatisfaction cost (c_l^m). Provider idle and overtime costs per slot are estimated based on the median annual compensation per primary-care provider in this study. On average, a provider works 40 hours per week, and the hourly cost of hiring a primary-care provider is about \$90 per hour. Thus, the provider idle cost per slot is \$45. In this study, the provider's overtime cost per slot is assumed \$70, which is around 50% higher than the regular hourly stipend of the provider. The reward per scheduled patient is estimated as the average payment per visit minus the cost per half an hour of hiring a primary-care provider, which equals \$55. Since usually extra effort is needed to take care of walk-in patients, a slightly lower reward of \$50 is considered for seeing a walk-in patient. The dissatisfaction cost of not admitting a walk-in patient is also considered as \$50. The patient wait cost per slot is estimated based on the average hourly wage of about \$19.33, and therefore the waiting cost per slot per app-patient of \$9.7 is used in this study. Walk-in patients arrive without an appointment or are late for their appointments and therefore are considered a lower priority to the clinic. The wait cost per slot per walk-in patient can be treated as a much lighter penalty to service performance than per app-patient. It is assumed that the wait cost per slot per app-patient is four times that per walk-in patient, and therefore, the wait cost per slot per walk-in patient is considered \$2.4.

Table 2: Simulation rates and parameters (Qu et al. 2015).

Parameter	Value
Average no-show and cancellation rate	16.46%
Average walk-in arrival rate	1.57 patient per provider per slot
Average early arrival rate	4.84-5.76%
Average on-time arrival rate	70.49-72.58%
Average late arrival rate	20.96-24.59%
Unit revenue of app visit ($r_a j$)	55
Unit revenue of walk-in visit ($r_w j$)	50
Provider's unit idle cost (c_l)	45
Provider's unit overtime cost (c_o)	70
App-patient unit wait cost (c_a)	9.7
walk-in patient unit wait cost (c_w)	2.4
walk-in patient unit dissatisfaction cost (c_d)	50

3.1 State Goal and Action Sets

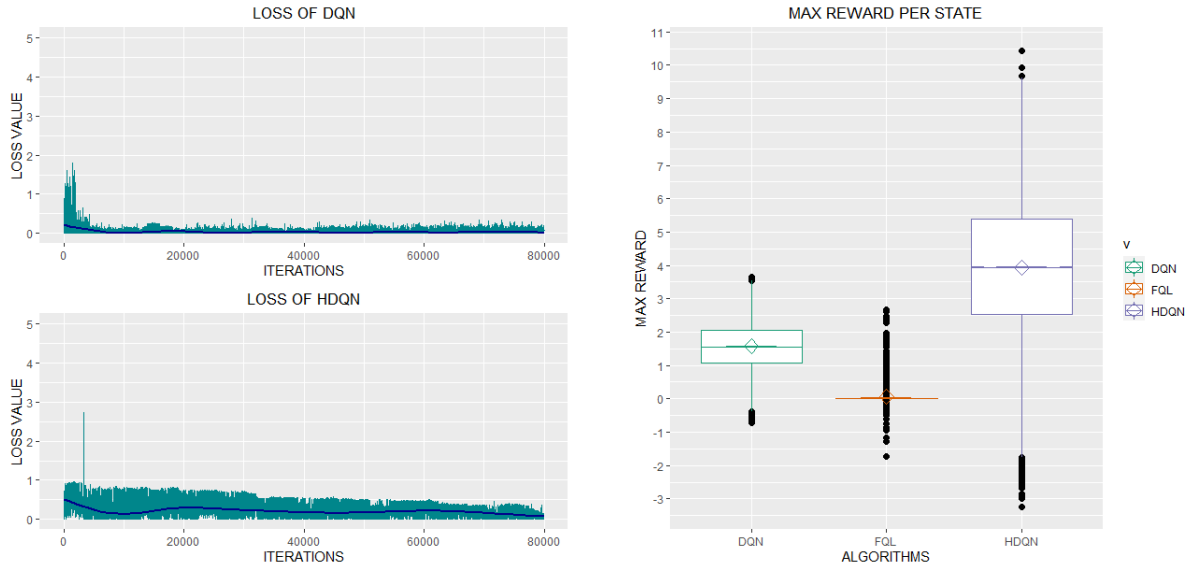
Because each provider works independently of the rest of the providers and only visits its patients, the problem can be set and solved for each treatment type independently and separately. A specific policy may be followed by each provider for a specific type of treatment. The state of the problem is composed of x and y arrays, which are either 0 or 1 (2 levels), and p value, which can be classified as low, medium, or high (3 values). The maximum size of the x array equals the number of slots per clinic session, and the maximum size of the y array equals the number of slots minus one. The number of states per provider is therefore equal to $2^8 \times 2^7 \times 3 = 98,304$. The possible actions in each state is composed of variables d_{ad}^{mj}, d_s^{mj} and d_w^{mj}, each of which has two levels. The number of possible actions is therefore $2^3 - 2 = 6$.

A goal set also needs to be defined for the HDQN algorithm (Issabakhsh 2021). One possible way would be to consider every state as a possible goal; however, since the state size is very large, considering the same goal set as the state set complicates the problem. Based on Gosavi (2015), one remedy to the dimensionality problem is goal aggregation, which means lumping similar goals together and considering them as a single goal. Therefore, to overcome the dimensionality and complexity problem, we define a simple goal aggregation rule in this study. In the aggregated goal set, instead of a binary array of size eight for app-patients and a binary array of size seven for walk-ins, which shows the exact information regarding the patients that are waiting in the queue to be visited, we considered the size of the queue for app and walk-in patients that can be zero to eight, and zero to seven respectively. This way, we almost kept all necessary information in our goal set but decreased its size significantly. The number of the possible goals in this problem therefore is equal to $8 \times 7 \times 3 = 168$.

3.2 Results

3.2.1 Comparison of the Algorithms

HDQN algorithm previously introduced by Issabakhsh (2021) is applied as the decision-making core of LGOM to solve the problem. The HDQN-based LGOM is compared with DQN and regular (flat) Q-learning (FQL) in a problem with 98,304 states and 6 possible actions in each state. FQL only visited 25% of the entire states of the problem during a simulation of 1,600,000 episodes (clinic sessions), each composed of eight slots, which shows the lack of ability of RL in solving this problem. In the following steps, DQN and HDQN algorithms were applied to solve this problem. The loss value is used to show the capability of DQN and HDQN algorithms to solve the problem, which is defined as the squared difference between the target and predicted values in each iteration of the algorithm, calculated as $L_i(\theta_i) = (y - Q(s,g,a;\theta_i))^2$ for iteration i of the algorithm. The left panel of Figure 2 shows the loss values of DQN and HDQN in 80,000 iterations (10,000 episodes) of the algorithms. Both DQN and HDQN managed to converge and reach zero loss, as shown in the left panel of Figure 2. The right panel of Figure 2 shows a comparison



(a) Loss of DQN and HDQN.

(b) Max reward comparison of FQL, DQN and HDQN.

Figure 2: Performance comparision of FQL, DQN, and HDQN algorithms.

between the max reward of each state of FQL, DQN, and HDQN after 80,000 iterations. As it can be seen, HDQN has been capable of finding a policy with a higher reward than both FQL and DQN at convergence.

As stated before, FQL only visited 25% of the total number of states, and therefore the max reward of most states was zero. HDQN outperforms both DQN and FQL in finding a policy with higher max reward at convergence.

3.2.2 LGOM Policy at Convergence

Since HDQN-based LGOM outperforms both FQL and DQN upon convergence, it is recommended to follow the policy by HDQN. In this section, the policy of HDQN-based LGOM is analyzed. Figure 3 shows the breakdown of the actions selected by the policy provided by converged LGOM for different states of the problem, and Table 3 shows a legend of all possible actions in this problem.

Table 3: Possible actions per state.

Action	New Walk-in Admission	App-patient assignment	walk-in Assignment
Action 1	No	No	No
Action 2	yes	No	No
Action 3	No	yes	No
Action 4	yes	yes	No
Action 5	No	No	yes
Action 6	yes	No	yes

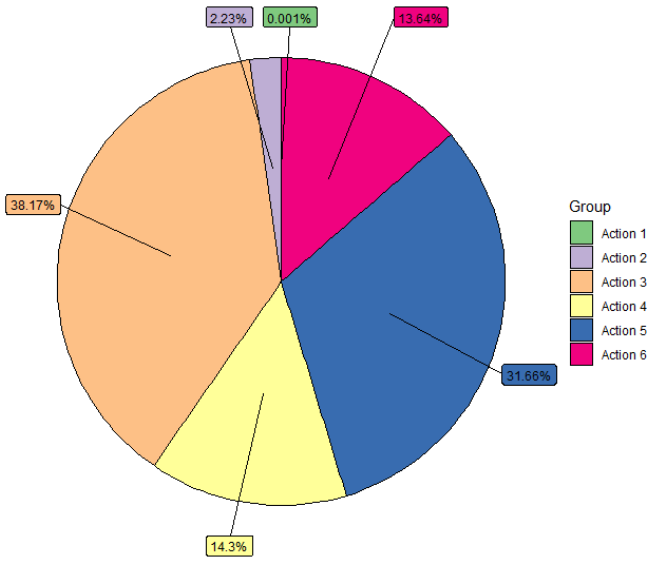


Figure 3: The breakdown of the actions selected by the LGOM policy.

According to the LGOM policy, only in around 2% of the states (actions 1 and 2), assigning a patient (either app or walk-in) to an available slot and provider is not recommended, basically because there is always patients waiting to be visited and the queue is almost never empty, and the cost of an idle server is very high. Based on the policy, in 70% of the states of the problem, it has been decided not to admit the new walk-in request (actions 1,3 and 5). In other words, walk-in patients admission policy suggests that only 30% of walk-in requests should be admitted for visit in a clinical session, basically to minimize patients wait time and provider's overtime. This can be considered as one of the most important insights provided by the LGOM policy , since as discussed before, among general outpatient literature, it is mainly assumed that all walk-in patients must be admitted and visited. LGOM policy simply proves that accepting all walk-in requests to the system is not necessarily profitable for a clinic. Patient assignment policy suggests

assigning more than 50% of the slots to app patients, which makes sense considering that walk-in patients arrive without an appointment or late for their appointments, and therefore are considered lower priority to the clinic. LGOM policy also suggests a significant relationship between the probability of appointment cancellation and walk-in patient admission. Based on the policy , it is better to admit more walk-in patients in a clinical session when the probability of future appointment cancellation is moderate or high, to prevent providers' idle time. Another important relationship that has been considered by the LGOM policy is the length of the queue of app and walk-in patients, and the assignment of app and walk-in patients to the appointment slots. The policy suggests assigning the available slots to the app-patients whenever the queue of the app-patients waiting to be visited is longer compared with the queue of walk-ins.

4 CONCLUSIONS

A patient admission and allocation model for general outpatient setting is proposed in this research, in which the probability of patient appointment cancellation is considered as a part of state definition. LGOM algorithm is introduced in this study which specifically is focused on managing patients admission and allocation in each clinic session. Decisions regarding walk-in patients admissions are made at the beginning of each appointment slot, during each clinic session. An MDP model is developed to capture the decision process of admitting patients into the general outpatient clinic, and assigning them to the available treatment slots. Patients' admission and allocation decisions are made based on the perspective of maximizing long-run profit of the system. LGOM is a general outpatient management system which applies a two-level HDQN decision-making core. LGOM is trained with simulation data then can be implemented to the live situation, to update the policy further.

In this study, we first compared the performance of HDQN-based LGOM with DQN and FQL in a problem with 98,304 states and 6 possible actions in each state. FQL only visited 25% of the total states of the problem during a simulation of 1,600,000 episodes (clinic sessions), each composed of eight slots, which shows the lack of ability of RL in solving this problem. Both DQN and HDQN algorithms on the other side, converged at around 80,000 iterations. HDQN outperformed both FQL and DQN, as it found an policy with higher reward at convergence compared with the other two algorithms.

The policy of HDQN-based LGOM provides the best action to take in each state of the problem. In contrary with the general outpatient literature, which assumes that all walk-in patients must be served, based on LGOM policy, only 30% of the walk-in requests should be admitted, to minimize the wait time of already admitted patients, and provider's overtime. The policy suggests assigning more than 50% of the available slots of a clinic session to app patients, considering the priority of app patients to walk-in patients who arrive without an appointment or late for their appointments. The policy also suggests assigning the available slots to the app patients whenever the queue of the app patients waiting to be visited is longer compared with the queue of walk-ins. Also, based on LGOM policy, more walk-in patients in a clinical session should be admitted when the probability of future appointment cancellation is moderate or high, to prevent providers' idle time.

Lastly, it should be noted that the model and the policy are generated based on specific data and parameters, and therefore the results cannot be directly generalize to other outpatient clinics. The proposed approach, however, can be adopted and rerun in other facilities to generate outpatient admission and allocation policies that satisfy their specific objectives.

REFERENCES

Cayirli, T., and E. Veral. 2003. "Outpatient Scheduling in Health Care: A Review of Literature". *Production and operations management* 12(4):519–549.

Fan, X., J. Tang, C. Yan, H. Guo, and Z. Cao. 2019. "Outpatient Appointment Scheduling Problem Considering Patient Selection Behavior: Data Modeling and Simulation Optimization". *Journal of Combinatorial Optimization* 42:677–699.

Gosavi, A. 2015. *Simulation-based Optimization*. 2nd ed. New York: Springer.

Green, L. V., S. Savin, and B. Wang. 2006. "Managing Patient Service in a Diagnostic Medical Facility". *Operations Research* 54(1):11–25.

Gupta, D., and B. Denton. 2008. "Appointment Scheduling in Health Care: Challenges and Opportunities". *IIE Transactions* 40(9):800–819.

Issabakhsh, M. 2021. *A Simulation-based Optimization Approach for Integrated Outpatient Flow and Medication Management*. Ph.D. thesis, Department of Industrial Engineering, University of Miami, Miami, Florida. https://miami-primo.hosted.exlibrisgroup.com/primo-explore/search?vid=uml_new, accessed 29th September 2022.

Issabakhsh, M., S. Lee, and H. Kang. 2021. "Scheduling Patient Appointment in an Infusion Center: A Mixed Integer Robust Optimization Approach". *Health Care Management Science* 24(1):117–139.

LaGanga, L. R., and S. R. Lawrence. 2007. "Clinic Overbooking to Improve Patient Access and Increase Provider Productivity". *Decision Sciences* 38(2):251–276.

Lee, V. J., A. Earnest, M. I. Chen, and B. Krishnan. 2005. "Predictors of Failed Attendances in a Multi-specialty Outpatient Centre Using Electronic Databases". *BMC Health Services Research* 5(1):1–8.

Li, N., X. Li, C. Zhang, and N. Kong. 2021. "Integrated Optimization of Appointment Allocation and Access Prioritization in Patient-centred Outpatient Scheduling". *Computers & Industrial Engineering* 154:107125.

Li, Y., C. Yang, Z. Hou, Y. Feng, and C. Yin. 2019. "Data-driven Approximate Q-learning Stabilization with Optimality Error Bound Analysis". *Automatica* 103:435–442.

Masselink, I. H., T. L. van der Mijden, N. Litvak, and P. T. Vanberkel. 2012. "Preparation of Chemotherapy Drugs: Planning Policy for Reduced Waiting Times". *Omega* 40(2):181–187.

Muthuraman, K., and M. Lawley. 2008. "A Stochastic Overbooking Model for Outpatient Clinical Scheduling with No-shows". *IIE Transactions* 40(9):820–837.

Nachum, O., H. Tang, X. Lu, S. Gu, H. Lee, and S. Levine. 2019. "Why Does Hierarchy (Sometimes) Work So Well in Reinforcement Learning?". *arXiv Preprint*. https://arxiv.org/abs/1909.10618, accessed 29th September 2022.

Pateria, S., B. Subagdja, A.-h. Tan, and C. Quek. 2021. "Hierarchical Reinforcement Learning: A Comprehensive Survey". *ACM Computing Surveys (CSUR)* 54(5):1–35.

Qu, X., Y. Peng, J. Shi, and L. LaGanga. 2015. "An MDP Model for Walk-in Patient Admission Management in Primary Care Clinics". *International Journal of Production Economics* 168:303–320.

Zacharias, C., and T. Yunes. 2020. "Multimodularity in the Stochastic Appointment Scheduling Problem with Discrete Arrival Epochs". *Management Science* 66(2):744–763.

AUTHOR BIOGRAPHIES

Mona Issabakhsh is a Research Instructor in the Lumbardi Comprehensive Cancer Center at Georgetown University. She earned her Ph.D. in the Department of Industrial Engineering at the University of Miami. Her research interest include simulation-based optimization, machine learning, data analytics, healthcare, and tobacco modeling. Her email address is mi416@georgetown.edu.

Seokgi Lee is an Assistant Professor in Industrial and Systems Engineering at the Youngstown State University. He received his Ph.D. in Industrial and Manufacturing Engineering from the Pennsylvania State University. His research interests include distributed control systems and simulation based optimization for manufacturing supply chain, transportation, and healthcare systems. His email address is slee10@ysu.edu.