Modeling and Optimizing Resource Allocation Decisions through Multi-model Markov Decision Processes with Capacity Constraints

Onur Demiray\textsuperscript{a,}\textsuperscript{*}, Evrim Didem Günes\textsuperscript{b}, Lerzan Örmeci\textsuperscript{a}

\textsuperscript{a}Department of Industrial Engineering, Koç University, 34450 Istanbul, Turkey
\textsuperscript{b}College of Administrative Sciences and Economics, Koç University, 34450 Istanbul, Turkey

Abstract

This paper proposes a new formulation for the dynamic resource allocation problem, which converts the traditional MDP model with known parameters and no capacity constraints to a new model with uncertain parameters and a resource capacity constraint. Our motivating example comes from a medical resource allocation problem: patients with multiple chronic diseases can be provided either normal or special care, where the capacity of special care is limited due to financial or human resources. In such systems, it is difficult, if not impossible, to generate good estimates for the evolution of health for each patient. We formulate the problem as a two-stage stochastic integer program. However, it becomes easily intractable in larger instances of the problem for which we propose and test a parallel approximate dynamic programming algorithm. We show that commercial solvers are not capable of solving the problem instances with a large number of scenarios. Nevertheless, the proposed algorithm provides a solution in seconds even for very large problem instances. In our computational experiments, it finds the optimal solution for 42.86\% of the instances. On aggregate, it achieves 0.073\% mean gap value. Finally, we estimate the value of our contribution for different realizations of the parameters. Our findings show that there is a significant amount of additional utility contributed by our model.

Keywords: Markov decision process, two-stage stochastic integer programming, approximate dynamic programming

1. Introduction

Markov Decision Processes (MDPs) are successfully used to find optimal policies in sequential decision making problems under uncertainty. The application areas of MDPs vary from inventory management, finance, robotics,
telecommunication to humanitarian logistics [3] [9] [24] [26]. It is also extensively employed in medical decision making literature. Practices in this literature range from determining the initiation time of a drug to scheduling patients for a treatment [2] [14] [13] [29]. We refer to [1] for a comprehensive review on the medical practices of MDPs. However, they have certain limitations, as they assume that both transition probability matrices for all actions and the rewards incurred for all state-action pairs are known. In practice, these parameters are generally estimated by maximum likelihood estimators derived using observational data [36]. However, having the right data set and deriving reliable estimators are notoriously hard and it is prone to errors which may cause suboptimal solutions [13] [18] [21] [22] [24] [36] [37]. Hence, in recent years, there are studies on MDP models with uncertain parameters, such as [31]. Another important feature that traditional MDP models generally ignore is the capacity constraints, which has also received increasing attention in the recent years [4] [11] [15]. Our work combines these two emerging areas, as we consider a system with uncertain transition probability matrices and rewards, which operate under a resource constraint.

Robust optimization is extensively employed to deal with parameter uncertainty in MDPs [18] [24] [36]. Robust optimization is one of the widely used techniques for optimization under uncertainty where no probability distribution is assumed for uncertain data. It is based on the worst-case realization of the parameters from a set of alternatives what is so-called uncertainty set [5] [6] [8] [17]. A number of studies approached the parameter uncertainty in MDP problem employing robust optimization with a polyhedral uncertainty set due to its properties allowing tractable solution algorithms [16] [27] [32] [33], whereas another group modeled the problem with the rectangularity assumption which ensures the independence of the rows in transition probability matrices [20] [25] [30]. Wiesemann et al. [34] studied robust MDPs by relaxing this assumption.

Parameter uncertainty in MDPs is also examined with stochastic programming where different types of Bayesian approaches are adopted [24]. In this context, Steimle et al. [31] propose a multi-model MDP (MMDP) approach for a finite-horizon MDP, where transition probabilities and rewards can come from different models. They further claim that having a single policy optimized with respect to all models can be a good idea to capture variations within MDP parameters. An MMDP model can be seen
as a two-stage stochastic programming formulation where different models correspond to different scenarios. Meraklı & Küçükyavuz [24] provided a stochastic programming model for an infinite-horizon MDP model under a risk-averse setting, and applied it to a humanitarian inventory management problem.

Studies focusing on parameter uncertainty in MDPs in the literature address only the unconstrained version of the problem. Nevertheless, some healthcare applications employing MDPs necessitate taking capacity constraints into account due to existence of resource scarcity [4, 11, 15]. Deo et al. [15] proposed a constrained MDP model scheduling patients for a community-based chronic care delivery problem. Ayvaci et al. [4] and Cevik et al. [11] developed constrained MDP and partially observable MDP (POMDP) models of diagnostic and screening decisions for breast cancer. However, none of these studies incorporated parameter uncertainty which may cause suboptimal solutions for such medical resource allocation problems.

With a motivation from medical resource allocation problems, we, in this paper, introduce a capacity constrained, finite-horizon, and finite-state MDP model with uncertain transition probabilities and rewards. In addition to the finite number of non-absorbing states, we consider only one absorbing state to reflect all conditions that stop the decision process. Under the healthcare setting, this absorbing state corresponds to all the terminating conditions such as death, stroke, or hearth failure. The another assumption we make in our model is that the action space consists of two actions, which is consistent considering the nature of the resource allocation problems (allocate versus don’t allocate or accept versus reject). Though we restrict the model to have two actions in each decision epoch, the formulation can be extended easily to the cases with more complicated action spaces. Moreover, the model can represent any capacity constrained problem using MDPs, i.e., its scope is larger than medical resource allocation problems. For example, our approach can be applied to the dynamic product portfolio management problem (Seifert et al. [28]) under a financial constraint, where the probabilities associated with product life cycle transitions are uncertain. The case of a varying level of working capital over different business cycles can also be addressed. Thus the proposed model can be used under different domains spanning from finance, inventory management to marketing.
We adopt a multi-model MDP approach to solve our model. In other words, we provide an extensive-form formulation that is an approximation of the underlying two-stage stochastic integer program with the help of multiple MDP representations, i.e., scenarios. In this respect, our model can be considered as an extension of the model developed by Steimle et al. [31] with capacity constraints. They formulated their models based on the primal linear program (LP) developed for finding optimal policies for an unconstrained MDP. However, we build our formulation based on the dual LP which is the version allowing us to incorporate additional constraints to the MDP framework. To the best of our knowledge, this is the first study combining capacity constraints and parameter uncertainties in MDPs.

The extensive-form formulation, i.e., the deterministic equivalent of the two-stage stochastic programming formulation, can easily become intractable because of capacity constraints, scenarios, and decision epochs. With this in mind, we first prove results leveraging the problem structure, and then develop a parallel approximate dynamic programming algorithm based on these results. This algorithm makes us capable of solving the model efficiently even if we have a large number of scenarios and decision epochs. Lastly, the proposed model and the corresponding solution approaches are applied to a chronic care delivery problem, which also forms the basis for the computational experiments. Through computational studies, we compare different solution approaches. We also measure the value of perfect information and stochastic solution introduced by taking transition probability and reward uncertainties into account. It is worth to mention that all experiments are conducted for a fixed state space which provides a reasonable number of states for our analysis. Thus, our algorithm reflects to the challenges for increasing number scenarios and decision epochs.

Our main contributions can be summarized as follows:

- We introduced a new model incorporating both parameter uncertainty and capacity constraints in MDPs. To the best of our knowledge, this is the first study considering these two dimensions together.

- We prove problem specific structural results, and propose a novel network representation, which together enable a parallel approximate dynamic programming algorithm to solve larger instances of the problem.

- We conduct extensive computational experiments to measure the value
of embracing parameter uncertainty and to compare the solution approaches using the chronic care delivery problem setting as an example.

The remainder of this paper is organized as follows. The model is formally described and mathematically modeled in Section 2. Section 3 explains the proposed solution approach. A comprehensive computational study is provided in Section 4 after introducing the chronic care delivery problem in Section 4. Finally, Section 6 concludes the paper and draws directions for future research.

2. Model Description and Formulation

We consider a capacity allocation problem in a system with \( N \) homogeneous individuals who can be in different states throughout the planning horizon. The actions are taken at the population level to respect a capacity constraint, whereas the individuals’ states evolve depending on their initial states and on the actions. We first describe the process for an individual, and then relate it with the process at the population level.

The state of an individual evolves according to an uncertain transition probability matrix. In addition, the rewards collected are also uncertain. Hence, her evolution can be represented by a Markov decision process with uncertain parameters, which corresponds to a multi-model Markov decision process (MMDP) with the corresponding tuple \((\mathcal{T}, \mathcal{S}, \mathcal{A}, \Omega, \Lambda)\) (see e.g., [31]). Here, \( \mathcal{T} \equiv \{1, \ldots, T-1\} \) is the set of decision epochs and \( \mathcal{T} = \tilde{T} \cup \{T\} \) is the set of all periods; an individual can be in state \( s \in \mathcal{S} \) in time period \( t \), where \( \mathcal{S} \) is the set of all states, \( \tilde{\mathcal{S}} \) is the set of non-absorbing states, and \( D \) is the single absorbing state, so that \( \mathcal{S} = \tilde{\mathcal{S}} \cup \{D\} \); \( \mathcal{A} = \{0, 1\} \) is the set of action space, where \( a = 1 \) corresponds to special service with a limited capacity, and \( a = 0 \) to regular service with unlimited capacity; \( \Omega \) is the set of models or scenarios that specify different transition probability distributions and rewards; and \( \Lambda = (\lambda_\omega)_{\omega \in \Omega} \), where \( \lambda_\omega > 0 \) is the probability that scenario \( \omega \) is in effect and \( \sum_{\omega \in \Omega} \lambda_\omega = 1 \).

Scenario \( \omega \) specifies two transition probability matrices; one among the non-absorbing states, \( \mathcal{P}^\omega \equiv \{P_{ia}^\omega\}_{(i,j) \in \tilde{\mathcal{S}} \times \tilde{\mathcal{S}}, a \in \{0, 1\}} \), and the other from non-absorbing states to the absorbing state \( D \), \( \mathcal{Q}^\omega \equiv \{Q_{ia}^\omega\}_{i \in \tilde{\mathcal{S}}, a \in \{0, 1\}} \). In words, when action \( a \in \mathcal{A} \) is taken in state \( i \in \tilde{\mathcal{S}} \) under scenario \( \omega \in \Omega \), \( P_{ia}^\omega \) is the probability that an individual in state \( i \) moves to state \( j \in \tilde{\mathcal{S}} \), whereas \( Q_{ia}^\omega \)
is the probability that the individual reaches the absorbing state. For each scenario \( \omega \in \Omega \), \( P^\omega \) and \( Q^\omega \) satisfy the following inequalities:

\[
P^\omega_{ij} \geq 0 \quad \forall (i, j) \in \hat{S} \times \hat{S}, a \in A, \omega \in \Omega \\
Q^\omega_{ia} \geq 0 \quad \forall i \in \hat{S}, a \in A, \omega \in \Omega \\
\sum_{j \in \hat{S}} P^\omega_{ij} + Q^\omega_{ia} = 1 \quad \forall i \in \hat{S}, a \in A, \omega \in \Omega
\]

For all non-absorbing states \( i \in \hat{S} \), the reward gained by action \( a \) under scenario \( \omega \) is denoted by \( r^\omega_{ia} \), whereas the reward of being in state \( i \) at the final stage \( T \) is given by \( R^\omega_i \). We further assume that the reward of visiting the absorbing state, \( R^D \), is fixed for all scenarios.

In general, the objective of MMDPs is to find a strategy over the entire planning horizon, denoted \( \Pi \), that performs well with respect to different scenarios accounted in MMDP. One of the widely-used techniques to solve such problems is to employ a two-stage stochastic program, where the first stage determines the strategy \( \Pi \) and the second stage evaluates the performance of \( \Pi \). The discrete set of scenarios called solution sample allows developing a mixed integer program that solves for both stages at the same time, which is also called extensive form of MMDP. Our solution approach also uses this technique; however it has to incorporate the additional capacity constraint at the population level, as we describe below.

The initial population consists of \( N \) individuals, where \( n_i \) of them are in state \( i \) so that \( N = \sum_{i \in \tilde{S}} n_i \). Consequently, the initial distribution of states is specified by \( \theta_i = \frac{n_i}{N} \). The state of the system in period \( t \) is given by \( \mathbf{n}(t) = (n_i(t))_{i \in \tilde{S}} \), where \( N - \sum_{i \in \tilde{S}} n_i(t) \) individuals are in the absorbing state at time \( t \). Individuals receive either regular or special service, where the special service needs additional resources. Hence, the problem is a capacity allocation problem, where scarce resources are rationed among the individuals in different states in each decision period to maximize the total expected reward over \( T \) periods. We represent the scarcity of the resources by a constraint that limits the expected number of individuals who receive special service by \( C_t \) at decision epoch \( t \in \tilde{T} \). This constraint requires tracking the evolution of all individuals through the states as well as of the actions over time.

The system manager decides on the group(s) of individuals who will
receive special service in each period, i.e., she chooses a subset of $\tilde{S}$ who are entitled to use the additional resources under this constraint. Hence, the action space in all states is given by $D = (0, 1)^{|\tilde{S}|}$. We set $\pi^t_i = 1$ if the resource is used for the individuals in state $i$ at decision epoch $t$, and $\pi^t_i = 0$ otherwise, where $i \in \tilde{S}$. The policy in decision period $t$ is then given by $\pi^t = (\pi^t_i)_{i \in \tilde{S}}$, and the strategy over the planning horizon by $\Pi \triangleq (\pi^t)_{t \in \tilde{T}}$. This study aims to find a deterministic strategy $\Pi$ that maximizes the total expected reward over $T$ periods, while satisfying the capacity constraints in each decision period under all scenarios. Hence, the optimal strategy $\Pi$, which is independent of the scenarios, determines the groups that will receive special service in all periods $t \in \tilde{T}$.

To account for the expected rewards and to respect the capacity constraints, we represent the evolution of the system in the model by defining the so-called occupancy measures for each scenario $\omega$:

- $X^t_{\omega i a}$ probability of being in state $i$ under action $a$ at decision epoch $t$
- $Z^t_{\omega}$ probability of visiting the absorbing state at period $t$
- $Y^t_{\omega i}$ probability of finalizing the decision process in non-absorbing state $i$

For a given strategy $\Pi$, the total expected reward is computed as follows:

$$U(\Pi) = N \left[ \sum_{\omega \in \Omega} \lambda_{\omega} \left( \sum_{t \in \tilde{T} \setminus \{1\}} Z^t_{\omega} R^D + \sum_{t \in \tilde{T}} \sum_{i \in \tilde{S}} X^t_{\omega i \pi^t_i} r^\omega_{\pi^t_i} + \sum_{i \in \tilde{S}} Y^t_{\omega i} R^\omega_{\pi^t_i} \right) \right].$$  \hspace{1cm} (2)

The quantity in the brackets corresponds to the expected reward of an individual under all scenarios, which gives the total expected reward of the system when multiplied by the total population $N$. The expected reward of an individual under scenario $\omega \in \Omega$ has three components: Component 1 computes the expected reward over all periods due to visits to the absorbing state. Component 2 accounts for the expected reward of an individual in the non-absorbing states throughout all stages $t \in \tilde{T}$. Finally, component 3 determines the expected reward in the final period, $T$.

At this point, we point out certain characteristics of the problem: (1) Actions $\pi^t_i$ can be either 0 or 1, so that either all individuals in state $i$ use
the additional resources or all are excluded from the special service. We can easily relax this condition to allow partial coverage, as demonstrated in some of our numerical experiments in Section 5. (2) Policy $\pi^t$ changes with respect to time $t$, but not with respect to the state of the system, $n(t)$. More explicitly, we determine the strategy $\Pi$ for the entire planning horizon at time 0 and commit to it till the end of period $T$. This allows us to account for the future effects of our current actions in expectation. Note that in practice, the realized number of individuals to receive the special service may exceed the capacity. However in this paper we aim to support strategic-level decisions, whereas such violations which may be encountered and dealt with at the operational level are ignored. Solving this model on a rolling-time horizon basis by replacing the initial population with the current number of individuals in each period will account for the dynamic change of the system state and ensure that the capacity constraint is never violated.

The mixed integer programming (MIP) formulation demonstrated through (3)-(11) finds $\Pi^*$ which maximizes the total expected reward of the population without violating the capacity constraints for the MMDP model. For the remainder of the paper, we call it (MIP-MMDP). It is the deterministic equivalent of the underlying two-stage stochastic integer programming formulation. In this formulation, $\Pi = \{\pi^t_i\}_{i \in \tilde{S}, t \in \tilde{T}}$ form the first-stage decision variables that need to be made before nature reveals the uncertainty. Hence, they are independent of scenarios. The second-stage problem, on the other hand, determines the values of the occupancy measures after values of the uncertain parameters are realized by the scenarios. As mentioned above, these two stages can be combined in a single mixed-integer program when solving for MMDPs.

\[
\begin{align*}
\max \quad & \mathcal{U}(\Pi) \\
\text{st;} \quad & \sum_{a \in A} x^{\omega,1}_{ia} = \theta_i \quad \forall i \in \tilde{S}, \omega \in \Omega \\
& \sum_{i \in \tilde{S}} \sum_{a \in A} x^{\omega,t-1}_{ia} P^{\omega}_{iaj} = \sum_{a \in A} x^{\omega,t}_{ja} \quad \forall t \in \tilde{T} \setminus \{1\}, j \in \tilde{S}, \omega \in \Omega \\
& \sum_{i \in \tilde{S}} \sum_{a \in A} x^{\omega,T-1}_{ia} P^{\omega}_{iaj} = y^\omega_j \quad \forall j \in \tilde{S}, \omega \in \Omega
\end{align*}
\]
\[ \sum_{i \in \tilde{S}} \sum_{a \in A} X_{ia}^{\omega,t-1} Q_{ia}^{\omega} = \Delta Z_{\omega,t}^{\omega} \quad \forall t \in \mathcal{T} \setminus \{1\}, \omega \in \Omega \quad (7) \]

\[ N \sum_{i \in \tilde{S}} X_{i1}^{\omega,t} \leq C_t \quad \forall t \in \tilde{T}, \omega \in \Omega \quad (8) \]

\[ X_{i1}^{\omega,t} \leq \pi_t^i \quad \forall i \in \tilde{S}, t \in \tilde{T}, \omega \in \Omega \quad (9) \]

\[ X_{i0}^{\omega,t} \leq (1 - \pi_t^i) \quad \forall i \in \tilde{S}, t \in \tilde{T}, \omega \in \Omega \quad (10) \]

\[ \pi_t^i \in \{0, 1\} \quad \forall t \in \tilde{T}, i \in \tilde{S} \quad (11) \]

\[ X_{ia}^{\omega,t} \geq 0 \quad \forall (t, i, a, \omega) \in (\tilde{T}, \tilde{S}, A, \Omega) \quad (12) \]

\[ Z_{\omega,t}^{\omega} \geq 0 \quad \forall t \in \mathcal{T} \setminus \{1\}, \omega \in \Omega \quad (13) \]

\[ Y_{i}^{\omega} \geq 0 \quad \forall i \in \tilde{S}, \omega \in \Omega \quad (14) \]

where \( \Delta Z_{\omega,t}^{\omega} = Z_{\omega,t}^{\omega} - Z_{\omega,t-1}^{\omega} \geq 0 \), and \( Z_{\omega,1}^{\omega} = 0 \), for all \( \omega \in \Omega \).

In (MIP-MMDP), the objective function is given by (3), which maximizes the total expected reward obtained by all individuals, as defined by Equation (2). Equations in (4)-(7) are called forward equations, and they recursively compute the values of the occupancy measures. From a different viewpoint, they balance the probability flows in the model. Furthermore, (8) ensures that capacity constraints are satisfied for each scenario. Particularly, the expected number of individuals entitled to special service, i.e., those in states with action \( a = 1 \), cannot exceed the allowed capacity \( C_t \) at decision epoch \( t \in \tilde{T} \) for each scenario. Constraints (9) and (10) establish the link between the first stage decisions \( \Pi \) and the second stage variables \( X \). Specifically, if the strategy \( \Pi \) does not take action \( a \) for the group of individuals in state \( i \) at time \( t \), then the corresponding occupancy measures has to be equal to 0. Otherwise, they can take values up to 1. Constraints (11) ensure that all individuals in state \( i \) receive either regular or special service.

We now provide two valid inequalities for the (MIP-MMDP) formulation. Their proofs are presented in Appendix A. Proposition 2.1 is a simple consequence of transition probability distributions, while Proposition 2.2 is achieved by aggregating the capacity constraints over all decision epochs.
Proposition 2.1. Hyperplanes in (15) are valid inequalities for (MIP-MMDP).

\[
1 = \begin{cases} 
\sum_{i \in \tilde{S}} \sum_{a \in A} \chi_{\omega,i}^{\omega,t}, & \text{if } t = 1 \\
\sum_{i \in \tilde{S}} \sum_{a \in A} \chi_{\omega,i}^{\omega,t} + Z_{\omega,t}, & \text{if } t \in \{2, \ldots, T-1\} \\
\sum_{i \in \tilde{S}} \chi_{\omega,i}^{\omega,t} + Z_{\omega,t}, & \text{if } t = T
\end{cases} \quad \forall \omega \in \Omega \quad (15)
\]

Proposition 2.2. \( \sum_{t \in \tilde{T}} \left[ \sum_{i \in \tilde{S}} \chi_{i,0}^{\omega,t} + Z_{\omega,t} \right] \geq T - \frac{\sum_{i \in \tilde{T}} \chi_{i,0}^{\omega,t} + Z_{\omega,t}}{n} \) defines a valid inequality for each scenario \( \omega \in \Omega \) for (MIP-MMDP).

3. Solution Approach

Global optima may be obtained in reasonable times for (MIP-MMDP) with small and medium-size problem instances by using commercial solvers. However, reaching the optimal solution(s) for larger instances of the problem is typically beyond the realm of possibility. Nevertheless, the applicability of the model for real settings requires a huge number of parameters such as a great number of scenarios in order to satisfy the convergence of the extensive-form formulation or many decision epochs. Therefore, we propose a parallel approximate dynamic programming (PADP) algorithm providing optimal solutions or very close solutions to global optima as we will show through computational experiments. To that end, we first prove structural results by which we are able to represent our problem in a specific graph structure. Then, we provide the details of the algorithm.

3.1. Structural Properties

Proposition 3.1. Given a strategy \( \tilde{\Pi} \), the values of the occupancy measures under \( \tilde{\Pi} \), denoted by \( (X, Y, Z)^{\tilde{\Pi}} \), are unique.

Proposition 3.1 is a consequence of the fact that when a strategy is fixed, then the problem is reduced to a Markov Reward Process (MRP). That is, \( (X, Y, Z)^{\tilde{\Pi}} \) is a result of the strategy evaluation algorithm illustrated in Algorithm 3 (see Appendix C) which utilizes the forward equations in (4)-(7) to find the corresponding occupancy measures. Algorithm 3 is a polynomial time algorithm in which computations for each scenario are tackled with parallel processing.

10
Corollary 3.1. The feasibility of a given strategy $\hat{\Pi}$ can be determined by Algorithm 3.

Corollary 3.2. Consider two strategies $\hat{\Pi}$ and $\bar{\Pi}$ with the following property: There exists $t^* \in \tilde{T} \setminus \{T-1\}$ such that $\hat{\pi}_t^i = \bar{\pi}_t^i$ for all $i \in \mathcal{S}$, $t \leq t^*$, and $\hat{\pi}_{t+1}^i \neq \bar{\pi}_{t+1}^i$ for some $i \in \mathcal{S}$. Even though $\hat{\Pi} \neq \bar{\Pi}$, the occupancy measures until $t^*$ under $\hat{\Pi}$ and $\bar{\Pi}$ equal each other.

Corollary 3.2 stems from the fact that Algorithm 3 computes the occupancy measures in a forward direction. Thus, any change in policy $\pi^t$ only impacts the occupancy measures related to the stage $t$ and afterwards. This idea is the cornerstone of the proposed algorithm, as it enables us to express the problem in a network structure.

3.2. The Parallel Approximate Dynamic Programming Algorithm

The network can be conceived as a 2-Dimensional topology in which the horizontal axes is constituted by the stages in $\tilde{T}$, whereas all policy combinations, $\Upsilon \triangleq \{1, \ldots, v, \ldots, 2^{|\mathcal{S}|}\}$, form the vertical axes. Then, we generate a node for each element in $\tilde{T} \times \Upsilon$. A node associated with stage $t \in \tilde{T}$ and combination $\upsilon \in \Upsilon$ is represented by $\pi_t(\upsilon)$. Visiting the node $\phi_t \triangleq \{\pi_t(\upsilon)\}_{\upsilon \in \Upsilon}$ shows the set of nodes associated with stage $t \in \tilde{T}$. Then, $\Phi \triangleq \{\phi_t\}_{t \in \tilde{T}}$ becomes the set of nodes in the graph. Arcs take place
only between the nodes of two consecutive stages. An arc from \( \pi_t(v_1) \in \phi_t \) to \( \pi_{t+1}(v_2) \in \phi_{t+1} \) means that the strategy ensures the policy represented by combination \( v_2 \in \Upsilon \) for stage \( t+1 \in \tilde{T} \) while the policy implied by \( v_1 \in \Upsilon \) is accepted for stage \( t \in \tilde{T} \). Then, \( \gamma_t \triangleq \{( \pi_t(v_1), \pi_{t+1}(v_2) ) : (v_1, v_2) \in \Upsilon \times \Upsilon \} \) shows the set of arcs outflowing from the nodes in \( \phi_t \), \( t \in \tilde{T} \setminus \{T-1\} \), and \( \Gamma \triangleq \{ \gamma_t \}_{t \in \tilde{T}\setminus\{T-1\}} \) becomes the set of arcs. Then, our graph can be defined as \( \mathcal{G} = (\Phi, \Gamma) \). A strategy path denoted by \( \sigma \) is nothing but an ordered list of the nodes in \( \Phi \) so that there is exactly one node from \( \phi_t \), \( \forall t \in \tilde{T} \).

Figure 2: A Simple Illustrative Example

Figure 2 provides an illustrative example with 2 non-absorbing states and 5 stages. In this simple example, we have the following set of policy combinations: \( \Upsilon = \{(0, 0), (0, 1), (1, 0), (1, 1)\} \), as we have 2 non-absorbing states. Each possible combination in \( \Upsilon \) implies a policy that can be utilized in any stage. For example, \( v = (1, 0) \) means that we use the resource for state 0 but not for the ones in state 1, i.e., \( \pi(0) = 1, \pi(1) = 0 \). Furthermore, \( \sigma^1, \sigma^2, \) and \( \sigma^3 \) all define a strategic path in \( \mathcal{G} \), equivalently a strategy for the problem. Figure 2 also illustrates how \( \sigma^1, \sigma^2, \) and \( \sigma^3 \) encode strategies.

Our ultimate goal for \( \mathcal{G} \) is to find the strategy path \( \sigma^* \) which provides the maximum total expected reward without violating the capacity constraints. To that end, we should specify the length of an arc in a way that finding the feasible longest strategy path in \( \mathcal{G} \) corresponds to finding \( \Pi^* \). This is the point where we use Corollary 3.2 which increases the efficiency by eliminating the iterations for \( t' = 1, \ldots, t-1 \) for the computation required.
for decision epoch $t$. Then, the length of arc $\iota = (\alpha', \alpha'') \in \Gamma$, denoted by $\eta_\iota$, is defined as follows:

- if $\alpha' \in \phi_1$: Then, $\eta_\iota$ is the sum of the expected rewards obtained in $t = 1$ and $t = 2$ based on the occupancy measures computed for $t = 1$ (let’s say $X_1, Z_1$) and $t = 2$ (let’s say $X_2, Z_2$). $(X_1, Z_1)$ is computed based on $\theta$ and $\pi^1$ implied by $\alpha'$ and $(X_2, Z_2)$ is computed based on $(X_1, Z_1)$ and $\pi^2$ implied by $\alpha''$. Then, we have the following expression:

$$
\eta_\iota = \sum_{\omega \in \Omega} \lambda_\omega \left[ (Z_{\omega,1} + Z_{\omega,2})R_D + \sum_{i \in S} (A_{i,\pi_1^1,i,\pi_1^1}^{\omega_1} + A_{i,\pi_1^2,i,\pi_1^2}^{\omega_2}) \right] \tag{16}
$$

- if $\alpha' \in \phi_t : t \in \{2, \ldots, T - 3\}$: Then, $\eta_\iota$ is the additional expected reward obtained by $(X_{t+1}, Z_{t+1})$ computed based on $(X_t, Z_t)$ and $\pi^{t+1}$ implied by $\alpha''$. Then, we have the following expression:

$$
\eta_\iota = \sum_{\omega \in \Omega} \lambda_\omega \left[ Z_{\omega,t+1}R_D + \sum_{i \in S} A_{i,\pi_{t+1}^1,i,\pi_{t+1}^1}^{\omega_{t+1}} \right] \tag{17}
$$

- if $\alpha' \in \phi_{T-2}$: Then, $\eta_\iota$ is the sum of the additional expected reward obtained by $(X_{T-1}, Z_{T-1})$ computed based on $(X_{T-2}, Z_{T-2})$ and $\pi^{T-1}$ implied by $\alpha''$ and the additional expected reward obtained by $(Y, Z_T)$ computed based on $(X_{T-1}, Z_{T-1})$. Then, we have the following expression:

$$
\eta_\iota = \sum_{\omega \in \Omega} \lambda_\omega \left[ (Z_{\omega,T-1} + Z_{\omega,T})R_D + \sum_{i \in S} A_{i,\pi_{T-1}^1,i,\pi_{T-1}^1}^{\omega_{T-1}} + \sum_{i \in S} Y_{i}^{\omega} R_i \right] \tag{18}
$$

It is important to notice that the length of an arc is not static but changes depending on the previous arc. With the appropriate length definition, our problem is reduced to finding the feasible longest strategy path. In this regard, we develop a parallel approximate dynamic programming algorithm. In theory, the capacity constraints prevent us to claim the optimality. However, we will see that, through computational experiments, we generally get optimal solutions or very close solutions to global optima with the proposed algorithm illustrated in Algorithm 1.
Algorithm 1 Approximate Dynamic Programming Algorithm (PADP)

1: for $v \in \Upsilon$ do
2: $X_{\pi(v)}^{\omega,1}, Z_{\pi(v)}^{\omega,1} \leftarrow \text{Policy Evaluation}(\theta, \pi_1(v)), \forall \omega \in \Omega$
3: $f_{\pi_1(v)} \leftarrow 0$
4: if $\exists \omega \in \Omega : n \sum_{i \in S} X_{i,1}^{\omega,1} > C_1$ then
5: $\phi_1 \leftarrow \phi_1 \setminus \{\pi_1(v)\}$
6: end if
7: end for
8: for $t \in \{2, \ldots, T-1\}$ do
9: for $v \in \Upsilon$ do
10: $f_{\pi_t(v)} \leftarrow \max_{i \in \Xi_{\pi_t(v)}} \{f_i + \eta(i, \pi_t(v))\}$
11: end for
12: end for
Ensure: $\{f_{\pi_t(v)}\}_{t \in T, v \in \Upsilon}$

Algorithm 1 shows the length of the feasible longest sub-strategic path found until the node $\pi_t(v) \in \phi_t$. The algorithm starts with the nodes in $\phi_1$. It computes $(X_1, Z_1)$ by using $\theta$ and $\pi_1(v)$ for each $v \in \Upsilon$. The nodes which provide $X_1$ occupancy measures in a way that the feasibility is violated are removed from $\phi_1$, and the value of the remaining nodes becomes zero. Then, iterations begin in a forward direction for the stages, i.e., $t = 2, \ldots, T - 1$. In each stage $t$, all nodes within $\phi_t$ are visited. For each node $\pi_t(v) \in \phi_t$, the set of nodes that constitute feasible arcs targeting $\pi_t(v)$ form the set $\Xi_{\pi_t(v)} \subseteq \phi_{t-1}$. An arc is feasible if and only if the occupancy measures found for the target node don’t violate the capacity constraints. Then, the value for $\pi_t(v)$, $f_{\pi_t(v)}$, is determined as $f_{\pi_t(v)} \leftarrow \max_{i \in \Xi_{\pi_t(v)}} \{f_i + \eta(i, \pi_t(v))\}$. The algorithm is terminated once all the nodes in $\phi_{T-1}$ are searched. Then, it returns the values found for each node in $G$. The objective function value found by PADP algorithm is determined as the maximum value tackled by the nodes within $\phi_{T-1}$, i.e., $f^{PADP} \triangleq \max_{i \in \phi_{T-1}} f_i$. Lastly, the algorithm performs very fast even if it is working with great number of scenarios and stages thanks to the parallel framework which tackles the scenarios separately.

4. Case Study: Chronic Care Delivery Problem

U.S. National Center for Health Statistics define a chronic condition as a disease which lasts three months or more, and they further claim that there is not any vaccine, medication, or cure that can stop it, or it does not disappear spontaneously. Therefore, it is assumed that recovery is not possible when a patient has a chronic condition. CDC demonstrate stroke,
cancer, and diabetes as major chronic diseases. Bernell and Howard \[7\] further include hypertension, pulmonary conditions, and mental illnesses to the list of chronic diseases.

According to the records of National Health Insurance in the United States, 65% of the population have multimorbidity, i.e., they have more than one chronic diseases. From societal perspective, a huge economic burden, approximately 80% of Medicare spending, is caused by the patients with 4 or more chronic diseases \[35\]. In addition, multimorbidity is more common in disadvantaged groups, which causes more social inequality \[10\]. Consequently, we are witnessing a changing paradigm called chronic care management where polychronic patients’ physical, mental and social needs are managed in one center. As in all systems, this approach has trade-offs since this system requires both fixed and variable costs. In this regard, the effective use of this resource is important. Therefore, the question of which patients should be targeted for chronic care delivery under capacity constraints becomes important. Here we will assume a hypothetical chronic care management system with a constrained resource. In this setting, the patient classes that will be eligible for the service for each period in the planning horizon should be determined at the system design stage.

The most distinctive and challenging part of designing a system for polychronic patients is that the underlying mechanism for the progress of health conditions may significantly vary from disease to disease. In the case of multimorbidity, the number of diseases, their severity, and which combinations of diseases occur can vary dramatically for each patient. For instance, a patient may have diabetes with type 2 and breast cancer, whereas another patient may have hypertension, asthma, and a cardiovascular disease. In this perspective, it is necessary to deal with transition probability and reward uncertainties, considering the range of chronic conditions, their breakdowns, and various levels of severity for the diseases.

Patient targeting problem for chronic care delivery model is a suitable case study to test our model with its capacity constraints and inherently appeared parameter uncertainty within the system. In this regard, the associated MDP model is introduced in Section 4.1.
4.1. MDP Model

In this section, we conceptualize the dynamics of a patient in regard of chronic care delivery. In the rest of this section, we explain the components of the underlying MDP model.

4.1.1. States

A state involves two components. The first one is *health status* which can be, without loss of generality, any measure indicating the physical well-being of a patient. In this study, we discretized this component into 3 categories: *simple*, *moderate*, and *complex*. The severity of conditions is getting worse from simple to complex.

The second component of a state includes the *engagement* of a patient due to the statistics revealing that about 50% of premature deaths are caused by behaviors that could be changed [12]. Moreover, [23] states that chronic diseases have very close link with lifestyle habits such as excessive alcohol use, poor nutrition, lack of physical activity, and tobacco use. In this regard, the term engagement corresponds to all behavioral aspects required to maintain patients’ own physical well-being. It is discretized by two alternatives: *low* and *high*. The engagement of patients increases from low to high.

In summary, we have six states coming from the pairwise combination of the physical and behavioral components: Low-Simple (0), Low-Moderate (1), Low-Complex (2), High-Simple (3), High-Moderate (4), High-Complex (5). In addition to these states, we also include the state *death* (6) as an absorbing state.

4.1.2. Actions

For each patient with a health status and a behavioral characteristic, an agent chooses either *normal care* or *special care*. Thus, the action space consists of two elements: 0, for normal care and 1, for special care. The action space is not binding for those reaching to the absorbing state.

4.1.3. Transition Probabilities and Immediate Rewards

Transition probabilities are subject to uncertainty due to highly combinatorial nature of multimorbidity. Hence, it is not possible to have data representing all possible transitions to obtain significant statistical inferences. Even if we have such data, the estimations are prone to large statistical
errors because of limited or missing data for each possible transition. Nevertheless, it is possible to approximate the probability distributions by using expert opinions. We demonstrate one approach to employ these opinions. In this respect, we first generate hierarchical rules, i.e. relations, among transition probabilities based on the prior beliefs (see Appendix B). These are the rules that can be generated by any business analyst who is familiar with the system. Then, the base model the so-called nominal model assumes that the parameters take the values the Monte Carlo algorithm depicted in Algorithm 2 converges to. The point where Monte Carlo converges may be seen as a center of the polyhedral set in which our nominal model for transition probabilities is likely to lie. Then, all scenarios are generated based on this center by adding uniform random noises around it. A similar approach is carried out for the immediate rewards with their own rules. Detailed description of scenario generation is explained in Section 5.1.

Algorithm 2 Monte Carlo Approach for Transition Probability Estimation

Require: number of iterations, expert opinions
1: nIteration ← 1
2: \( P \leftarrow \emptyset \)
3: \( Q \leftarrow \emptyset \)
4: for \( nIteration \leq \) number of iterations do
5: \( P'' \leftarrow \) A random realization which satisfies the required rules
6: \( Q'' \leftarrow \) A random realization which satisfies the required rules
7: \( \bar{P} \leftarrow \bar{P} \cup \{P''\} \)
8: \( \bar{Q} \leftarrow \bar{Q} \cup \{Q''\} \)
9: \( nIteration \leftarrow nIteration + 1 \)
10: end for
11: \( \hat{P} \leftarrow \text{mean}(\bar{P}) \)
12: \( \hat{Q} \leftarrow \text{mean}(\bar{Q}) \)
Ensure: \( \hat{P}, \hat{Q} \)

Our Monte Carlo algorithm starts with empty set of patients. Then, a patient following the hierarchical rules is randomly generated in each iteration until reaching to the desired number of iterations. Then, the parameters of the nominal model are concluded as the mean of the corresponding parameters of the patients generated through the simulation.
5. Computational Experiments

In this section, we conduct computational experiments\footnote{All experiments are conducted on a computer with an Intel® Core™ i7-6500U 2.39 GHz processor and 64 GB of RAM, with the Windows 10 operating system.} to obtain an understanding of (i) the contribution of the proposed model; (ii) how effective our solution approach is. In this context, Section 5.1 explains the way problem instances are generated. Then, Section 5.2 is devoted to investigate the impact of valid inequalities, whereas Section 5.3 examines the computational aspects of the proposed algorithm. Next, Section 5.4 analyzes the value of perfect information after Section 5.5 studies the value of stochastic solution. Finally, Section 5.6 provides managerial insights for potential practitioners with a special focus on the price of fairness, the value of flexibility, and capacity management.

5.1. Instance Generation

For chronic care delivery problem, it is not possible to have longitudinal data capturing all possible transitions in sufficient numbers. This is due to the large number of chronic conditions each of which has a variety of types. Furthermore, patients may have different numbers of chronic conditions of numerous severity. In this respect, it becomes an unrealistic expectation to have reliable time-series data of patients as in specialized services such as diabetes, HIV, and cancer treatments. Nevertheless, we have some prior beliefs about the ranks among the uncertain parameters. These are not complicated medical consequences, but simple justifiable facts allowing us to come up with a polyhedral uncertainty set for the nominal model. Point estimations, what is so-called nominal model, are generated through Monte Carlo approach depicted in Algorithm 2. Then, different scenarios are generated by adding uniform noise to the nominal model. We refer to Hörman et al. (2013) and Zhang et al. (2017) for different scenario generation methodologies \cite{19, 37}.

Let $\bar{\omega}_x$ denotes the nominal value of uncertain parameter $x$, whereas $\omega_x$ denotes the corresponding parameter value in scenario $\omega \in \Omega$. Given nominal model $\bar{\omega}$, scenario $\omega$ is generated as follows:

$$\omega \triangleq \left\{ \omega_x : \omega_x \sim UNIF\left( (1 - \epsilon)\bar{\omega}_x, (1 + \epsilon)\bar{\omega}_x \right) \right\}, \quad \epsilon \in (0, 1) \quad (19)$$
Then, elements of transition probability matrices are normalized so that they satisfy the conditions in (1a)-(1c).

A problem instance is uniquely characterized by 4 parameters. (i) $|\Omega|$, the number of scenarios in the solution sample; (ii) $T$, the number of decision epochs; (iii) $c = C/n$, the proportion of the population that can utilize the resource at a particular decision epoch; (iv) $\epsilon$, the maximum allowed variation from the nominal model. Without loss of generality, we assume that the capacities allocated per decision epoch are equal. Then, a problem instance with the given parameters is denoted by $I(|\Omega|, T, c, \epsilon)$. Furthermore, prior probabilities are assumed to be equal. That is, the probability for a new patient to start the decision process in a state is equal for each non-absorbing state in $\tilde{S}$, i.e., $P\{s_1 = i\} = \frac{1}{|\tilde{S}|}, \forall i \in \tilde{S}$. Lastly, the reward obtained after visiting the absorbing state, $r^D$, is determined as zero.

5.2. Impact of Valid Inequalities

There are three components affecting the computational complexity of the problem: the number of decision epochs, scenarios, and states. In the case study, i.e., chronic care delivery problem, the size of the state space is fixed which is not problem since our objective is to embrace the problems with a large number of scenarios and/or stages. Therefore, we generated problem instances for a varying number of scenarios and decision epochs. To this end, we created 35 problem instances, each with the values of $c = 0.4$ and $\epsilon = 0.25$. While generating them, we use the set of different numbers of scenarios $\{5, 10, 25, 50, 100, 250, 500\}$ and the set of various numbers for decision epochs $\{5, 10, 20, 30, 40\}$.

In order to prepare an experiment analyzing the impact of the valid inequalities, we first solve (MIP-MMDP) by using the well-known commercial solver CPLEX 12.10 for all problem instances. Then, each valid inequality is separately included to the formulation, and solved again by the solver. All computational runs are imposed a time limit of 4 hours. In this time horizon, the solver could find the optimal solutions only for 21 instances among the 35 instances we generated. Table 1 provides required solution times in seconds and relative percentage gap values for each instance and setting. We compare the required solution times for the 21 instances where the optimal solutions could be found, whereas the relative percentage gap values are compared for the rest of the instances.
Table 1: Impact of Valid Inequalities in 35 Problem Instances

| nScenario | Time (sec) | Gap (%) | Time (sec) | Gap (%) | Time (sec) | Gap (%) |
|-----------|------------|---------|------------|---------|------------|---------|
| 5         | 0.2        | 0       | 0.247      | 0       | 0.965      | 0       |
| 10        | 2.371      | 0       | 2.36       | 0       | 2.538      | 0       |
| 20        | 8.241      | 0       | 8.95       | 0       | 7.27       | 0       |
| 30        | 10.31      | 0       | 9.99       | 0       | 10.818     | 0       |
| 40        | 9.63       | 0       | 9.788      | 0       | 10.232     | 0       |
| 10        | 0.225      | 0       | 0.221      | 0       | 0.216      | 0       |
| 10        | 6.8        | 0       | 6.845      | 0       | 8.441      | 0       |
| 20        | 53.848     | 0       | 80.211     | 0       | 116.207    | 0       |
| 30        | 101.423    | 0       | 162.751    | 0       | 132.885    | 0       |
| 40        | 138.449    | 0       | 157.163    | 0       | 170.913    | 0       |
| 25        | 0.511      | 0       | 0.532      | 0       | 0.47       | 0       |
| 10        | 256.734    | 0       | 229.86     | 0       | 251.09     | 0       |
| 20        | 1891.068   | 0       | 1515.171   | 0       | 1368.60    | 0       |
| 30        | 1425.902   | 0       | 1193.413   | 0       | 1783.105   | 0       |
| 40        | 3015.04    | 0       | 2818.656   | 0       | 3052.309   | 0       |
| 50        | 9.664      | 0       | 9.486      | 0       | 9.846      | 0       |
| 10        | 3643.571   | 0       | 1176.563   | 0       | 2355.285   | 0       |
| 20        | 14401.456  | 0.6     | 14401.213  | 0.9     | 14401.537  | 0.8     |
| 30        | 14401.767  | 0.5     | 14439.982  | 0.01    | 14402.806  | 0.6     |
| 40        | 14470.37   | 1.1     | 14429.526  | 1.4     | 14430.794  | 1.2     |
| 100       | 106.115    | 0       | 196.078    | 0       | 96.072     | 0       |
| 10        | 9669.398   | 0       | 13778.265  | 0       | 14425.731  | 0.1     |
| 20        | 14401.549  | 2.0     | 14406.66   | 1.9     | 14409.823  | 2.3     |
| 30        | 14421.458  | 2.1     | 14406.278  | 2.1     | 14432.652  | 1.9     |
| 40        | 14406.975  | 1.8     | 14434.273  | 2.1     | 14411.393  | 1.9     |
| 250       | 193.304    | 0       | 339.134    | 0       | 272.081    | 0       |
| 10        | 14406.464  | 5.4     | 14407.621  | 5.4     | 14409.304  | 4.8     |
| 20        | 14408.738  | 6.4     | 14405.341  | 6.0     | 14405.035  | 5.3     |
| 30        | 14407.866  | 5.7     | 14407.235  | 6.1     | 14403.145  | 5.5     |
| 500       | 742.221    | 0       | 861.041    | 0       | 1103.164   | 0       |
| 10        | 14407.241  | 4.5     | 14402.248  | 5.4     | 14406.788  | 5.3     |
| 20        | 14406.308  | 7.0     | 14407.461  | 7.1     | 14410.242  | 6.9     |
| 30        | 14406.978  | 7.7     | 14405.62   | 7.6     | 14406.41   | 7.7     |
| 40        | 14404.374  | 8.9     | 14409.774  | 8.9     | 14407.159  | 9.2     |

When we focus on the instances with zero gap values\(^2\), we realize that valid inequality 1 provides the best solution time in 33.3% of the instances, whereas this statistic becomes 15.15% for valid inequality 2. In the case where we look at the remaining instances, valid inequality 2 provides the best gap values for almost half of the instances, whereas valid inequality 1 provides it only for 2 problem instances.

\(^2\)Valid inequality 1 and 2 refer to Propositions 2.1 and 2.2 respectively.
5.3. Computational Performance of the Parallel Approximate Dynamic Programming Algorithm

Table 1 shows that commercial solvers are not capable of solving problem instances with the desired number of scenarios for convergence. Therefore, an efficient algorithm providing solutions of high quality in a short span of time is essential for the applicability and the value of the developed model. In this context, a high quality solution refers to the ones that are optimal or very close to it and found within an acceptable amount of time.

In Section 3, a parallel approximate dynamic programming algorithm is proposed as an alternative to solvers to provide an efficient algorithm required for the model. In this section, we test its capabilities in terms of time and optimality. In accordance with this purpose, we first use the 21 problem instances for which the solver could provide the optimal solutions in Section 5.2. The proposed algorithm, PADP, is run for these problem instances, and the corresponding solution times in seconds and the percentage gap values are presented through Table 2. PADP finds the optimal solutions for 42.857% of the problem instances. For the remainder of them, the maximum deviation from the optimal solution is only 0.24% which is quite acceptable for an algorithm trying to find an approximate solution. When we focus on the aggregate level performance metrics, we see an outstanding approximation so that the mean percentage gap value is only 0.073%.

Another remarkable fact about the superiority of the algorithm is that it achieves these approximations in considerably short span of times. It finds the solution about 1000 times faster than the solver.

In Table 3, we compare the capabilities of CPLEX 12.10 and PADP for the instances the solver could not find the optimal solution in 4 hours in Section 5.2. Recall that the solver loses its power for the instances with a large number of scenarios. The column improvement(%) shows the relative percentage improvement provided by PADP. A positive value in the column shows that the algorithm provides a better solution than the solver with 4 hours time limitation. On the contrary, the negative value indicates that the solution found by the solver in 4 hours provides a better objective function than the algorithm. Results suggest that the algorithm produces better results than the solver when the number of scenarios exceeds 250, i.e., for more realistic instances. The amount of improvement becomes more significant as the number of scenarios increases.
Table 2: Computational Comparison of PADP and CPLEX 12.10 for Small-Medium Problem Instances

| Instance | Time (MIP) | Time (PADP) | GAP(%) |
|----------|------------|-------------|--------|
| Z(5, 5, 0.4, 0.25) | 0.2 | 0.084 | 0 |
| Z(5, 10, 0.4, 0.25) | 2.371 | 0.204 | 0 |
| Z(5, 20, 0.4, 0.25) | 8.241 | 0.683 | 0.13 |
| Z(5, 50, 0.4, 0.25) | 10.31 | 1.127 | 0.16 |
| Z(5, 50, 0.4, 0.25) | 9.63 | 1.567 | 0.16 |
| Z(10, 5, 0.4, 0.25) | 0.225 | 0.115 | 0 |
| Z(10, 10, 0.4, 0.25) | 6.8 | 0.36 | 0 |
| Z(10, 20, 0.4, 0.25) | 53.848 | 1.041 | 0.02 |
| Z(10, 30, 0.4, 0.25) | 101.423 | 1.742 | 0.05 |
| Z(10, 40, 0.4, 0.25) | 138.449 | 2.437 | 0.06 |
| Z(25, 5, 0.4, 0.25) | 0.511 | 0.149 | 0 |
| Z(25, 10, 0.4, 0.25) | 256.734 | 0.466 | 0 |
| Z(25, 20, 0.4, 0.25) | 1891.068 | 1.332 | 0.14 |
| Z(25, 30, 0.4, 0.25) | 1425.692 | 2.248 | 0.2 |
| Z(25, 40, 0.4, 0.25) | 3015.04 | 3.176 | 0.22 |
| Z(50, 5, 0.4, 0.25) | 9.664 | 0.17 | 0.02 |
| Z(50, 10, 0.4, 0.25) | 3643.571 | 0.555 | 0.24 |
| Z(100, 5, 0.4, 0.25) | 106.115 | 0.245 | 0 |
| Z(100, 10, 0.4, 0.25) | 9669.398 | 0.802 | 0.13 |
| Z(250, 5, 0.4, 0.25) | 193.304 | 0.304 | 0 |
| Z(500, 5, 0.4, 0.25) | 742.221 | 0.434 | 0 |

| Instance | Time (MIP) | Time (PADP) | GAP(%) |
|----------|------------|-------------|--------|
| min   | 0.2 | 0.084 | 0 |
| max   | 9669.398 | 3.176 | 0.24 |
| mean  | 1013.563 | 0.916 | 0.073 |

Table 3: Computational Comparison of PADP and CPLEX 12.10 for Large Problem Instances

| Instance | CPLEX Time(sec) | CPLEX Gap(%) | PADP Time(sec) | Improvement(%) |
|----------|-----------------|--------------|----------------|----------------|
| Z(50, 20, 0.4, 0.25) | 14401.456 | 0.6 | 1.629 | -0.33 |
| Z(50, 30, 0.4, 0.25) | 14401.767 | 0.5 | 2.699 | -0.36 |
| Z(50, 40, 0.4, 0.25) | 14470.37 | 1.1 | 4.025 | -0.37 |
| Z(100, 20, 0.4, 0.25) | 14404.549 | 2 | 2.251 | 0.05 |
| Z(100, 30, 0.4, 0.25) | 14421.458 | 2.1 | 3.635 | 0.01 |
| Z(100, 40, 0.4, 0.25) | 14406.975 | 1.8 | 4.9 | -0.14 |
| Z(250, 10, 0.4, 0.25) | 14405.094 | 1.9 | 0.98 | 0.09 |
| Z(250, 20, 0.4, 0.25) | 14406.464 | 5.4 | 3.005 | 0.66 |
| Z(250, 30, 0.4, 0.25) | 14404.798 | 6.4 | 5.658 | 0.87 |
| Z(250, 40, 0.4, 0.25) | 14406.866 | 5.7 | 7.724 | 0.13 |
| Z(500, 10, 0.4, 0.25) | 14407.241 | 4.5 | 1.502 | 0.62 |
| Z(500, 20, 0.4, 0.25) | 14406.308 | 7 | 5.662 | 1.13 |
| Z(500, 30, 0.4, 0.25) | 14406.978 | 7.6 | 8.255 | 1.48 |
| Z(500, 40, 0.4, 0.25) | 14405.674 | 8.9 | 10.203 | 2.66 |
5.4. Value of Stochastic Solution

If agents do not take parameter uncertainty into account then the actions will become sub-optimal. With this in mind, in this section, we investigate the additional value of incorporating stochastic solution. In other words, we estimate the potential loss in objective function value, if we don’t consider the transition probability and reward uncertainties. There is certainly a strong relation between the value of stochastic solution and the way we generate the problem instances. The main assumption we make during the generation of data is the maximum variation from the nominal model. Thus, we conduct sensitivity analysis for different values of $\epsilon$. Particularly, we choose three values of $\epsilon$ for the experiments: 0.10, 0.25, and 0.50.

We also postulate uniform distribution when we consider the deviations from the nominal model, and by force of uniform distribution, it is not possible to observe any realization for the parameters outside of the sphere determined by $\epsilon$. However, observing lots of outliers which lie in the exterior region of the sphere is very likely, considering the large number of combinations within the nature of the problem. Thus, the problem includes more randomness than our problem instances. Nevertheless, there are two advantages of this approach. Firstly, the value of stochastic solution asserted for these problem instances intuitively provides a lower bound for the actual value since we don’t allow outliers. Therefore, if we find a considerable value of stochastic solution then we can assert that the actual value is expected to perform higher than it, which makes our contribution realistic. The second advantage is that this way of generating data materializes itself without loss of generality. That is, results associated with other experiments about the complexity of the model and the efficiency of our algorithm are not affected by the assumptions made during the generation of the problem instances.

Expected value of stochastic solution, $\text{EVSS}(\%)$, under solution sample $\Omega = \{1, \ldots, \omega, \ldots, |\Omega|\}$ is computed as follows:

$$\text{EVSS}(\%) \triangleq \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \frac{f_{\text{MIP-MMDP}} - f_{\omega}(\Omega)}{f_{\omega}(\Omega)} \times 100$$

(20)

where $f_{\text{MIP-MMDP}}$ is the total expected reward of the optimal strategy found for $\Omega$, i.e., the optimal objective function value of (MIP-MMDP) for $\Omega$, and $f_{\omega}(\Omega)$ is the total expected reward of following the optimal strategy for scenario $\omega$, where the realization of nature is still explained by all scenar-
ios in \( \Omega \). It is clear that \( f^\omega(\Omega) \) yields a lower bound for \( f^{\text{MIP-MMDP}} \), i.e., \( f^\omega(\Omega) \leq f^{\text{MIP-MMDP}}, \forall \omega \in \Omega \) due to the sub-optimal behavior of the strategy governed by a single scenario. Thus, \( \frac{f^{\text{MIP-MMDP}} - f^\omega(\Omega)}{f^\omega(\Omega)} \) shows the relative loss in the objective function value if agents consider only scenario \( \omega \in \Omega \) as a deterministic model. Then, \( \text{EVSS}(\%) \) gives the expected value of stochastic solution. Appendix D shows how \( f^\omega(\Omega) \) is evaluated for scenario \( \omega \in \Omega \).

Table 4: Expected Value of Stochastic Solution for Different \( \epsilon \) and \( T \) Values with 200 Scenarios

| \( \epsilon \) | nStage | Value of stochastic solution(%) |
|-----------|--------|-------------------------------|
| 0.1       | 5      | 2.97                          |
|           | 10     | 4.52                          |
|           | 20     | 4.47                          |
|           | 30     | 4.53                          |
| 0.25      | 5      | 4.03                          |
|           | 10     | 5.70                          |
|           | 20     | 6.83                          |
|           | 30     | 7.31                          |
| 0.5       | 5      | 3.95                          |
|           | 10     | 5.89                          |
|           | 20     | 7.80                          |
|           | 30     | 8.30                          |

Table 4 shows the expected value of stochastic solution for different values of \( \epsilon \) and \( T \) when the number of scenarios is 200. Results suggest that there is a significant value of incorporating stochastic solution. Expected value is increasing as the value of \( \epsilon \) increases. Another important observation is that it also increases with the growing number of decision epochs which implies that our approach matters more for long-term problems.

5.5. Value of Perfect Information

Value of perfect information (VPI) is defined as the maximum cost a policy-maker is willing to pay in order to obtain the perfect information about the uncertain parameters. Since we are dealing with a maximization problem, the objective function of single scenario deterministic problem, denoted by \( f^\omega \), becomes an upper bound for \( f^{\text{MIP-MMDP}} \), i.e., \( f^{\text{MIP-MMDP}} \leq f^\omega, \forall \omega \in \Omega \). Then, the expected VPI (EVPI) under the set of scenarios \( \Omega \) is computed as follows:

\[
\text{EVPI} = \frac{\sum_{\omega \in \Omega} f^\omega}{|\Omega|} - f^{\text{MIP-MMDP}}
\]  

(21)
Then, the percentage expected VPI, denoted by EVPI(%) is defined as
\[
EVPI_{\text{MIP-MMDP}} \times 100.
\]
Table 5 illustrates EVPI(%) for different values of \( \epsilon \) and \( T \) when the solution sample consists of 200 scenarios. We compute EVPI(%) for each pairwise combinations of \( \epsilon = 0.1, 0.25, 0.5 \) and \( T = 5, 10, 20, 30 \). Results suggest that the average EVPI(%) value is 7.88\%. They also indicate that EVPI(%) increases with the increasing number of \( \epsilon \) and \( T \).

Table 5: Expected Value of Perfect Information for Different \( \epsilon \) and \( T \) Values with 200 Scenarios

| \( \epsilon \) | nStage | Value of perfect information(%) |
|--------------|--------|---------------------------------|
| 0.1          | 5      | 1.72                            |
|              | 10     | 2.14                            |
|              | 20     | 1.84                            |
|              | 30     | 1.76                            |
| 0.25         | 5      | 5.62                            |
|              | 10     | 7.09                            |
|              | 20     | 5.81                            |
|              | 30     | 5.17                            |
| 0.5          | 5      | 12.14                           |
|              | 10     | 16.36                           |
|              | 20     | 17.66                           |
|              | 30     | 16.92                           |

5.6. Managerial Insights

In this section, we investigate managerial aspects of applying (MIP-MMDP). In this context, we first examine the price of fairness, that is, the loss in total expected reward caused by ethical issues. Then, we focus on the value of having flexibility in operations. Lastly, we analyze the impact of different capacity levels. It is important to notice that each part, except the one measuring the effect of different capacity levels, requires certain modifications in the model. For these parts, we change the required components of the model, and then solve them with the solver. That is, we do not modify our algorithm for the new cases, which is beyond the scope of our study. Thus, the analyzes are limited with the problem instances solved by the solver in 4 hours, i.e., small-medium problem instances. However, we think that even this can be useful to understand the potential values and drawbacks of the model for practical issues.

5.6.1. Price of Fairness

(MIP-MMDP) is limited to deterministic policies, even a randomized policy can promise a better objective function value. Nevertheless, we require deterministic policies due to the fairness concerns raised in healthcare
This is also the case for other domains such as humanitarian relief operations [31]. The term fairness relates to the ethical issues which oblige policy-makers to follow the same policy for all patients in the same state. Hence, (MIP-MMDP) with randomized policies, i.e., without (9) - (11), becomes a relaxation of the original formulation. Then, price of fairness (%), determined as \(\frac{f^r - f^\text{MIP-MMDP}}{f^r} \times 100\), corresponds to the loss in the objective function value because of the constraints which eliminate randomized policies, where \(f^r\) denotes the optimal objective function value of the model allowing randomized policies.

Table 6: Price of Fairness for Small-Medium Problem Instances

| nScenario | nStage | Price of Fairness (%) |
|-----------|--------|-----------------------|
| 5         | 5      | 4.10                  |
| 5         | 10     | 4.66                  |
| 5         | 20     | 4.15                  |
| 5         | 30     | 3.89                  |
| 5         | 40     | 3.80                  |
| 10        | 5      | 4.19                  |
| 10        | 10     | 5.02                  |
| 10        | 20     | 4.89                  |
| 10        | 30     | 4.82                  |
| 10        | 40     | 4.80                  |
| 25        | 5      | 5.96                  |
| 25        | 10     | 6.98                  |
| 25        | 20     | 6.45                  |
| 25        | 30     | 6.31                  |
| 25        | 40     | 6.27                  |
| 50        | 5      | 6.81                  |
| 50        | 10     | 7.99                  |
| 50        | 20     | 7.13                  |
| 50        | 30     | 8.39                  |
| 50        | 40     | 7.08                  |
| 50        | 50     | 7.07                  |
| 100       | 5      | 7.13                  |
| 250       | 5      | 7.08                  |
| 500       | 5      | 7.07                  |

Price of fairness (%) for each small-medium problem instance is demonstrated in Table 6. Results point out that the average loss caused by the ethical concerns is 5.75%. It stems from the fact that a portion of the capacity at each decision epoch becomes idle due to the deterministic policy constraints. Allowing randomized policies enables the model to fulfill the capacity, even if only a subset of patients in a state utilize the resource. Results also show that as the number of scenarios increases, price of fairness tends to increase.

5.6.2. Value of Flexibility

We have so far assumed flexibility in actions, where in each period the actions are independent of the actions taken in other periods. That is, one state may be accepted in stage \(t\) and not accepted in stages \(t' \in \tilde{T} \setminus \{t\}\). In practice, policy makers may not have this flexibility; a state is either entitled to the resource for all decision epochs from the beginning or not at all. Therefore, here we consider the case in which once action is 1 for a state in a particular decision epoch, then it must also be 1 for other decision epochs. Mathematically, this corresponds to adding the following constraint
Table 7: Value of Flexibility for Small-Medium Problem Instances

| Scenario | Stage | Value of Flexibility (%) |
|----------|-------|--------------------------|
| 5        | 5     | 2.53                     |
| 5        | 10    | 10.24                    |
| 5        | 20    | 12.39                    |
| 5        | 30    | 13.25                    |
| 5        | 40    | 13.25                    |
| 10       | 5     | 2.16                     |
| 10       | 10    | 5.88                     |
| 10       | 20    | 11.81                    |
| 10       | 30    | 15.11                    |
| 10       | 40    | 16.63                    |
| 25       | 5     | 2.12                     |
| 25       | 10    | 5.34                     |
| 25       | 20    | 13.33                    |
| 25       | 30    | 17.61                    |
| 25       | 40    | 19.59                    |
| 50       | 5     | 1.55                     |
| 50       | 10    | 4.73                     |
| 100      | 5     | 1.23                     |
| 100      | 10    | 4.15                     |
| 250      | 5     | 1.21                     |
| 500      | 5     | 1.25                     |

Value of flexibility (%) is determined as \( \frac{f_{\text{MIP-MMDP}} - \tilde{f}}{f_{\text{MIP-MMDP}}} \times 100 \). It is clear that including flexibility narrows the feasible region so that the previous objective function value becomes an upper bound for the new model, i.e., \( \tilde{f} \leq f_{\text{MIP-MMDP}} \). In this study, flexibility implies that policy-makers can change their course of actions over time. Since \( f_{\text{MIP-MMDP}} \) allows it, \( \text{value of flexibility (\%)} \) is determined as \( \frac{f_{\text{MIP-MMDP}} - \tilde{f}}{f_{\text{MIP-MMDP}}} \times 100 \). It is clear that including it narrows the feasible region so that the previous objective function value becomes an upper bound for the new model, i.e., \( \tilde{f} \leq f_{\text{MIP-MMDP}} \) where \( \tilde{f} \) is the optimal objective function value under the same policy approach. In this study, flexibility implies that policy-makers can change their course of actions over time. Since \( f_{\text{MIP-MMDP}} \) allows it, \( \text{value of flexibility (\%)} \) is determined as \( \frac{f_{\text{MIP-MMDP}} - \tilde{f}}{f_{\text{MIP-MMDP}}} \times 100 \).

Value of flexibility (%) is computed for each small-medium problem instance, and the values are represented in Table 7. It shows that 7.98% additional value can be obtained by \( f_{\text{MIP-MMDP}} \) instead of using the same policy approach. In other words, embracing flexibility yields almost 7.98% additional total expected reward. Furthermore, average values for \( T = 5, 10, 20, 30, \) and 40 are computed and summarized in Figure 3. It demonstrates that as number of decision epochs increases, the value of embracing flexibility also increases dramatically since the suboptimal behavior is retained for longer periods. This also shows the relative importance of our model for long-term planning problems compared to the ones for short-terms.

5.6.3. Capacity Management

In this section, we analyze the additional total expected reward that can be obtained by adding one more capacity at each period. To this end, we find total expected rewards for each instance from \( \bigcup_{c=0.2}^{0.8} I(200, t, c, 0.25) \) for \( t \in \{10, 20, 30, 40\} \). In words, for \( t \in \{10, 20, 30, 40\} \), we start with stage capacity 200 and gradually increases it until 800 for a fixed solution sample which consists of 200 scenarios. We also follow this procedure for the case
where randomized policies are allowed. Figure 4 and Figure 5 illustrate the results of the experiment for the cases with deterministic policies and randomized policies, respectively. The main difference between deterministic and randomized policies become more distinguishable when we compare the graphs in Figure 4 and Figure 5. Table 5 clearly shows that the graph tends to perform concave characteristics. That is, the marginal utility of adding one unit of stage capacity is decreasing, as the value of stage capacity increases. Unlike the graph in Figure 5, we do not observe smooth increases within the graph in Figure 4. That is, increasing the capacity for deterministic policies does not necessarily improves the current conditions since the model still may not find an available group to fulfill the remaining capacity. The frequently observed constant lines in Figure 4 corresponds to these cases. Thus, system managers embracing deterministic policies must take this situation into account when they need to decide how much to increase the capacity.
6. Conclusion

In this study, we introduce a multi-model Markov decision process model with capacity constraints, motivated from medical resource allocation problems. Multi-model approach implies relaxation of the traditional assumption which ensures deterministic transition probabilities and rewards. Capacity
constraints pose limitations on the selection of particular actions. To the best of our knowledge, the proposed model is the first constrained MDP model considering the parameter uncertainty.

We develop a mixed integer programming (MIP) formulation for the model in order to find the strategy which maximizes the total expected reward without violating capacity constraints. It corresponds to an extensive-form formulation for the underlying two-stage stochastic integer program. The scenarios used in the extensive-form formulation basically corresponds to different MDP models. The MIP formulation becomes easily intractable as the size of the problem grows. In this regard, we propose a parallel approximate dynamic programming algorithm leveraging the problem structure. We also propose two valid inequalities in the hope that they can strengthen the formulation.

We test our model and algorithm using a chronic care delivery problem as an example. We adopt a Monte Carlo approach which uses prior beliefs about system dynamics to generate problem instances. Extensive computational experiments are then performed to test the computational aspects of the proposed algorithm as well as the value of our model. We show that the algorithm works pretty fast so that it generates solutions in seconds even for very large problem instances. It also offers very high quality solutions in terms of optimality. However, while our algorithm embraces the cases with a large number of scenarios and stages, it does not address the last level of complexity which is the state space.

There are several potential approaches to complement our work. First, a new algorithm can be proposed that also works fast for large state space problems. In this context, it may be promising to address constrained reinforcement learning models. Secondly, for some medical decision making problems such as breast cancer screening, we may not have have perfect information of the health status of patient [11]. To deal with these kind of challenges, our model can be extended for POMDP models.
References

[1] Alagoz, O., Hsu, H., Schaefer, A. J., and Roberts, M. S. (2010). Markov decision processes: a tool for sequential decision making under uncertainty. *Medical Decision Making*, 30(4):474–483.

[2] Alagoz, O., Maillart, L. M., Schaefer, A. J., and Roberts, M. S. (2004). The optimal timing of living-donor liver transplantation. *Management Science*, 50(10):1420–1430.

[3] Altman, E. (1999). *Constrained Markov decision processes*, volume 7. CRC Press.

[4] Ayvaci, M. U., Alagoz, O., and Burnside, E. S. (2012). The effect of budgetary restrictions on breast cancer diagnostic decisions. *Manufacturing & Service Operations Management*, 14(4):600–617.

[5] Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. (2009). *Robust optimization*, volume 28. Princeton University Press.

[6] Ben-Tal, A. and Nemirovski, A. (2002). Robust optimization—methodology and applications. *Mathematical programming*, 92(3):453–480.

[7] Bernell, S. and Howard, S. W. (2016). Use your words carefully: what is a chronic disease? *Frontiers in public health*, 4:159.

[8] Bertsimas, D., Brown, D. B., and Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM review*, 53(3):464–501.

[9] Boucherie, R. J. and Van Dijk, N. M. (2017). *Markov decision processes in practice*, volume 248. Springer.

[10] Britt, H. C., Harrison, C. M., Miller, G. C., and Knox, S. A. (2008). Prevalence and patterns of multimorbidity in australia. *Medical Journal of Australia*, 189(2):72–77.

[11] Cevik, M., Ayer, T., Alagoz, O., and Sprague, B. L. (2018). Analysis of mammography screening policies under resource constraints. *Production and Operations Management*, 27(5):949–972.

[12] Council, N. R., on Population, C., et al. (2015). *Measuring the Risks and Causes of Premature Death: Summary of Workshops*. National Academies Press.

[13] Denton, B. T. (2018). Optimization of sequential decision making for chronic diseases: From data to decisions. In *Recent Advances in Optimization and Modeling of Contemporary Problems*, pages 316–348. INFORMS.
[14] Denton, B. T., Kurt, M., Shah, N. D., Bryant, S. C., and Smith, S. A. (2009). Optimizing the start time of statin therapy for patients with diabetes. *Medical Decision Making*, 29(3):351–367.

[15] Deo, S., Iravani, S., Jiang, T., Smilowitz, K., and Samuelson, S. (2013). Improving health outcomes through better capacity allocation in a community-based chronic care model. *Operations Research*, 61(6):1277–1294.

[16] Givan, R., Leach, S., and Dean, T. (2000). Bounded-parameter markov decision processes. *Artificial Intelligence*, 122(1-2):71–109.

[17] Gorissen, B. L., Yanoğlu, İ., and den Hertog, D. (2015). A practical guide to robust optimization. *Omega*, 53:124–137.

[18] Grand-Clement, J., Chan, C. W., Goyal, V., and Escobar, G. (2020). Robust policies for proactive icu transfers. *arXiv preprint arXiv:2002.06247*.

[19] Hörmann, W., Leydold, J., and Derflinger, G. (2013). *Automatic nonuniform random variate generation*. Springer Science & Business Media.

[20] Iyengar, G. N. (2005). Robust dynamic programming. *Mathematics of Operations Research*, 30(2):257–280.

[21] Mannor, S., Mebel, O., and Xu, H. (2016). Robust mdps with k-rectangular uncertainty. *Mathematics of Operations Research*, 41(4):1484–1509.

[22] Mannor, S., Simester, D., Sun, P., and Tsitsiklis, J. N. (2007). Bias and variance approximation in value function estimates. *Management Science*, 53(2):308–322.

[23] Meng, L., Maskarinec, G., Lee, J., and Kolonel, L. N. (1999). Lifestyle factors and chronic diseases: application of a composite risk index. *Preventive medicine*, 29(4):296–304.

[24] Merakh, M. and Kıcıkuyavuz, S. (2019). Risk aversion to parameter uncertainty in markov decision processes with an application to slow-onset disaster relief. *IIE Transactions*, pages 1–21.

[25] Nilim, A. and El Ghaoui, L. (2004). Robustness in markov decision problems with uncertain transition matrices. In *Advances in neural information processing systems*, pages 839–846.

[26] Puterman, M. L. (2014). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.
[27] Satia, J. K. and Lave Jr, R. E. (1973). Markovian decision processes with uncertain transition probabilities. Operations Research, 21(3):728–740.

[28] Seifert, R. W., Tancrez, J.-S., and Biçer, I. (2016). Dynamic product portfolio management with life cycle considerations. International Journal of Production Economics, 171:71–83.

[29] Shechter, S. M., Bailey, M. D., Schaefer, A. J., and Roberts, M. S. (2008). The optimal time to initiate hiv therapy under ordered health states. Operations Research, 56(1):20–33.

[30] Sinha, S. and Ghate, A. (2016). Policy iteration for robust nonstationary markov decision processes. Optimization Letters, 10(8):1613–1628.

[31] Steimle, L. N., Kaufman, D. L., and Denton, B. T. (2018). Multimodel markov decision processes. Optimization Online URL http://www.optimization-online.org/DB_FILE/2018/01/6434.pdf.

[32] Tewari, A. and Bartlett, P. L. (2007). Bounded parameter markov decision processes with average reward criterion. In International Conference on Computational Learning Theory, pages 263–277. Springer.

[33] White III, C. C. and Eldeib, H. K. (1994). Markov decision processes with imprecise transition probabilities. Operations Research, 42(4):739–749.

[34] Wiesemann, W., Kuhn, D., and Rustem, B. (2013). Robust markov decision processes. Mathematics of Operations Research, 38(1):153–183.

[35] Wolff, J. L., Starfield, B., and Anderson, G. (2002). Prevalence, expenditures, and complications of multiple chronic conditions in the elderly. Archives of internal medicine, 162(20):2269–2276.

[36] Zhang, Y., Steimle, L., and Denton, B. (2017a). Robust markov decision processes for medical treatment decisions. Optimization online.

[37] Zhang, Y., Wu, H., Denton, B. T., Wilson, J. R., and Lobo, J. M. (2017b). Probabilistic sensitivity analysis on markov decision processes with uncertain transition probabilities: an application in evaluating treatment decisions for type 2 diabetes.
Appendix A. Proof of Valid Inequalities

Appendix A.1. Proof of Proposition 2.1

• for $t = 1$:

We are given that $\sum_{a \in A} x_{ia}^{\omega,1} = \theta_i$ for each $i \in \tilde{S}$ and $\omega \in \Omega$ by (4). Then,

$$\sum_{i \in \tilde{S}} \sum_{a \in A} x_{ia}^{\omega,t} = \sum_{i \in \tilde{S}} \theta_i, \quad \forall \omega \in \Omega$$

$$= 1, \quad \forall \omega \in \Omega \quad \text{(Since \(\theta\) is a valid PMF over \(\tilde{S}\))}$$

• for $2 \leq t \leq T - 1$:

We are given that $\sum_{a \in A} x_{ia}^{\omega,t} = \sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-1} P_{hai}$ for each $i \in \tilde{S}$ and $\omega \in \Omega$ by [5]. Moreover, we can re-write $Z^{\omega,t}$ as shown in (7). Then, following equations are satisfied for each scenario $\omega \in \Omega$:

$$\sum_{i \in \tilde{S}} \sum_{a \in A} x_{ia}^{\omega,t} + Z^{\omega,t} = \sum_{i \in \tilde{S}} \sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-1} P_{hai} + \sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-1} Q_{ha} + Z^{\omega,t-1}$$

$$= \sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-1} \left[ \sum_{i \in \tilde{S}} P_{ha}^{\omega} + Q_{ha}^{\omega} \right] + Z^{\omega,t-1}$$

$$= \sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-1} + Z^{\omega,t-1}$$

Now, we should prove that $\sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-1} + Z^{\omega,t-1}$ must be equal to 1. To that end, we utilize proof by induction as follows:

- **Base Case:** for $t = 2$, $\sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,1} + Z^{\omega,1} \geq 1, \forall \omega \in \Omega$

  For $t = 2$, the lhs is reduced to $\sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,1} + Z^{\omega,1}$. We have already proved that $\sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,1} = 1, \forall \omega \in \Omega$. We also know that $Z^{\omega,1} = 0, \forall \omega \in \Omega$. Thus, base case is satisfied.

- **Induction Step:** $\sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-1} + Z^{\omega,t-1} \geq 1$ given that $\sum_{h \in \tilde{S}} \sum_{a \in A} x_{ha}^{\omega,t-2} + Z^{\omega,t-2} = 1, \forall \omega \in \Omega$

  We are given that $\sum_{a \in A} x_{ha}^{\omega,t-1} = \sum_{i \in \tilde{S}} \sum_{a \in A} x_{ia}^{\omega,t-2} P_{iah}, \forall \omega \in \Omega$ for each $h \in \tilde{S}$ by [5]. Moreover, we can re-write $Z^{\omega,t-1}$ as expressed.
Then following equations are satisfied for each scenario $\omega \in \Omega$,
\[
\sum_{h \in \hat{S}} \sum_{a \in A} X_{ha}^{\omega, t-1} + Z_{\omega, t-1}^{\omega} = \sum_{i \in \hat{S}} \sum_{a \in A} X_{ia}^{\omega, t-2} \left[ \sum_{h \in \hat{S}} P_{iah}^{\omega} + Q_{ia}^{\omega} \right] + Z_{\omega, t-2}^{\omega} = 1, \quad \forall (i,a) \in (\hat{S}, A)
\]
\[
\sum_{i \in \hat{S}} \sum_{a \in A} X_{ia}^{\omega, t-2} + Z_{\omega, t-2}^{\omega} = 1
\]

- for $t = T$:

We are given that $Y_{i}^{\omega} = \sum_{h \in \hat{S}} \sum_{a \in A} X_{ha}^{\omega, T-1} P_{hai}^{\omega}$ for each $i \in \hat{S}$ and $\omega \in \Omega$. Moreover, we can re-write $Z_{\omega, T-1}^{\omega}$ as depicted in (7). Then, following equations are satisfied for each scenario $\omega \in \Omega$:

\[
\sum_{i \in \hat{S}} Y_{i}^{\omega} + Z_{\omega, T}^{\omega} = \sum_{h \in \hat{S}} \sum_{a \in A} X_{ha}^{\omega, T-1} P_{hai}^{\omega} + \sum_{h \in \hat{S}} \sum_{a \in A} X_{ha}^{\omega, T-1} Q_{ha}^{\omega} + Z_{\omega, T-1}^{\omega} = 1
\]

**Appendix A.2. Proposition 2.2**

\[
\sum_{i \in \hat{S}} X_{i,1}^{\omega, t} \leq \frac{C_t}{n} \quad \forall t \in \hat{T}, \omega \in \hat{\Omega}
\]
\[
\sum_{t \in \hat{T}} \sum_{i \in \hat{S}} X_{i,1}^{\omega, t} \leq \frac{\sum_{t \in \hat{T}} C_t}{n} \quad \forall \omega \in \hat{\Omega}
\]
\[
\sum_{t \in \hat{T}} \left[ \sum_{i \in \hat{S}} \left[ X_{i,1}^{\omega, t} + X_{i,0}^{\omega, t} \right] + Z_{\omega, t}^{\omega} \right] \leq \frac{\sum_{t \in \hat{T}} C_t}{n} + \sum_{t \in \hat{T}} \left[ \sum_{i \in \hat{S}} X_{i,0}^{\omega, t} + Z_{\omega, t}^{\omega} \right] \quad \forall \omega \in \hat{\Omega}
\]
\[
T - 1 \leq \frac{\sum_{t \in \hat{T}} C_t}{n} + \sum_{t \in \hat{T}} \left[ \sum_{i \in \hat{S}} X_{i,0}^{\omega, t} + Z_{\omega, t}^{\omega} \right] \quad \forall \omega \in \hat{\Omega}
\]
\[ T - \frac{\sum_{t \in T} C_t + n}{n} \leq \sum_{t \in T} \sum_{i \in S} \lambda_{t,0}^{\omega t} Z_{t,0}^{\omega t} \quad \forall \omega \in \Omega \]

**Appendix B. Hierarchical Rules for Transition Probabilities and Rewards**

*Appendix B.1. Transition Probabilities*

1. Consider two patients in the same awareness level. The patient with worse health status is more likely to worsen.

\[ P_{1,a,2} \geq P_{0,a,1} \quad \forall a \in \{0, 1\} \]
\[ P_{4,a,5} \geq P_{3,a,4} \quad \forall a \in \{0, 1\} \]

2. Consider two patients in the same complexity level. The patient with the higher awareness level is less likely to worsen.

\[ P_{0,a,1} \geq P_{3,a,4} \quad \forall a \in \{0, 1\} \]
\[ P_{1,a,2} \geq P_{4,a,5} \quad \forall a \in \{0, 1\} \]

3. Consider two patients in the same complexity and awareness levels. The patient taking normal care is more likely to worsen than the patient taking special care.

\[ P_{0,0,1} \geq P_{0,1,1} \]
\[ P_{1,0,2} \geq P_{1,1,2} \]
\[ P_{3,0,4} \geq P_{3,1,4} \]
\[ P_{4,0,5} \geq P_{4,1,5} \]

4. Probability of making a transition from low awareness level to high awareness level are equal regardless of the health status for all patients under same action. Moreover, special care is increasing this probability.

\[ P_{0,a,3} = P_{1,a,4} = P_{2,a,5} \quad \forall a \in \{0, 1\} \]
\[ P_{i,1,i+3} \geq P_{i,0,i+3} \quad \forall i \in \{0, 1, 2\} \]

5. Probability of making a transition from high awareness level to low awareness level are equal regardless of the health status for all patients under same action. Moreover, special care is decreasing this probability.

\[ P_{3,a,0} = P_{4,a,1} = P_{5,a,2} \quad \forall a \in \{0, 1\} \]

36
\[ P_{i,0,i-3} \geq P_{i,1,i-3} \quad \forall i \in \{3, 4, 5\} \]

6. A patient cannot change its health status and awareness level in one decision epoch. That is, diagonal arcs in the network is not possible for all patients under any course of action. Therefore, related transition probabilities equal to zero.

\[
\begin{align*}
P_{0,0,4} &= P_{1,0,5} = P_{3,0,1} = P_{4,0,2} = P_{4,0,0} = P_{5,0,1} = P_{1,0,3} = P_{2,0,4} = 0 \\
P_{0,1,4} &= P_{1,1,5} = P_{3,1,1} = P_{4,1,2} = P_{4,1,0} = P_{5,1,1} = P_{1,1,3} = P_{2,1,4} = 0
\end{align*}
\]

7. There is not any direct transition from simple to complex.

\[
P_{0,a,2} = P_{3,a,5} = 0 \quad \forall a \in \{0, 1\}
\]

8. Recovery is not possible for all patients under any course of action. Therefore, transition probabilities associated with the arcs to the left equal to zero.

\[
P_{1,a,0} = P_{2,a,1} = P_{4,a,3} = P_{5,a,4} = 0 \quad \forall a \in \{0, 1\}
\]

9. Following relations related to the death probabilities have to be satisfied:

\[
Q_{2,a} \geq Q_{5,a} = Q_{1,a} \geq Q_{4,a} = Q_{0,a} \geq Q_{3,a} \quad \forall a \in \{0, 1\}
\]

10. Special care reduces the death probabilities

\[
Q_{i,1} \leq Q_{i,0} \quad \forall i \in \{0, \ldots, 5\}
\]

11. There is an upper bound for death probabilities

\[
Q_{ia} \leq 0.20 \quad \forall i \in \{0, \ldots, 5\}, a \in A
\]

Appendix B.2. Rewards

1. Following relations have to be satisfied:

\[
r_{2,a} \geq r_{5,a} = r_{1,a} \geq r_{4,a} = r_{0,a} \geq r_{3,a} \quad \forall a \in \{0, 1\}
\]

2. Special care is more desirable than normal care.

\[
r_{i,1} \geq r_{i,0} \quad \forall i \in \{0, \ldots, 5\}
\]
3. Final stage rewards are computed as follows:

\[ R_i \triangleq \sum_{a \in \mathcal{A}} r_{ia} \quad \forall i \in \{0, \ldots, 5\} \]

4. Domain of rewards is \([100, 1000]\) except \(R^D\) where \(R^D = 0\).

\[ 100 \leq r_{ia} \leq 1000 \quad \forall i \in \{0, \ldots, 5\}, a \in \mathcal{A} \]

Appendix C. Pseudocode of the Strategy Evaluation Algorithm

**Algorithm 3** Strategy Evaluation Algorithm

**Require:** \(\hat{\Pi}\): the given strategy

1: for \(t \in T\) do
2: if \(t = 1\) then
3: for \(j \in \tilde{S}\) do
4: if \(\hat{\pi}_j = 1\) then
5: \(X_{j,0}^w \leftarrow 0\) and \(X_{j,1}^w \leftarrow \theta_j\) \(\forall \omega \in \Omega\)
6: else
7: \(X_{j,0}^w \leftarrow \theta_j\) and \(X_{j,1}^w \leftarrow 0\) \(\forall \omega \in \Omega\)
8: end if
9: \(Z_{\omega}^w \leftarrow 0\) \(\forall \omega \in \Omega\)
10: end for
11: end if
12: if \(1 < t < T\) then
13: for \(j \in \tilde{S}\) do
14: if \(\hat{\pi}_j = 1\) then
15: \(X_{j,0}^w \leftarrow 0\)
16: \(X_{j,1}^w \leftarrow \sum_{i \in S} X_{i,0}^{w,t-1} \bar{P}_{\omega}^w_{iaj}\) \(\forall \omega \in \Omega\)
17: else
18: \(X_{j,0}^w \leftarrow 0\)
19: \(X_{j,1}^w \leftarrow \sum_{i \in S} X_{i,1}^{w,t-1} \bar{P}_{\omega}^w_{iaj}\) \(\forall \omega \in \Omega\)
20: end if
21: \(Z^w \leftarrow Z^{w,t-1} + \sum_{i \in \tilde{S}} X_{i,1}^{w,t-1} Q_{\omega}^w\) \(\forall \omega \in \Omega\)
22: end for
23: end if
24: if \(t = T\) then
25: for \(j \in \tilde{S}\) do
26: \(Y_j^w \leftarrow \sum_{i \in \tilde{S}} X_{i,1}^{w,t-1} \bar{P}_{\omega}^w \forall \omega \in \Omega\)
27: end for
28: end if
29: end for

**Ensure:** \((X, Y, Z)^{\hat{\Pi}}\): the resulting occupancy measures

Appendix D. Evaluating \(f^w(\Omega)\)

An optimal strategy for a single scenario does not guarantee the feasibility under a large number of scenarios. Because the occupancy measures
implied by the strategy of a single scenario may violate the capacity constraints for one of the other scenarios. Thus, we embrace the following approach to evaluate $f^\omega(\Omega)$.

Let $\hat{\Pi}^\omega \triangleq \{\hat{\pi}^\omega_{i,t}\}_{i\in\tilde{S},t\in\tilde{T}}$ be the optimal strategy, and $\hat{\pi}^\omega_{i,t}$ be the optimal policy at stage $t \in \tilde{T}$ for state $i \in \tilde{S}$ under scenario $\omega$. We first evaluate $\hat{\Pi}^\omega$ under $\Omega$ by using Algorithm 3. If it does not involve any infeasibility, the total expected reward characterized by $\hat{\Pi}^\omega$ becomes $f^\omega(\Omega)$. Otherwise, $\hat{\Pi}^\omega$ consists of inapplicable policies. Thus, we find the nearest feasible strategy $\Pi^*$ to $\hat{\Pi}^\omega$, and evaluate $f^\omega(\Omega)$ as the total expected reward characterized by $\Pi^*$. Here, the nearest feasible strategy refers to the one that entails feasible policies with a minimum number of changes in the original policy. The distance is specified based on the changes in the policy since we assume that policy makers tend to apply as few as possible changes in the original plan as inconveniences appear during the execution. In this regard, the nearest strategy $\Pi^*$ to $\hat{\Pi}^\omega$ is determined by the following mixed-integer program.

\[
\begin{align*}
\text{min} & \quad \sum_{t \in \tilde{T}} \sum_{i \in \tilde{S}} \beta^t_i \\
\text{st;} & \quad \hat{\Pi} - \Pi \\
& \quad \pi^t_i - \pi^\omega_{i,t} \leq \beta^t_i \quad \forall t \in \tilde{T}, i \in \tilde{S} \\
& \quad \hat{\pi}^\omega_{i,t} - \pi^t_i \leq \beta^t_i \quad \forall t \in \tilde{T}, i \in \tilde{S} \\
& \quad \beta^t_i \in \{0, 1\} \quad \forall t \in \tilde{T}, i \in \tilde{S}
\end{align*}
\]