Dynamic Capacity Planning of Hospital Resources under COVID-19 Uncertainty using Approximate Dynamic Programming

Elvan Gökalp[a], M. Selim Cakir[b], Hasan Satis[c]

[a]School of Management, University of Bath, Bath, BA2 7AY, UK.; [b]School of Management, University of Bristol;

[c]Gazi University Faculty of Medicine, Besevler, Ankara, TURKEY

ARTICLE HISTORY

Compiled January 11, 2023

Abstract

COVID-19 pandemic has resulted in an inflow of patients into the hospitals and overcrowding of healthcare resources. Healthcare managers increased the capacities reactively by utilizing expensive but quick methods. Instead of this reactive capacity expansion approach, we propose a proactive approach considering different realizations of demand uncertainties in the future due to COVID-19. For this purpose, a stochastic and dynamic model is developed to find the right amount of capacity increase in the most critical hospital resources. Due to the problem size, the model is solved with Approximate Dynamic Programming. Based on the data collected in a large tertiary hospital in Turkey, the experiments show that ADP performs better than a benchmark myopic heuristic. Finally, sensitivity analysis is performed to explore the impact of different epidemic dynamics and cost parameters on the results.

KEYWORDS

stochastic programming; dynamic programming; health services; simulation

Email: eg768@bath.ac.uk

1. Introduction

At the time of writing in late August 2020, the acute respiratory syndrome coronavirus 2 (COVID-19) has affected more than 24 million people worldwide and caused more than 800,000 lives so far. The inflow of the patients led to inefficiency of the available resources and resulted in higher death rates. The governments then behaved reactively and adopted very expensive but quick strategies for treating patients such as building new temporary critical care hospitals or renting private healthcare capacity for buffering. However, these reactive approaches result in a huge burden on public spending as one-time investment and associated running costs. Nevertheless, they are quicker than the usual procedures of increasing capacities, especially for public hospitals, that require long processes of bidding and contracting.

The future of the pandemic is still unknown but second waves in several countries are already observed after the relaxation of the restrictions put in the first place. Governments' decisions on lockdown measures are likely to cause large deviations on the transmission rate and in consequence demand on healthcare. Instead of behaving reactively to the inflow of the patients, the capacity management can be done in a proactive manner considering different possibilities in the future. Using mathematical programming, we can generate optimum policies for all possible scenarios in the future, and thus get prepared for the uncertainties beforehand. Such a proactive approach considers both the normal and fast ways of increasing the capacities while modelling the uncertainties in the demand.

The most vital resources at hospitals during the COVID-19 pandemic are the ventilators and intensive care unit (ICU) beds for the critically ill patients. Both of these resources are quite expensive and production of which are subject to long lead times. Another resource that is needed for moderate COVID-19 patients is separate COVID-19 wards that need to be isolated from the rest of the inpatient wards. Though the beds in COVID-19 wards are not different than normal wards, they require special equipment for the staff and should be separated, thus takes much space. Not only these resources bring extra costs, they also lead to lack of resources and puts a strain on the non-COVID patients' treatments. The UK National Health Service (NHS), for instance, have advised hospitals to postpone or cancel elective surgeries for 3 months through the peak period, resulting 516,000 surgery cancellation (Negopdiev, CovidSurg Collaborative, & Hoste, 2020). Approximately 36000 of these cancellations were cancer-related procedures, delay of which are likely to cause otherwise prevented deaths (Shin et al., 2013).

COVID-19 has undoubtedly created a large backlog in healthcare systems, clearing of which will take a long time due to the shortage of resources. Thus, it is of critical importance to implement capacity management strategies during and after the pandemic in order to clear backlogs in the system quickly and efficiently. Accordingly, this paper proposes a stochastic and dynamic mathematical model to find the best capacity increment policy over a fixed planning period. The proposed policy finds both the right amount of capacity change at each time period and whether the increase should be made in the usual procedures or expensive but quickly.

Our model comprises the capacity management of the three most distressed resources of a hospital during the pandemic: ventilators, ICU beds and COVID ward beds. The uncertainty considered in the model is the patient admissions, discharges and deaths, that are aggregated as net changes of demand for each resource separately. The model aims to minimize the total cost comprising the variable and fixed costs of the capacities as well as the penalty for the lack of enough capacity. Due to the large size of the resulting model, we use an Approximate Dynamic Programming (ADP) method to obtain approximately optimum policies.

An alternative approach for this problem is to model it in a reactive fashion: find the best capacity expansions as the uncertainties (demand) is realized, such as in online optimization. However, such an approach would not be able to consider all the possibilities beforehand, and thus would produce inferior decisions compared to the proactive approach, as considered in this paper. On the other hand, the proactive approach leads to developing a stochastic dynamic programming model that is also called as multi-stage stochastic programming. These problems usually suffer from computational difficulties due to large problem sizes. Applying exact optimization methods, such as classical backward recursion (Powell, 2007), require significant computational efforts, and thus not practical for realistic size instances. Therefore, we utilize ADP, the most popular approximate solution method for these problems (Powell, 2007). This method combines the powers of simulation and optimization and does not require any conditions on the uncertain parameters or the problem structure (e.g. constrained) as in Roohnavazfar, Manerba, De Martin, and Tadei (2019). More details regarding the ADP algorithm are provided in Section 4.

The contributions of the paper can be summarized as:

(1) To the best of our knowledge, we model a significant operational problem related to COVID-19 for the first time,
(2) By developing a computationally efficient solution algorithm, we solve the resulting model, and thus can support the decisions of the authorities,
(3) Based on the computational experiments, we reveal important managerial insights that can reduce the costs of managing the COVID-19 and at the same time improve the patient experience.

2. Literature Review

As expected, the literature on mathematical modelling for decision support regarding COVID-19 is scarce. The extant literature focused on the prediction of the disease dynamics for different countries. For instance, building on the past data of the outbreak, Manca, Caldiroli, and Storti (2020) developed a mathematical model to estimate the ICU bed demand. Other scholars, on the other hand, assessed the impact of alternative mitigation strategies using mathematical modelling (e.g. Ambikapathy and Krishnamurthy (2020); Silva et al. (2020); Van Zandvoort et al. (2020)).

However, to the best of our knowledge, the dynamic capacity planning of a hospital's resources is not studied with mathematical modelling in the literature yet. The healthcare related capacity planning studies are mostly static (Ordu, Demir, Tofallis, & Gunal, 2020) which do not fit well with volatile nature of the COVID-19 pandemic. Static capacity planning problems do not require to use dynamic programming approaches. Thus, they mostly constitute mixed-integer linear or nonlinear programming models. In these models, the computational difficulties due to the large problem size is not an eminent issue as in the dynamic programming models. Therefore, they can be solved with a classical exact method such as branch-and-bound, branch-and-cut, column generation, etc. (Ben Abdelaziz & Masmoudi, 2012; Pehlivan, Augusto, Xie, & Crenn-Hebert, 2012).

Most capacity planning studies for healthcare facilities are static; they aim to find the best capacity levels considering the demand and service rates. Among those static models, location-allocation problem is the most popular one (Ben Abdelaziz & Masmoudi, 2012). In this problem, the purpose is to find where to have the clinical sites among a set of feasible locations and then allocate the patient groups to these facilities to achieve certain service targets. These models are usually integer programming models and static; for example see Santibáñez, Bekiou, and Yip (2009) and Ben Abdelaziz and Masmoudi (2012). Another example for static capacity planning is that of Y. Li, Zhang, Kong, and Lawley (2016) who aim to find the best capacity levels in a network of long-term care facilities. They mostly focus on population dynamics and how much demand would change for each location. Simulation is another method used in the static capacity management problems to especially handle the demand uncertainties.

Being closer to the problem studied in this paper, there are few dynamic capacity management studies within healthcare settings (Hutzschenreuter, Bosman, & La Poutré, 2009; Vermeulen et al., 2009). However, they do not consider the demand uncertainty explicitly as scenarios, but rather utilize queuing formulations to arrive approximate performance measures. Pehlivan et al. (2012) develop a multi-period, dynamic model to determine the acceptable level of capacity to achieve certain service target (e.g. patient rejection probability) in a perinatal network. They have computed the service performance using a queuing theory approach. Although their model is dynamic as in ours, the demand uncertainty is dealt with the queuing theory rather than explicitly included in the model as in the forms of scenarios. In a similar study, Pehlivan, Augusto, and Xie (2014) deal with dynamic capacity update and relocation of services for a perinatal network of hospitals. They also aim to keep the patient rejection probability (modelled as a non-linear function of the service rates and servers) below a certain level for hospitals. They assume that the demand at each time period is known beforehand. By developing a dynamic, integer, linear programming problem, they could solve the instances with Cplex. Jang (2019) develop a similar approach for neonatal care services. They compute the delay probability of a patient using queuing formulations and assume that the demand rates are known.

Possibly the closest paper to ours is that of Akcali, Coˆté, and Lin (2006). They aim to find the optimum changes in the bed capacity of a hospital in a multi-period fashion. However, they also incorporate the demand uncertainty with a queueing formulation that provides the patient waiting times for the beds if the capacity is lower than the demand. They model the problem as a network flow problem where the decisions lead to different states, i.e. nodes of the network. By using existing algorithms for network planning optimization such as Dijktra, they solve the problem. Their model also does not have the delay between acquiring the new capacity and start using it which allowed them to represent the problem as a network flow that would not be possible otherwise.

Dynamic programming studies aforementioned above do not deal with COVID-19 situation. Hence, they have not captured unusual characteristics of the uncertainties brought by COVID-19: the demand is very volatile and behaves in stochastic cycles in the COVID-19 pandemic. Also, the amount of this demand is significant and life-threatening and thus requires frequent and carefully planned capacity increases. Therefore, hospital capacity planning during the COVID-19 pandemic requires significantly different methods than the ones used in the existing studies in the literature. This paper fulfills this gap in the literature by developing a stochastic dynamic programming model and solving it with ADP.

In healthcare related studies, ADP has been mainly used for stochastic and dynamic problems such as surgical scheduling (Astaraky & Patrick, 2015), tactical resource planning for patient admission (Hulshof, Mes, Boucherie, & Hans, 2016), capacity planning/appointment planning of rehabilitation centres (Bikker, Mes, Sauré, & Boucherie, 2020), appointment scheduling (X. Li, Wang, & Fung, 2018; Wang & Fung, 2015), ambulance dispatching and relocation problem (Maxwell, Restrepo, Henderson, & Topaloglu, 2010; Schmid, 2012), overflow problem in inpatient beds (Dai & Shi, 2019). However, to the best of our knowledge, ADP has not been used for dynamic capacity expansion problem of a hospital. Note that ADP is a solution method for stochastic dynamic problems that can also be called as multi-stage stochastic programming. The main motivation to adopt ADP over other possible methods is the computational difficulty of solving the resulting models.

3. Dynamic Capacity Management Model for Critical Hospital Resources

3.1. Problem Description

We consider the capacity management of the most crucial resources in a large tertiary hospital during the COVID-19 pandemic which are (i) ICU beds, (ii) ventilators, and (iii) normal ward beds allocated for COVID-19 patients, that is labelled as 'C-ward' thereafter. As the disease spreads, several patients get directly admitted to the ICU, some of which require ventilators, and mild patients are admitted to the C-ward. Similarly, patients get discharged, move to the C-ward from the ICU or die due to the COVID-19. These changes in the patient demand are aggregated as 'net changes' indicating how much the demand increases (positive) or decreases (negative) for each resource at each week of the planning period, e.g. half a year.

The admissions due to COVID-19 has been high and unpredictable in many countries. Since the hospital capacities were not sufficient to meet the demand, extra and expensive measures have been taken such as renting private hospitals' resources or building new hospitals. To represent these measures in a concrete way, we assume that a hospital has two main options for capacity expansion: (i) buying expensive but quickly available resources or (ii) increasing the resources from the default ways in a time-consuming but comparatively cheaper manner. More specifically, the first (fast) option involves renting these resources from private providers or importing them from the countries that have excess capacities. The default ways include going through the usual procurement process, getting company bids, and putting formal agreements with the company with the best bid.

3.2. Problem Formulation

The problem summarized above is a stochastic and dynamic problem that is modelled as a finite, discrete Markov Decision Process (MDP). The planning horizon is denoted with T, where each time period is $t = 1, \cdots, T$. The number of main resources considered is denoted with I, where each resource is shown with i, such that $i \in \{Ventilators, ICU\ beds, Ward\ beds\}$.

Assumptions:

- The capacity of a resource can be increased with a certain percentage, in the multitudes of u%.
- The capacity considered in this paper is not only physical resources but also comprises of hiring necessary staff along with the resources to satisfy minimum care requirements. These human resources represent the source of a variable cost. The human resources such as nurses cannot be changed so quickly depending on the number of patients. For example, the NHS operates in a way that the necessary number of nurses is computed based on the number of beds; there should be one nurse for at least three beds in an ICU (Hugonnet, Chevrolet, & Pittet, 2007).
- The duration of the delay for a capacity order is independent of the amount of the order. This is due to the fact that the delays are mainly caused by the bureaucratic administrative tasks.
- Since the capacities can only be increased as a percentage of the current capacities, they cannot be increased to a very large value or infinity. On the other hand, we have not assumed that the capacities cannot be increased after a certain level as this was not a real concern in the hospital. However, these type of limitations can be easily added to the model with additional constraints if needed.
- A new order is not possible until the waiting time for the previous one is completed. This assumption is mainly due to the administration concern of the hospital management. These orders are significant and expensive ones and a limited number of human resources make the capacity changes happen, namely the procurement teams and facilities & estates teams. Therefore, instead of putting the new order one week after the previous one, making it when the previous order is realized is a better approach since the delivery dates would be similar, while the later decision is affected by less uncertainty. Because of this, the hospital management discourages the teams to put a new order before the other one is

realized as this leads to ineffectiveness and extra unused capacity.

Uncertainties: The uncertainties in the problem are the net changes in the demands of the ICU beds, ventilators and C-ward beds. The net change is computed as *admissions - deaths - discharges*, and is denoted with $\tilde{z}_{it}$ indicating % of the current number of patients for resource i at period t. Note that an uncertain variable is denoted with a $\tilde{}$ above, while its realization would be without a $\tilde{}$.

State: The state at period t is denoted with $\mathbf{S}_t$ and comprises of $\mathbf{S}_t = \{\mathbf{C}_t, \mathbf{O}_t, \mathbf{w}_t, \zeta_t, \mathbf{k}_t, \mathbf{z}_t\}$, where,

- C_{it} is the capacity of resource i at period t,
- O_{it} is the actual number of patients using resource i at period t,
- w_{it} represents the number of periods passed since the capacity increase is ordered for resource i at period t,
- ζ_{it} is the amount of the order that has not been delivered yet for resource i in period t,
- k_{it} is the form of the order that has not been delivered yet for resource i in period t, and can be either usual or the fast form represented with 1 and 2, respectively.
- z_{it} is the % net change in the demand of resource i at period t.

Action: The action at period t, denoted with $\mathbf{a}_t$ consists of (i) the factor of capacity increase in resource i at period t, denoted with $\Delta_{it} \in \{0, 1, \cdots, \overline{\Delta}\}$, and (ii) the form of capacity increase, denoted with $m_{it} \in \{0, 1, 2\}$, where the usual and fast forms are represented with 1 and 2, respectively. For example, when $u = 15$ and $\overline{\Delta} = 2$, the capacity of a resource can be increased by maximum 30%. When a resource is already waiting for a capacity increase, i.e. $k_{it} > 0$, then capacity increase is not possible for that resource. Thus, the feasible action space can be defined as $\mathcal{A}_t = \{\Delta_{it} \in \{0, 1, \cdots, \overline{\Delta}\}, m_{it} \in \{0, 1, 2\}$ for $i | k_{it} = 0\}$.

Cost: The total cost at period t in state $\boldsymbol{S}_t$ is denoted with ϕ_t that consists of (i) the fixed cost of action, $\boldsymbol{a}_t$, (ii) the variable cost of current capacity $\boldsymbol{C_t}$, and (iii) the penalty due to excess patients and formulated as:

$$\phi(\boldsymbol{S}_t, \boldsymbol{a}_t) = \sum_i c_i^v C_{it} + c_{i,m_{it}}^f \Delta_{it} + c_i^p |O_{it} - C_{it}|^+, \tag{1}$$

where c_i^v, $c_{i,m_{it}}^f$ and c_i^p represent the variable cost of one unit of capacity of resource i, cost of one unit of capacity bought in the form m_{it} and penalty due to each excess patient, respectively. Also, the function $|.|^+$ takes the value inside if that value is positive, and 0, otherwise. There is no cost at the end of the planning period. The end of horizon cost is more applicable to the problems that have a clear finite planning period such as scheduling for a project or surgery scheduling of a day. However, our problem is not such type, in the sense that the capacity planning can continue even after the current planning horizon (in such a case, the whole approach can be repeated for another planning horizon).

State update: The state at period $t+1$, $\boldsymbol{S}_{t+1}$, depends on the state at period t, and the action taken in the current time period, i.e. $\boldsymbol{S}_{t+1}(\boldsymbol{S}_t, \boldsymbol{a_t})$. The capacity is increased when the waiting time for the capacity realization reaches to its limit, denoted with $\overline{w}_{k_{it},i}$. The update on the state variables can be formulated as follows:

$$O_{i,t+1} = O_{it}(1 + z_{it}), \quad \forall i,$$

$$C_{i,t+1} = \begin{cases} \lfloor C_{it}(1 + \zeta_{it} u\%) \rfloor, & \text{if } w_{it} = \overline{w}_{k_{it},i}, \\ C_{it}, & \text{otherwise} \end{cases}$$

$$w_{i,t+1} = \begin{cases} w_{it} + 1 & \text{if } w_{it} < \overline{w}_{k_{it},i}, \\ 0, & otherwise \end{cases}$$

$$\zeta_{i,t+1} = \begin{cases} \Delta_{it}, & \text{if } \Delta_{it} \geq 1 \;\&\; w_{it} < \overline{w}_{k_{it}}, \\ 0 & \text{if } \; w_{it} = \overline{w}_{k_{it}}, \\ \zeta_{it}, & \text{otherwise} \end{cases}$$

$$k_{i,t+1} = \begin{cases} m_{it}, & \text{if } m_{it} \geq 1 \text{ \& } w_{it} < \overline{w}_{k_{it}}, \\ 0 & \text{if } \ w_{it} = \overline{w}_{k_{it}}, \\ k_{it}, & \text{otherwise} \end{cases}$$

Finally, z_{it} is updated as $z_{i,t+1}$, i.e. the exogenous information based on the realization of the demand uncertainty. Note that although we have not modelled the correlation between demand changes of different resources, this is implicitly ensured because the distribution parameters of the demand are estimated from the historical data.

A summary of the model notation is provided in Table 1.

Table 1. Model Notation

Notation	**Decription**
	Indices
i	Resources $i = 1, \cdots, I$,
t	Time periods $t = 1, \cdots, T$.
	Parameters
c_i^v	Variable cost (per time period) of one unit capacity of resource i.
c_{im}^f	Cost of increasing the capacity of resource i by one unit using the mode m.
c_i^p	Penalty cost for each patient that is over the capacity of resource i.
u	% increase in the capacity in one unit of capacity change.
$\overline{\Delta}$	Maximum units of increase in capacity.
$\overline{w}_{mi}$	Delay in capacity expansion for resource i in mode m.
	Variables (Decisions)
Δ_{it}	Factor of capacity increase for resource i in period t.
m_{it}	Form of capacity increase for resource i in period t.
$\mathbf{a}_{it}$	Action at period t for resource i comprising of $\{\Delta_{it}, m_{it}\}$.
$\mathbf{S}_t$	State at period t.
$V(\mathbf{S}_t)$	Value of state $\mathbf{S}_t$.
C_{it}	Capacity of resource i at period t.
O_{it}	Actual number of patients using resource i at period t.
w_{it}	Periods passed since the capacity increase is ordered for resource i at period t.
ζ_{it}	Amount of the order that has not yet delivered for resource i in period t,
k_{it}	Form of the order that has not yet delivered for resource i in period t,
$\tilde{z}_{it}$	Net change in the demand of resource i at period t, denoted as % of the patient level.

Each state, $\boldsymbol{S}_t$ has a value denoted with $V_t(\boldsymbol{S}_t)$ that depends on the value of the possible future states and can be formulated as follows:

$$V_t(\boldsymbol{S}_t) = \max_{\boldsymbol{a}_t \in \mathcal{A}_t} \Big\{ E[V_{t+1}(\boldsymbol{S}_{t+1}|\boldsymbol{S}_t, \boldsymbol{a_t})] - \phi(\boldsymbol{S}_t, \boldsymbol{a}_t) \Big\}, \ \forall \boldsymbol{S}_t, \ \ t = 1, \cdots, T-1, \tag{2}$$

which gives the optimum action that maximizes the subtraction of the action cost from the expected future (state) value. The expected state value is computed first by multiplying the probability of corresponding future state, $P(\tilde{\boldsymbol{z}}_{t+1} = \boldsymbol{z}_{t+1})$, and the estimated value of that state, $V_{t+1}(\boldsymbol{S}_{t+1}|\boldsymbol{z}_{t+1})$, and summing them for all possible $\tilde{\boldsymbol{z}}_{t+1}$. The values of the states in the final periods are assumed to be zero: $V_T(\boldsymbol{S}_T) = 0$.

For the sake of computational efficiency, we assume that $\tilde{\boldsymbol{z}}_t$ can take only a finite number of values, denoted with S. For example, $S = 3$ corresponds to only 3 net demand change scenarios at each time period: low, normal, and high.

4. Solution Approach: Approximate Dynamic Programming

The mathematical formulation outlined in the previous section is computationally difficult to solve due to the large state space. Several state variables, such as current capacities and number of patients, can take a large range of integer values. On the other hand, due to few possible actions and the feasibility conditions, the action space is relatively smaller. Therefore, we use enumeration to find the optimal action at each decision point.

ADP is an algorithm based on forward simulation used to solve large stochastic dynamic programming problems (Powell, 2009). We develop a lookup table based, value iteration ADP algorithm. A linear programming (LP) based ADP is not applied since the value function (2) is complex (Powell, 2009). For the LP-based ADP to work efficiently, the problem should naturally lend itself to a linear problem. However, in our case, there are many non-linear formulations as in the form of *if statements*. Besides, almost all variables in our model are integer, and thus classical LP methods to reduce the problem size are not applicable to our problem. Also, the value iteration is preferred over policy iteration because of the large state space and a comparatively small action set (Sun & Li, 2013).

The probability distributions of net changes, the number of iterations, N, are given as the inputs to the algorithm. The initial state, $\mathbf{S}^0$, is the same in each iteration, and its value $\overline{V}(\mathbf{S}^0_{n=1})$ is initialized as zero in the first iteration. After initialization, two main stages are repeated for each simulated planning horizon: forward and backward passes. In the forward pass, net changes are simulated for each time period and an action is chosen based on value function (2) or randomly based on the probability Γ in the first half of the run, i.e. $n < N/2$. If there are multiple optima, the action is chosen randomly among them. Each visited state, its estimated value and the selected action are entered to a table, called as a lookup table. Note that the costs incurred in the later time periods should be taken into account in an earlier state. To overcome this problem, a backward pass is implemented to update the value function estimations by moving backwards in time in the simulated trajectory in each iteration (Powell, 2007). Algorithm 1 shows the pseudo-code of the ADP algorithm with value iteration, lookup table and double pass.

There are two main value-function related variables: the value function approximations stored in the lookup table, $\overline{V}^n_t$, and the state values computed during the algorithm, v^n_t. In each iteration, after the forward pass, the algorithm goes backward in time and recursively adds the values of the future states (in the sample path) into v^n_t for $t = T-1, \cdots, 1$. If a state $\mathbf{S}^n_t$ is visited for the first time by the algorithm, then its computed value v^n_t is directly added to the lookup table, i.e. $\overline{V}^n_t(\mathbf{S}^n_t) = v^n_t(\mathbf{S}^n_t)$. Otherwise, $\overline{V}^n_t(\mathbf{S}^n_t)$ is computed as the weighted sum of its computed value and the most recent value of the state from the lookup table: $\overline{V}^n_t(\mathbf{S}^n_t) = \alpha_n \overline{V}^{n-1}_t(\mathbf{S}^n_t) + (1-\alpha_n) v^n_t(\mathbf{S}^n_t)$, where α_n is a smoothing parameter. Because the state values are expected to approach their true levels through iterations, α_n is formulated as linearly dependent on the iteration counter n: $\alpha_n = a + b\alpha_n$ where a and b are estimated by trial-and-error for best convergence (Powell, 2007). Since the algorithm does not assume independence of demand for different resources, it is not affected by the internal features of the simulated dataset.

Basis function approximation: The lookup table based ADP algorithm provides an action for the states generated by the simulation. In other words, it does not guarantee to find the right action for each possible state, but for most of them. Therefore,

Algorithm 1 Pseudo-code of the ADP algorithm

Step 0: Initialization: Fix number of iterations N and parameter Γ, initialize $n = 1$, set the value function at initial state as $\overline{V}_1^0(\mathbf{S}_1^0) = 0$.

Step 1. Set $\overline{V}_t^n(\mathbf{S}_k^t) = \overline{V}_t^{n-1}(\mathbf{S}_k^t)$ for $k = 1, \cdots, n-1$ and $t = 1, \cdots, T$.

Step2. Forward Pass:
for $t = 1, 2, \cdots, T-1$, **do**
– Generate $\boldsymbol{z}_t^n$.
– Generate a random number ω and,
if $n \leq N/2$ and $\omega \leq \Gamma$ **then**
Randomly select $\boldsymbol{a}_t^n$ among the feasible action set $\mathcal{A}_t$, and compute $v_t^n(\mathbf{S}_t^n)$ by using (2).
else
– Find the action $\boldsymbol{a}_t^n$ and $v_t^n(\mathbf{S}_t^n)$ by solving (2) based on the state values stored in the lookup table.
– If a state value does not exist in the lookup table, then its value is assumed to be computed based on (1) but without the cost of capacity expansion.
end if
– Update state variables based on the action $\boldsymbol{a}_t^n$ and $\boldsymbol{S}_t^n$: $\mathbf{S}_{t+1}^n = \mathbf{S}_{t+1}(\mathbf{S}_t^n, \boldsymbol{a}_t^n)$.
end for

Step3. Backward Pass:
for $t = T-1, \cdots, 1$ **do**
– Compute $v_t^n(\mathbf{S}_t^n) = v_{t+1}^n(\mathbf{S}_{t+1}^n) - \varphi(\mathbf{a}_t^n, \mathbf{S}_t^n)$, where $\varphi(\mathbf{a}_t^n, \mathbf{S}_t^n)$ is defined as in (1).

if state $\mathbf{S}_t^n$ exists in the lookup table, **then**
– Update $\overline{V}_t^n(\mathbf{S}_t^n) = (\alpha^{n-1})\overline{V}_t^{n-1}(\mathbf{S}_t^n) + (1 - \alpha^{n-1})v_t^n(\mathbf{S}_t^n)$,
else
– Set $\overline{V}_t^n(\mathbf{S}_t^n) = v_t^n(\mathbf{S}_t^n)$.
end if
end for

Step 4. Update iteration number $n := n + 1$. If $n \leq N$, go to *Step 1.* Otherwise, go to *Step 5.*

Step 5. Return all value function approximations ($\overline{V}_t^N$, i.e. lookup table) for $t = 1, \cdots, T$.

we also implement a basis function based ADP algorithm which approximates the state values using a linear weighted formulation of few state variables (Powell, 2007) and the value estimations from the lookup table. After trying several formulations, the best fit for the state-value pairs in the lookup table (with R^2 of 0.88) is achieved by:

$$\overline{V}(\boldsymbol{S}_t) = w_0 + w_1 t + \sum_i w_{2i} |O_{it} - C_{it}|^+,$$

where w_1 and w_{2i} for $i = 1, \cdots, I$ represent the weights of the basis functions. This approximation only considers the extra patients and the current time period, and used for the states not visited by the lookup table ADP. Note that the value of the weight parameters need to be recomputed for each problem instance.

5. Computational Experiments

This section first analyses the computational performance of the ADP algorithm and then presents the results for different scenarios. All the experiments are conducted in Intel Core i7-6700K CPU @ 4.00 Ghz with 32 GB memory and x64 based processor. The ADP algorithm is coded in Matlab 2018b with the following parameters: $\Gamma = 0.6$, $a = 0.05$ and $b = 0.95/n$, where N is 4000. More details regarding the algorithm tuning are provided in the Supplementary material.

Comparison with Exact Algorithm: As mentioned before, we use ADP because it is not possible to solve the problem to optimality within reasonable times for realistic size instances. However, to provide an insight regarding the performance of the algorithm, we implement the backward recursion algorithm to solve the model to optimality for a small instance. This method enumerates all possible states and computes the best action for each one starting from the last period. To be able to solve the problem in a reasonable time, we gradually decreased the problem size by reducing the number of possible scenarios and time periods. First, the mode of possible demand changes in a period is limited to 2 for each resource. The length of planning horizon was dropped to 7 periods which requires to also decrease the waiting times for capacity realizations (such that the capacities are realized before the end of the planning horizon). Finally, the number of possible capacity increase was limited with 1 instead of 2. Even with these reductions, the model has more than 264 million possible states (requiring more than 40GB storage in Matlab). Thus, we had to drop the planning horizon to 6 periods to overcome the memory problem. With efficient coding techniques, we solved this small instance in a couple of hours. The same small instance was also solved with the ADP within a couple of seconds. To understand the performance of the ADP, we computed the difference in the (initial) state values computed with the ADP and the backward recursion algorithm which was less than 1%. This result proves that the ADP produces almost optimum policies.

5.1. Input Data

The pandemic behaves in certain cycles with three main phases: (i) *stability*, where the net change of the demand is stable, (ii) *decline*, where the net change of the demand is negative, and (iii) *escalation*, where the net change of the demand is positive. In any phase, the net changes are still uncertain. For example, in an escalation phase, the net demand changes has a positive mean and a certain standard deviation. The order, duration, means, and standard deviations of different phases can vary between countries. Usually, an escalation phase is much intense and shorter than a decline or a stability phase. Also, the escalation is either followed by a stability or a decline phase depending on the restriction policies in countries.

The case study is based on a large public tertiary hospital in Turkey. The disease dynamics are computed based on the historical trajectory of the pandemic in Turkey starting from the beginning of June, after which detailed data are available (Republic of Turkey Ministry of Health, 2020). Note that the net changes are simulated based on the estimations from the real data, and thus the natural correlation between the demands of different resources are implicitly ensured. Furthermore, the possibility of generating unrealistic cases are prevented with the appropriate coding of the simulation.

Hospital related parameters such as the initial capacities and patient volumes are estimated based on an expert opinion from a hospital in Ankara, Turkey. The fast form of capacity increase involves utilizing the private hospital resources. In the case of overcrowding, the existing resources are shared between patients, such as switching one ventilator between two patients. To deal with the overcrowding in ICU beds, three actions are taken in the following order: (i) delayed discharges from ICU are expedited, (ii) extra beds are requested from other departments until the new orders are realized, and (iii) older beds (stored for these possible extreme scenarios) are used.

Table 2 shows the input data used for the base case. The planning period (horizon) is assumed to be 20 weeks where the duration of one time period is assumed to be one week, i.e. decisions are taken at each week. For the base case, we solve a setting with the following disease trend: stability, increase, and decline. The weekly mean and standard deviation of the net changes in C-ward, ICU, and ventilator demand of the country is computed using the publicly available official reports (Republic of Turkey Ministry of Health, 2020). These distribution related parameters are provided in the Supplementary material. In the second part of the experiments, we analyse the impact of different disease trajectories on the results.

Table 2. Input data used for the base case

Resource	**Ventilators**		**ICU beds**		**C-Ward beds**	
Mode of capacity increase	**Normal**	**Fast**	**Normal**	**Fast**	**Normal**	**Fast**
Waiting times (weeks)	12	4	10	2	7	2
Cost of each additional capacity	20,000	40,000	187,500	360,000	900	1,800
Penalty cost for each extra patient	10,000		5,000		50	
Variable cost of capacity	100		300		10	
Initial capacities	25		50		200	
Initial number of patients	18		40		170	
Unit of capacity increase (u)	15%					
Duration of escalation phase	5					
Duration of decline phase	8					
Duration of stability phase	7					
Probabilities of scenarios	0.25, 0.5, 0.25					

The unit of capacity increase (u) is set to 15%. In other words, the capacity of a resource can be increased by 15% or 30% of the existing capacity or not increased at all. The scenarios of the net demand changes is assumed to be the same for all resources: *low*, *usual*, and *high*.

In addition to the expert opinion, we collected some of the cost parameter levels from publicly available sources. The cost of a ward bed and an ICU bed are estimated from NHS Wales (n.d.) and then checked with the expert for the Turkish case. The cost of establishing a new ICU is estimated from Güngör et al. (2013) which is then updated based on the inflation rate. The fast and normal costs of ventilators are supported by Anadolu Ajansi (2020). We define a very large cost for capacity shortage because it can create life-threatening conditions for COVID-19 patients.

5.2. Results

In this section, we summarize the computational performance of the ADP and analyze a policy generated by the ADP for a single scenario.

Computational Performance of the ADP: The computational time required to obtain the approximate policy with the ADP is around 5 minutes for 4000 iterations. Since the policy is generated for a planning period of half a year, the computation time is acceptable.

As a benchmark solution method to the ADP, we also implement a myopic heuristic that only considers the cost in equation (1) to find the best action, assuming that the capacity expansions are realized immediately, i.e. the demand not satisfied by the capacity is computed based on the expansions. Next, we generate 1000 scenarios and apply the ADP policy as well as the myopic heuristic. The costs by two policies are then compared. In addition to the base case explained in Section 5.1, we also compare two approaches in five other hypothetical cases obtained by varying several parameters from their default levels. First three cases represent different initial hospital capacity and/or utilization rate, i.e. patients/capacities. The last two cases have shorter and longer planning periods than the base case, respectively. The gap between two approaches, computed as *(value of myopic - value of ADP)/value of ADP*, is presented in Table 3 for each case. In the base case, ADP generates 29% less cost compared to the myopic heuristic. In all cases, the myopic approach performs worse than the ADP. The gap between the performances of two methods increases when (i) initially empty capacity is higher, and (ii) the planning horizon is longer. These results suggest that ADP is a more efficient technique than a simple heuristic.

Table 4 shows the policy computed by the ADP for a randomly chosen scenario. The fast and usual forms are shown with 1 and 2, respectively, while the amount of increase represents the multitudes of 15%. Although the initial weeks have a stable pandemic dynamics, the capacity increases are made in the first two weeks to be able to get prepared for an escalation phase in the coming weeks. The ICU beds are increased in a fast form since the penalty for overcrowding in the ICU is quite high. Due to the

Table 3. Comparison of ADP with the myopic approach for various cases

Case	**Explanation**	**Gap between myopic and ADP**
Base	29%	
1	The initial capacities are doubled	-40.89%
2	Both initial capacities and patient levels are doubled	-36.38%
3	The initial capacities are the same, the initial patient levels are halved	-41.58%
4	Planning period is dropped to 20	-17.35%
5	Planning period is increased to 35	-64.33%

low level of the ventilator demand, the capacity is increased once by the usual form, but the amount of increase is 30% to ensure enough fit for the demand.

The capacity of C-ward beds is increased in the fast form at periods 9 and 10, through the middle of the escalation phase, where the net demand for the C-ward beds has increased significantly. However, in most of the periods, the policy generated by the ADP takes a proactive approach instead of a reactive one.

To provide additional insight regarding the decision-making, we analysed the policies obtained for 500 scenarios. For ventilators, the capacity increases were mostly done in period 3, very rarely in period 4 and 5 where the patient demand was higher, i.e. 18 and 19. Also, the capacity increase was mostly double amount (77%), instead of single, while the form of the capacity increase was fast in 72% of the scenarios. This may be because this resource has a higher penalty and longer waiting times than the others which leads to a more cautious decision-making. There was no relationship between the amount of capacity and the form of increase: the % of decisions with fast form was not affected by the amount of capacity in that decision.

For ICU beds, the capacity increase is only done in the initial period and always as double capacity and default form. This may be because this resource has a higher penalty than the third one, and thus the decision-maker is less willing to take the risk of overcrowding in this resource, and thus become cautious. Also, it takes longer to arrive than the third resource.

Table 4. Policy generated by the ADP for a randomly chosen scenario

Time Period	**Net change in demand**			**Amount of increase**	**Form of increase**
	Ventilator	ICU bed	C-ward bed		
1	0	0	0	0,1,2	0,2,1
2	0	2	-12	2,0,0	1,0,0
3	0	2	11	0,0,0	0,0,0
4	0	0	12	0,0,0	0,0,0
5	0	0	13	0,0,0	0,0,0
6	0	0	-14	0,0,0	0,0,0
7	2	4	18	0,0,0	0,0,0
8	0	5	20	0,0,0	0,0,0
9	2	5	37	0,1,0	0,2,0
10	1	6	26	0,0,1	0,0,2
11	2	3	48	0,0,0	0,0,0
12	-1	-7	-46	0,0,0	0,0,0
13	-2	-9	-40	0,0,0	0,0,0
14	-1	-5	-17	0,0,0	0,0,0
15	-1	-2	-32	0,0,0	0,0,0
16	0	-4	-27	0,0,0	0,0,0
17	-2	-4	-35	0,0,0	0,0,0
18	0	-1	-10	0,0,0	0,0,0
19	0	-2	-12	0,0,0	0,0,0
20	-1	-1	-8	0,0,0	0,0,0

We see the highest variation on the capacity decisions of C-ward beds: 7% of the increases are made in period 4, very few on periods 5 and 10, and the rest in the first period. Most of the increases (96%) are single capacity increase which are mostly in the fast form, indicating a rush response to the demand. When the capacity increase is made in a later period, such as 4, mostly it is double amount (no difference in the form), indicating a cautious cover for the expected demand rise in these scenarios. Overall, the decision trends for this resource indicate that mostly an initial capacity increase is needed and enough to cover the demand fluctuations over the planning horizon.

5.3. Sensitivity Analysis

In this section, we analyse the impact of several parameters on the results. For this purpose, we develop a discrete-event simulation model of the problem which applies the ADP policy to 10000 randomly generated scenarios.

Impact of Pandemic Dynamics: The most significant parameter setting is the dynamics of the pandemic, i.e. the order and duration of decline, expansion and stability phases. Therefore, this experiment aims to investigate alternative pandemic dynamics and their impact on the results. Table 5 shows four different pandemic scenarios developed based on different cases around the world. The 'escalation', 'stability' and 'decline' phases are represented with 1, 0, and -1, respectively, in the table. We increase the planning period by 5 weeks in these experiments to allow longer trajectories. These additional 5 weeks are added to the stability phase in the base case. The mean and standard deviation of the net demand changes in each phase is assumed to be the same as those in the base case, which were computed from the Turkish database.

- **Scenario 1:** follows the same trajectory as in the base case, but experiences a second wave (an escalation) instead of a stability. With this scenario, we can examine the impact of a second wave on the results.
- **Scenario 2:** represents a country that is in the beginning of a phase that will decline before a second peak. This scenario is inspired from the Spanish case.
- **Scenario 3:** represents a country that is also in the beginning of a phase that will be placed by a second phase before the first one declines. This situation resembles the US case where the restrictions were not strong in the initial phase and an almost unnoticable decline is experienced before a second wave.
- **Scenario 4:** resembles a country that just started a decline phase and will experience a second wave.

Table 6 shows the *means* ± *standard errors* of the most significant outputs obtained from the simulation using the ADP policy in these scenarios. In the table, *Cost* refers to total cost of the capacity increases during the planning period. *Difference* is the total difference between the patient population and the capacity during the planning period. The *final capacity* is the sum of the resource capacities in the final period. Figure 1 shows the normalized values of those results based on 100 for the base case results (i.e. each output is divided by the respective base case value and multiplied by 100). Note that the figure only provides the normalized means, not standard errors which are negligibly small also the since the value is negative, a normalized value with less than 100 is better.

Table 5. The structure of pandemic trajectory scenarios

Base case					
Phase	0	1	-1	0	
Duration	6	4	8	0	
Scenario 1					
Phase	0	1	-1	1	
Duration	6	4	8	4	
Scenario 2					
Phase	1	0	-1	0	1
Duration	3	2	8	5	4
Scenario 3					
Phase	1	0	1	-1	
Duration	3	8	4	8	
Scenario 4					
Phase	-1	0	1	0	-1
Duration	5	6	4	3	6

Comparing the base case with Scenario 1, we see that the objective value decreases significantly with a second wave. Similarly, the cost as well as the final capacity increase due to higher capacity increases. Additionally, the difference is larger in a second wave scenario possibly because the capacity increases were not enough to satisfy the demand from the second wave. These results show the importance of keeping the safety measures tight and preventing a second wave after the first wave (i.e. Scenario 1).

Table 6. Simulation outputs for different scenarios using ADP

Scenario	**Objective value**	**Cost**	**Difference**	**Final capacity**
Base	-616272.6 ± 643.5	1562.7 ± 1.0	62.2 ± 0.7	321.1 ± 0.1
1	-651047.2 ± 1123.3	1591.4 ± 1.1	85.5 ± 1.1	325.4 ± 0.2
2	-671244.7 ± 1115.2	1572.1 ± 1.1	169.1 ± 1.2	321.3 ± 0.0
3	-789138.3 ± 1519.2	7027.7 ± 21.3	359.8 ± 2.9	364.2 ± 0.1
4	-538600.3 ± 64.5	1509.3 ± 0.7	0.3 ± 0.0	319.4 ± 0.0

Among other scenarios, the worst is Scenario 3, where the second phase starts just after the first one without a decline of the first phase. The hospital experienced a significant overcrowding problem although the capacity is increased in the fast form with maximum amount. Because of this large capacity expansion, the cost is almost 4.5 times of the base case.

The best objective value is achieved in Scenario 4, where the disease trajectory allowed enough time to get prepared for a single wave. Although the amount of

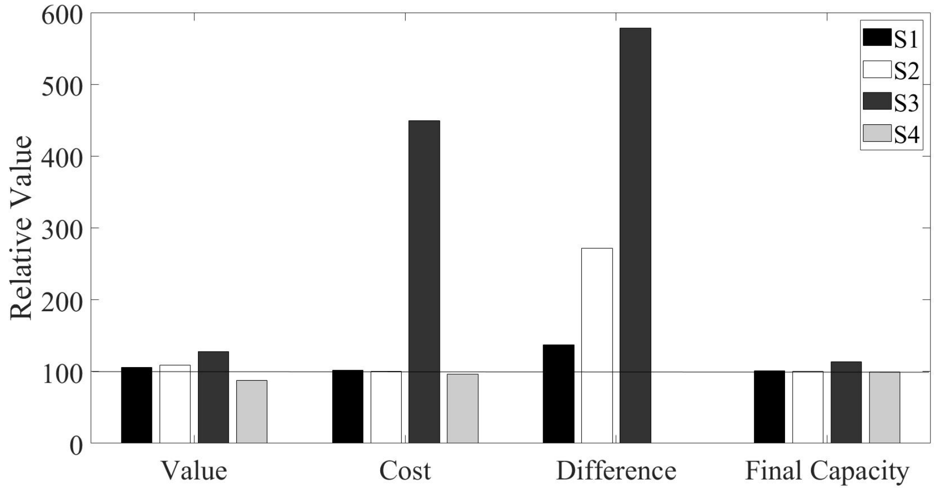

Figure 1. Relative value (compared to 100) of the simulation outputs obtained in different disease scenarios

capacity increases is similar to that in the base case, the overcrowding problem is almost never observed.

Impact of Capacity Increase Unit: The unit of capacity increase, 15% in the base case, significantly depends on the conditions of the country such as the production or service capabilities of the producers and contractors. Thus, in this experiment we examine the impact of this parameter on the results. For this purpose, the unit of increase is set to a smaller (10%) value than the base case, and the ADP policy is obtained again, where the maximum increase (30%) is kept the same. Note that the number of feasible actions increases by 1 in this new setting. The simulation results of the base case and the case with smaller unit increase are shown in Table 7. Figure 2 shows the normalized values of the same outputs (compared to 100% for the base case outputs) as in the previous experiment.

Table 7. Simulation outputs of the base case and with smaller unit increase

Setting	**Value**	**Cost**	**Difference**	**Final capacity**
Base	-616272.6 $\pm$ 643.5	1562.7 $\pm$ 1.0	62.2 $\pm$ 0.7	321.1 $\pm$ 0.1
10% unit change	-603215.7 $\pm$ 700.4	1010.5 $\pm$ 0.7	97.7 $\pm$ 1.0	305.9 $\pm$ 0.2

The results indicate that a smaller unit provides more flexibility to the decisionmakers and leads to a slightly better objective value. The cost decreases significantly although the final capacity does not differ much. This implies that the capacity is increased in the usual form more often compared to the base case. With smaller unit, the overcrowding is higher than the base case possibly because the ward capacity, that has a low penalty, is increased less than the base case. In other words, a smaller unit gives the ADP the flexibility to choose 10% increase for the states where a smaller capacity increase brings a better objective value.

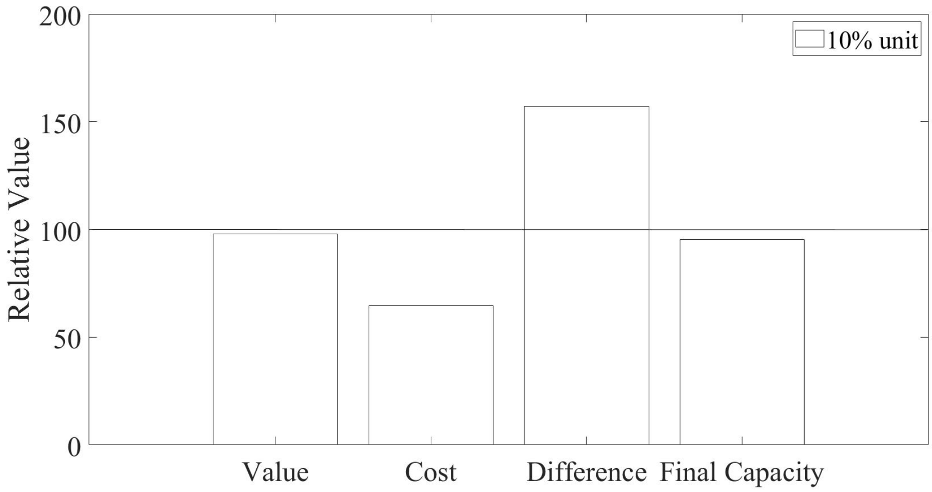

Figure 2. Relative value (compared to 100) of the simulation outputs obtained by 10% unit of change

Impact of Penalty Cost: The level of the penalty for overcrowding is a subjective measure and can differ based on the available solutions in an overcrowding situation, i.e. a safer solution to the overcrowding would mean a lower penalty cost. To investigate such a situation, we solve two additional settings: the penalty cost in the base case is (i) doubled, and (ii) halved for all resources. The simulation outputs are presented in Table 8 and in Figure 3 (in the normalized version as in the previous experiments).

Table 8. Simulation outputs of the base case and different penalty costs

Scenario	Value	Cost	Difference	Final capacity
Base	-616272.6 $\pm$ 643.5	1562.7 $\pm$ 1.0	62.2 $\pm$ 0.7	321.1 $\pm$ 0.1
Doubled penalty	-627516.3 $\pm$ 1016.3	2596.5 $\pm$ 15.9	55.2 $\pm$ 0.8	320.6 $\pm$ 0.1
Halved penalty	-576710.6 $\pm$ 326.2	1510.5 $\pm$ 1.0	81.7 $\pm$ 1.0	319.2 $\pm$ 0.3

Comparing three settings, the final capacities are very close to each other indicating

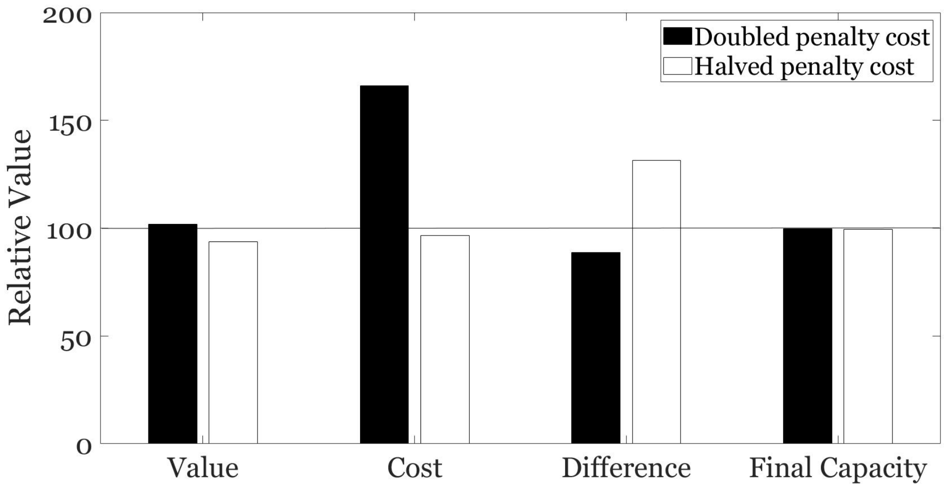

Figure 3. Relative value (compared to 100) of the simulation outputs obtained by different penalty costs

that total amount of capacity increase is the same in all settings. On the other hand, the difference and the cost are quite different. Thus, we can infer that when the penalty is harsher, the policy chooses the fast form more often, rather than increasing the capacity amount. When the fast form is used, there is no need to increase the amount of the order too, since the fast form allows to make a new order fairly soon.

6. Conclusions

Instead of acting reactively to COVID-19 patient inflow, healthcare providers can plan capacity expansions in a more proactive manner that would reduce the patient deaths as well as the expenditures. To obtain proactive capacity policies, we present a stochastic dynamic programming model that considers the uncertainties in the patient demand and the dynamic nature of the actions. The experiments conducted with data collected from various resources indicate that ADP is a quick solution method that generates approximately optimum policies compared to a benchmark method. The experiments with different disease trajectory scenarios indicate the heavy burden of a second wave even when the best policies are applied. On the other hand, if a decline phase is observed before another cycle, the overcrowding is almost not observed at all, while the expenditures are quite low. When the unit of expansion is lowered, the policy has higher flexibility which leads to a better objective value. It suggests that large capacity expansions are not always the best. Also, due to its quick availability, the fast form allows the capacity expansions in smaller amounts which leads to lower under-utilization of the capacity.

References

Akcali, E., Coˆté, M. J., & Lin, C. (2006). A network flow approach to optimizing hospital bed capacity decisions. *Health Care Management Science*, *9*(4), 391–404.

Ambikapathy, B., & Krishnamurthy, K. (2020). Mathematical Modelling to Assess the Impact of Lockdown on COVID-19 Transmission in India: Model Development and Validation. *JMIR Public Health and Surveillance*, *6*(2), e19368.

Anadolu Ajansi. (2020). *AA yerli solunum cihazının üretim aşamalarını görüntüledi.* Retrieved from `https://www.aa.com.tr/tr/bilim-teknoloji/aa-yerli-solunum-cihazinin-uretim-asamalarini-goruntuledi/1821021{\#}`

Astaraky, D., & Patrick, J. (2015, aug). A simulation based approximate dynamic programming approach to multi-class, multi-resource surgical scheduling. *European Journal of Operational Research*, *245*(1), 309–319. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0377221715001332`

Ben Abdelaziz, F., & Masmoudi, M. (2012, apr). A multiobjective stochastic program for hospital bed planning. *Journal of Operational Research Society*, *63*(4), 530–538. Retrieved from `http://dx.doi.org/10.1057/jors.2011.39`

Bikker, I. A., Mes, M. R. K., Sauré, A., & Boucherie, R. J. (2020). Online capacity planning for rehabilitation treatments: An approximate dynamic programming approach. *Probability in the Engineering and Informational Sciences*, *34*(3), 381–405.

Dai, J. G., & Shi, P. (2019). Inpatient overflow: An approximate dynamic programming approach. *Manufacturing & Service Operations Management*, *21*(4), 894–911.

Güngör, G., Karakurt, Z., Adiguzel, N., Moçin, O., Kalamanoglu Balci, M., Saltürk, C., . . . Karahalli, E. (2013, nov). Can the Intensive Care Standards of the

Ministry of Health be Achieved with the Pricing Policy of a Social Security Institution? *Dahili ve Cerrahi Bilimler Yoğun Bakım Dergisi/ Turkish Journal of Medical and Surgical Intensive Care*, *3*, 23–26.

Hugonnet, S., Chevrolet, J.-C., & Pittet, D. (2007). The effect of workload on infection risk in critically ill patients. *Critical Care Medicine*, *35*(1), 76–81.

Hulshof, P. J. H., Mes, M. R. K., Boucherie, R. J., & Hans, E. W. (2016). Patient admission planning using approximate dynamic programming. *Flexible Services and Manufacturing Journal*, *28*(1-2), 30–61.

Hutzschenreuter, A. K., Bosman, P. A. N., & La Poutré, H. (2009). Evolutionary multiobjective optimization for dynamic hospital resource management. In *International conference on evolutionary multi-criterion optimization* (pp. 320–334). Springer.

Jang, H. (2019). Designing capacity rollout plan for neonatal care service system in Korea. *OR Spectrum*, *41*(3), 809–830.

Li, X., Wang, J., & Fung, R. Y. K. (2018). Approximate dynamic programming approaches for appointment scheduling with patient preferences. *Artificial intelligence in Medicine*, *85*, 16–25.

Li, Y., Zhang, Y., Kong, N., & Lawley, M. (2016). Capacity planning for long-term care networks. *IIE Transactions*, *48*(12), 1098–1111.

Manca, D., Caldiroli, D., & Storti, E. (2020). A simplified math approach to predict ICU beds and mortality rate for hospital emergency planning under Covid-19 pandemic. *Computers & Chemical Engineering*, 106945.

Maxwell, M. S., Restrepo, M., Henderson, S. G., & Topaloglu, H. (2010). Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing*, *22*(2), 266–281.

Negopdiev, D., CovidSurg Collaborative, & Hoste, E. (2020). Elective surgery cancellations due to the COVID-19 pandemic: global predictive modelling to inform surgical recovery plans. *British Journal of Surgery*.

NHS Wales. (n.d.). *Delivery plan for the critically ill* (Tech. Rep.). NHS Wales. Retrieved from `http://www.wales.nhs.uk/documents/delivery-plan-for-the-critically-ill.pdf`

Ordu, M., Demir, E., Tofallis, C., & Gunal, M. M. (2020, feb). A novel healthcare resource allocation decision support tool: A forecasting-simulation-optimization approach. *Journal of the Operational Research Society*, 1–16. Retrieved from https://doi.org/10.1080/01605682.2019.1700186

Pehlivan, C., Augusto, V., & Xie, X. (2014). Dynamic capacity planning and location of hierarchical service networks under service level constraints. *IEEE Transactions on Automation Science and Engineering*, *11*(3), 863–880.

Pehlivan, C., Augusto, V., Xie, X., & Crenn-Hebert, C. (2012). Multi-period capacity planning for maternity facilities in a perinatal network: A queuing and optimization approach. In *Automation science and engineering (case), 2012 ieee international conference on* (pp. 137–142). IEEE.

Powell, W. B. (2007). *Approximate Dynamic Programming: Solving the curses of dimensionality* (Vol. 703). John Wiley & Sons.

Powell, W. B. (2009). What you should know about approximate dynamic programming. *Naval Research Logistics*, *56*, 239–249.

Republic of Turkey Ministry of Health. (2020). *COVID-19 Situation Reportss* (Tech. Rep.). Ankara: Ministry of Health. Retrieved from https://sbsgm.saglik.gov.tr/TR,66424/covid-19-situation-report-turkey.html

Roohnavazfar, M., Manerba, D., De Martin, J. C., & Tadei, R. (2019). Optimal paths in multi-stage stochastic decision networks. *Operations Research Perspectives*, *6*, 100124. Retrieved from https://www.sciencedirect.com/science/article/pii/S221471601930096X

Santibáñez, P., Bekiou, G., & Yip, K. (2009). Fraser Health uses mathematical programming to plan its inpatient hospital network. *Interfaces*, *39*(3), 196–208.

Schmid, V. (2012). Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming. *European Journal of Operational Research*, *219*(3), 611–621.

Shin, D. W., Cho, J., Kim, S. Y., Guallar, E., Hwang, S. S., Cho, B., ... Park, J. H. (2013, aug). Delay to curative surgery greater than 12 weeks is associated with increased mortality in patients with colorectal and breast cancer but not lung or thyroid cancer. *Annals of surgical oncology*, *20*(8), 2468–2476.

Silva, P. C. L., Batista, P. V. C., Lima, H. S., Alves, M. A., Guimarães, F. G., & Silva, R. C. P. (2020). COVID-ABS: An agent-based model of COVID-19 epidemic to simulate health and economic effects of social distancing interventions. *Chaos, Solitons & Fractals*, *139*, 110088.

Sun, Y., & Li, X. (2013). Response surface optimisation of surgery start times in a single operating room using designed simulation experiments. *International Journal of Healthcare Technology and Management*, *14*(1), 61–72.

Van Zandvoort, K., Jarvis, C. I., Pearson, C., Davies, N. G., Russell, T. W., Kucharski, A. J., ... Checchi, F. (2020). Response strategies for COVID-19 epidemics in African settings: a mathematical modelling study. *MedRxiv*.

Vermeulen, I. B., Bohte, S. M., Elkhuizen, S. G., Lameris, H., Bakker, P. J. M., & Poutré, H. L. (2009). Adaptive resource allocation for efficient patient scheduling. *Artificial intelligence in medicine*, *46*(1), 67–80.

Wang, J., & Fung, R. Y. K. (2015). Adaptive dynamic programming algorithms for sequential appointment scheduling with patient preferences. *Artificial intelligence in medicine*, *63*(1), 33–40.