Multi-model Markov Decision Processes: A New Method for Mitigating Parameter Ambiguity

(Author’s names blinded for peer review)

Problem definition: Markov decision processes (MDPs) have found success in many areas including the evaluation and design of treatment and screening protocols for medical decision making problems. However, the usefulness of these models is only as good as the data used to parameterize them, and multiple competing data sources are common in medicine. That is, there may be multiple models for parameter calibration, creating ambiguity. We ask: How can we improve stochastic dynamic programming methods to account for parameter ambiguity in MDPs? Further, how much benefit is there to mitigating the effects of ambiguity? Academic/practical relevance: The problem of multiple, possibly conflicting studies is common in medicine, but this problem of multiple models has not been addressed previously for the MDP framework. Methodology: In this paper, we introduce the Multi-model Markov Decision Process (MMDP) which generalizes a standard MDP by allowing for multiple models of the rewards and transition probabilities. Solution of the MMDP generates a single policy that maximizes the weighted performance over all models. This approach allows for the decision maker to explicitly trade off conflicting sources of data while at the same time relaxing strong assumptions made in the prior literature. Results: We study the structural properties of this problem and show that this problem is at least NP-hard. We develop exact methods and fast approximation methods supported by error bounds for solving the weighted value problem. Finally, we illustrate the effectiveness and the scalability of our approach using a case study in preventative blood pressure and cholesterol management that accounts for conflicting published cardiovascular risk models. Managerial implications: The MMDP framework is a practical solution for medical policymakers to determine a single informative solution in the face of multiple models.

Key words: Robust dynamic programming; medical decision making; Markov decision processes; parameter ambiguity; healthcare applications
1. Introduction

Markov decision processes (MDPs) are a mathematical framework for sequential decision making under uncertainty that have informed decision making in a variety of application areas including inventory control, scheduling, finance, and medicine (Puterman 1994, Boucherie and Van Dijk 2017). MDPs generalize Markov chains in that a decision maker (DM) can take actions to influence the rewards and transition dynamics of the system. Typically though, the rewards and transition dynamics are either estimated from observational data or from external sources and can be subject to errors. Unfortunately, when the policy found via an optimization process using the estimates is evaluated under the true model parameters, the performance can be much worse than anticipated (Mannor et al. 2007). This motivates the need for MDPs that account for this ambiguity in model parameters.

We are motivated by medical applications for which Markov chains are among the most commonly used stochastic models for decision making. A keyword search of the US Library of Medicine Database using PubMed from 2007 to 2017 reveals more than 7,500 articles on the topic of Markov chains. Generalizing Markov chains to include decisions and rewards, MDPs are useful for designing optimal treatment and screening protocols, and have found success doing so for a number of important diseases; e.g., end-stage liver disease (Alagoz et al. 2007), HIV (Shechter et al. 2008), breast cancer (Ayer et al. 2012), and diabetes (Mason et al. 2014).

Despite the potential of MDPs to inform medical decision making, the utility of these models is often at the mercy of the data available to parameterize the models. The transition dynamics in medical decision making models are often parameterized using longitudinal observational patient data and/or results from the medical literature. However, longitudinal data are often limited due to the cost of acquisition, and therefore transition probability estimates are subject to statistical uncertainty. Challenges also arise in controlling observational patient data for bias and often there are unsettled conflicts in the results from different clinical studies; see Mount Hood 4 Modeling Group (2007), Etzioni et al. (2012), and Mandelblatt et al. (2016) for examples in the contexts of breast cancer, prostate cancer, and diabetes, respectively.

A specific example, and one that we will explore in detail, is in the context of cardiovascular disease for which cardiovascular risk calculators estimate the probability of a major cardiovascular event, such as a heart attack or stroke. There are multiple well-established risk calculators in the clinical literature that could be used to estimate these transition probabilities, including the American College of Cardiology/ American Heart Association Risk Estimator (Goff et al. 2014) and the risk equations resulting from the Framingham Heart Study (Wolf et al. 1991, Wilson et al.
However, these two credible models give conflicting estimates of a patient’s risk of having a major cardiovascular event. Steimle and Denton (2017) showed that the best treatment protocol for cardiovascular disease is sensitive to which of these conflicting estimates are used leaving an open question as to which clinical study should be used to parameterize the model.

This example motivates our main research questions: How can we improve stochastic dynamic programming methods to account for parameter ambiguity in MDPs? Further, how much benefit is there to mitigating the effects of ambiguity?

In this article, we describe a new approach for handling parameter ambiguity in MDPs, which we refer to as the Multi-model Markov Decision Process (MMDP). An MMDP generalizes an MDP to allow for multiple models of the transition probabilities and rewards, each defined on a common state space and action space. In this model formulation, the DM places a weight on each of the models and seeks to find a single policy that will maximize the weighted value function. We analyze the structure of this model and categorize two important variants: 1) a case where the DM is limited to policies determined by the current state of the system and 2) a more general case in which the DM attempts to find an optimal history-dependent policy based on all previously observed information. We further classify the computational complexity of these problems and use the underlying structure of the models to present exact and approximation methods with error bounds. We identify several connections between the MMDPs, stochastic programming, and partially-observable MDPs (POMDPs) that provide insights into these alternative approaches for decision-making under uncertainty. We then test these methods on randomly generated problem instances. Finally, we establish the effectiveness and scalability of this new modeling approach using a case study that addresses ambiguity in the context of preventive treatment of cardiovascular disease for patients with type 2 diabetes.

The remainder of this article is organized as follows: In Section 2, we provide some important background on MDPs and discuss the literature that is most related to our work. We formally define the MMDP in Section 3, and in Section 4 we present analysis of our proposed MMDP model. In Section 5, we discuss exact solution methods as well as fast and scalable approximation methods that exploit the model structure. We test these approximation algorithms on randomly generated problem instances and describe the results in Section 6. In Section 7, we present our case study. Finally, in Section 8, we summarize the most important findings from our research and discuss the limitations and opportunities for future research.
2. Background and literature review

In this article, we focus on discrete-time, finite-horizon MDPs with parameter ambiguity. In this section, we will describe the MDP and parameter ambiguity, as well as the related work aimed at mitigating the effects of ambiguity in MDPs.

2.1. Markov decision processes

MDPs are a common framework for modeling sequential decision-making that influences a stochastic reward process. For ease of explanation, we introduce the MDP as an interaction between an exogenous actor, nature, and the DM. The sequence of events that define the MDP are as follows: first, nature randomly selects an initial state $s_1 \in S$ according to the initial distribution $\mu_1 \in \mathcal{M}(S)$, where $\mathcal{M}(\cdot)$ denotes the set of probability measures on the discrete set. The DM observes the state $s_1 \in S$ and selects an action $a_1 \in A$. Then, the DM receives a reward $r_1(s_1, a_1) \in R$ and then nature selects a new state $s_2 \in S$ with probability $p_1(s_2 | s_1, a_1) \in [0, 1]$. This process continues whereby for any decision epoch $t \in T \equiv \{1, \ldots, T\}$, the DM observes the state $s_t \in S$, selects an action $a_t \in A$, and receives a reward $r_t(s_t, a_t)$, and nature selects a new state $s_{t+1} \in S$ with probability $p_t(s_{t+1} | s_t, a_t)$. The DM selects the last action at time $T$ which may influence which state is observed at time $T+1$ through the transition probabilities. Upon reaching $s_{T+1} \in S$ at time $T+1$, the DM receives a terminal reward of $r_{T+1}(s_{T+1}) \in R$. Future rewards are discounted at a rate of $\alpha \in (0, 1]$ which accounts for the preference of rewards received now over rewards received in the future. In this article, we assume without loss of generality that the discount factor is already incorporated into the reward definition. We will refer to the times at which the DM selects an action as the set of decision epochs, $T$, the set of rewards as $R \in \mathbb{R}^{|S \times A \times T|}$, and the set of transition probabilities as $P \in \mathbb{R}^{|S \times A \times S \times T|}$ with elements satisfying $p_t(s_{t+1} | s_t, a_t) \in [0, 1]$ and $\sum_{s_{t+1} \in S} p_t(s_{t+1} | s_t, a_t) = 1$, $\forall t \in T$, $s_t \in S$, $a_t \in A$. Throughout the remainder of this article, we will use the tuple $(T, S, A, R, P, \mu_1)$ to summarize the parameters of an MDP.

The realized value of the DM’s sequence of actions is the total reward over the planning horizon:

$$\sum_{t=1}^{T} r_t(s_t, a_t) + r_{T+1}(s_{T+1}).$$  \hspace{1cm} (1)

The objective of the DM is to select the sequence of actions in a strategic way so that the expectation of (1) is maximized. Thus, the DM will select the actions at each decision epoch based on some information available to her. The strategy by which the DM selects the action for each state at decision epoch $t \in T$ is called a decision rule, $\pi_t \in \Pi_t$, and the set of decision rules over the planning horizon is called a policy, $\pi \in \Pi$. 
There exist two dichotomies in the classes of policies that a DM may select from: 1) history-dependent vs. Markov, and 2) randomized vs. deterministic. History-dependent policies may consider the entire history of the MDP, \( h_t := (s_1, a_1, \ldots, a_{t-1}, s_t) \), when prescribing which action to select at decision epoch \( t \in T \), while Markov policies only consider the current state \( s_t \in S \) when selecting an action. Randomized policies specify a probability distribution over the action set, \( \pi_t(s_t) \in \mathcal{M}(A) \), such that action \( a_t \in A \) will be selected with probability \( \pi_t(a_t|s_t) \). Deterministic policies specify a single action to be selected with probability 1. Markov policies are a subset of history-dependent policies, and deterministic policies are a subset of randomized policies. For standard MDPs, there is guaranteed to be a Markov deterministic policy that maximizes the expectation of (1) (Proposition 4.4.3 of Puterman 1994) which allows for efficient solution methods that limit the search for optimal policies to the Markov deterministic (MD) policy class, \( \pi \in \Pi^{MD} \). We will distinguish between history-dependent (H) and Markov (M), as well as randomized (R) and deterministic (D), using superscripts on \( \Pi \). For example, \( \Pi^{MR} \) denotes the class of Markov randomized policies.

To summarize, given an MDP \((T, S, A, R, P, \mu_1)\), the DM seeks to find a policy \( \pi \) that maximizes the expected rewards over the planning horizon:

\[
\max_{\pi \in \Pi} \mathbb{E}_{\pi, P, \mu_1}\left[ \sum_{t=1}^{T} r_t(s_t, a_t) + r_{T+1}(s_{T+1}) \right].
\] (2)

A standard MDP solution can be computed in polynomial time because the problem decomposes when the search over \( \Pi \) is limited to the Markov deterministic policy class, \( \Pi^{MD} \). We will show that this and other properties of MDPs no longer hold when parameter ambiguity is considered.

### 2.2. Parameter Ambiguity and Related Work

MDPs are known as models of sequential decision making under uncertainty. However, this “uncertainty” refers to the imperfect information about the future state of the system after an action has been taken due to stochasticity. The transition probability parameters are used to characterize the likelihood of these future events. For the reasons described in Section 1, the model parameters themselves may not be known with certainty. For clarity, throughout this article, we will refer to uncertainty as the imperfect information about the future which can be characterized via a set of transition probability parameters. We refer to ambiguity as the imperfect information about the transition probability parameters themselves. In this article, we consider a variation on MDPs in which parameter ambiguity is expressed through multiple models of the underlying Markov chain and the goal of the DM is to find a policy that maximizes the weighted performance across these
different models. The concept of multiple models of parameters is seen in the stochastic programming literature whereby each set corresponds to a “scenario” representing a different possibility for the problem data (Birge and Louveaux 1997). Stochastic programming problems typically consist of multiple stages during which the DM has differing levels of information about the model parameters. For example, in a two-stage stochastic program, the DM selects initial actions during the first-stage before knowing which of the multiple scenarios will occur. The DM subsequently observes which scenario is realized and takes recourse actions in the second stage. In contrast, an MMDP, the model parameters will never be revealed to the DM.

Perhaps the most closely related research to this article is that Bertsimas et al. (2016) who recently addressed ambiguity in simulation modeling in the context of prostate cancer screening. The authors propose solving a series of optimization problems via an iterated local search heuristic to find screening protocols that generate a Pareto optimal frontier on the dimensions of average-case and worst-case performance in a set of different simulation models. This article identified the general problem of multiple models in medical decision making; however, they do not consider this issue in MDPs. The concept of multiple models of problem parameters in MDPs has mostly been used as a form of sensitivity analysis. For example, Craig and Sendi (2002) propose bootstrapping as a way to generate multiple sets of problem parameters under which to evaluate the robustness of a policy to variation in the transition probabilities. There has been less focus on finding policies that perform well with respect to multiple models of the problem parameters in MDPs. As pointed out in a report from the Cancer Intervention and Surveillance Modeling Network regarding a comparative modeling effort for breast cancer, the authors note that “the challenge for reporting multimodel results to policymakers is to keep it (nearly) as simple as reporting one-model results, but with an understanding that it is more informative and more credible. We have not yet met this challenge” (Habbema et al. 2006). This highlights the goal of designing policies that are as easily translated to practice as those that optimize with respect to a single model, but with the robustness of policies that consider performance in multiple models.

The approach of incorporating multiple models of parameters is also seen in the reinforcement learning literature, however the objective of the DM in these problems is different than the objective of the DM in this article. For example, consider what is perhaps the most closely related reinforcement learning problem: the Contextual Markov Decision Process (CMDP) proposed by Hallak et al. (2015). The CMDP is essentially the same as the MMDP set-up in that one can think of the CMDP as an integer number, C, of MDPs all defined on the same state space and action space, but with different reward and transition probability parameters. In the CMDP problem, the
DM will interact with the CMDP throughout a series of episodes occurring serially in time. At the beginning of the interaction, the DM neither has any information about any of the $C$ MDPs’ parameters, nor does she know which MDP she is interacting with at the beginning of each episode. Our work differs from that of Hallak et al. (2015) in that we assume the DM has a complete characterization of each of the MDPs, but due to ambiguity the DM still does not know which MDP she is interacting with. Others have studied related problems in the setting of multi-task reinforcement learning (Brunskill 2012). Our work differs from this line of research in that we are motivated by problems with shorter horizons while multi-task learning is appropriate for problems in which the planning horizon is sufficiently long to observe convergence of estimates to their true parameters based on a dynamic learning process.

We view our research as distinct from the more traditional approach of mitigating parameter ambiguity in MDPs, known as robust dynamic programming, which represents parameter ambiguity through an ambiguity set formulation. The standard robust dynamic programming is a “max-min” approach in which the DM seeks to find a policy that maximizes the worst case performance when the transition probabilities are allowed to vary within an ambiguity set. The ambiguity set can be constructed as intervals around a point estimate and the max-min approach represents that the DM is risk neutral with respect to uncertainty and risk adverse with respect to ambiguity. One of the key results is that the max-min problem is tractable for instances that satisfy the rectangularity property (Iyengar 2005, Nilim and El Ghaoui 2005). Essentially, rectangularity means that observing the realization of a transition probability parameter gives no information about the values of other parameters for any other state-action-time triplet. Because each parameter value for any given state-action-time triplet is independent of the others, the problem can be decomposed so that each worst-case parameter is found via an optimization problem called the inner problem. Iyengar (2005) and Nilim and El Ghaoui (2005) provide algorithms for solving the max-min problem for a variety of ambiguity sets by providing polynomial-time methods for solving the corresponding inner problem. While rectangular ambiguity sets are desirable from a computational perspective, they can give rise to policies that are overly-conservative because the DM must account for the possibility that parameters for each state-action-time triplet will take on their worst-case values simultaneously.

Much of the research in robust dynamic programming has focused on ways to mitigate the effects of parameter ambiguity while avoiding policies that are overly conservative by either finding non-rectangular ambiguity sets that are tractable for the max-min problem or optimizing with respect to another objective function usually assuming some a priori information about the model.
parameters (Delage and Mannor 2009, Xu and Mannor 2012, Wiesemann et al. 2014, Mannor et al. 2016, Li et al. 2017, Scheftelowitsch et al. 2017). To our knowledge, Saghafian (2016) and Ahmed et al. (2017) are the only articles that have considered addressing ambiguity in the MDP parameters by using multiple discrete sets of parameters. Ahmed et al. (2017) propose sampling rewards and transition probabilities at each time step to generate a set of discrete MDPs and then seek to find one policy that minimizes the maximum regret over the set of MDPs. To do this, they formulate a mixed-integer linear program (MIP) to approximate an optimization problem with quadratic constraints which minimizes regret. They also propose cumulative expected myopic regret as a measure of regret for which dynamic programming algorithms can be used to generate an optimal policy. The authors require that the sampled transition probabilities and rewards are stage-wise independent, satisfying the rectangularity property. In the POMDP setting, Saghafian (2016) uses multiple models of the parameters to address ambiguity in transitions among the core states in a partially-observable MDP and use an objective function that weights the best-case and worst-case value-to-go across the models. This is in contrast to our work which considers the expected value-to-go among multiple models. The author assumes that the best-case and worst-case model are selected independently across decision epochs, again satisfying the rectangularity assumption. In our proposed MMDP formulation, the rectangularity assumption is not required; the objective is to find a single policy that will perform well in each of the models which may have interdependent transition probabilities across the planning horizon.

Later in this article, we will describe a case study that illustrates the effectiveness and scalability of the MMDP formulation on a medical decision making problem with parameter ambiguity in the context of prevention of cardiovascular disease. As pointed out in Section 1, MDPs are increasingly used for designing optimal treatment and screening protocols; however, the literature on addressing ambiguity in MDPs for medical decision making is very sparse. As mentioned previously, Bertsimas et al. (2016), Goh et al. (2015) proposed finding the best-case and worst-case transition probability parameters for this policy when these parameters are allowed to vary within an ambiguity set. The authors assumed that this ambiguity set is a row-wise independent set that generalizes the existing row-wise uncertainty models in Iyengar (2005) as well as Nilim and El Ghaoui (2005). This rectangularity assumption allows for the authors to solve a semi-infinite linear programming problem efficiently. The authors apply their methods to fecal immunochemical testing (FIT) for colorectal cancer and show that, despite the ambiguity in model parameters related to FIT, this screening tool is still cost-effective relative to the most prevalent method, colonoscopy.
To our knowledge, the optimal design of medical screening and treatment protocols under parameter ambiguity is limited to the work of Kaufman et al. (2011), Sinha et al. (2016), and Zhang et al. (2017). Kaufman et al. (2011) consider the optimal timing of living-donor liver transplantations, for which some critical health state are seldom visited historically. They use the robust MDP framework, modeling ambiguity sets as confidence regions based on relative entropy bounds. The resulting robust solutions are of a simple control-limit form that suggest transplanting sooner, when patients are healthier, than otherwise suggested by traditional MDP solutions based on maximum likelihood estimates of transition probabilities. Sinha et al. (2016) use a robust MDP formulation for response-guided dosing decisions in which the dose-response parameter is allowed to vary within an interval uncertainty set and show that a monotone dosing policy is optimal for the robust MDP. Zhang et al. (2017) propose a robust MDP framework in which transition probabilities are confined to statistical confidence intervals. They employ a rectangularity assumption implying independence of rows in the transition probability matrix and they assume an adversarial model in which the decision maker decides on a policy and an adversary optimizes the choice of transition probabilities that minimizes expected rewards subject to an uncertainty budget on the choice of transition probabilities. While these articles address parameter ambiguity in the transition probabilities, they all assume a rectangular ambiguity set which decouples the ambiguity across decision epochs and states. In contrast, the MMDP formulation that we propose allows a much needed relaxation of this assumption to allow for the ambiguity in model parameters to be linked across tuples of states, actions, and decision epochs.

3. Multi-model Markov decision processes

In this section, we introduce the detailed mathematical formulation of the MMDP starting with the following definition:

**Definition 1 (Multi-model Markov decision process).** An MMDP is a tuple \((\mathcal{T}, \mathcal{S}, \mathcal{A}, \mathcal{M}, \Lambda)\) where \(\mathcal{T}\) is the set of decision epochs, \(\mathcal{S}\) and \(\mathcal{A}\) are the state and action spaces respectively, \(\mathcal{M}\) is the finite discrete set of models, and \(\Lambda := \{\lambda_1, \ldots, \lambda_{|\mathcal{M}|}\}\) is the set of exogenous models weights with \(\lambda_m \in (0, 1), \forall m \in \mathcal{M}\) and \(\sum_{m \in \mathcal{M}} \lambda_m = 1\). Each model \(m \in \mathcal{M}\) is an MDP, \((\mathcal{T}, \mathcal{S}, \mathcal{A}, R^m, P^m, \mu^m)\), with a unique combination of rewards, transition probabilities, and initial distribution.

The requirement that \(\lambda_m \in (0, 1)\) is to avoid the trivial cases: If there exists a model \(m \in \mathcal{M}\) such that \(\lambda_m = 1\), the MMDP would reduce to a standard MDP. If there exists a model \(m \in \mathcal{M}\) such that \(\lambda_m = 0\), then the MMDP would reduce to an MMDP with a smaller set of models, \(\mathcal{M} \setminus \{m\}\).
The model weights, Λ, may be selected via expert judgment to stress the relative importance of each model, as tunable parameters which the DM can vary (as illustrated in the case study in Section 7), according to a probability distribution over the models, or as uninformed priors when each model is considered equally reputable (as in Bertsimas et al. (2016)).

In an MMDP, the DM considers the expected rewards of the specified policy in the multiple models. The value of a policy \( \pi \in \Pi \) in model \( m \in \mathcal{M} \) is given by its expected rewards evaluated with model \( m \)’s parameters:

\[
v^m(\pi) := \mathbb{E}^{\pi,P^m,\mu^m_1} \left[ \sum_{t=1}^{T} r^m_t(s_t,a_t) + r^m_{T+1}(s_{T+1}) \right].
\] (3)

We associate any policy, \( \pi \in \Pi \), for the MMDP with its weighted value:

\[
W(\pi) := \sum_{m \in \mathcal{M}} \lambda_m v^m(\pi) = \sum_{m \in \mathcal{M}} \lambda_m \mathbb{E}^{\pi,P^m,\mu^m_1} \left[ \sum_{t=1}^{T} r^m_t(s_t,a_t) + r^m_{T+1}(s_{T+1}) \right].
\] (4)

Thus, we consider the weighted value problem in which the goal of the DM is to find the policy \( \pi \in \Pi \) that maximizes the weighted value defined in (4):

**Definition 2 (Weighted value problem).** Given an MMDP \((\mathcal{T}, \mathcal{S}, \mathcal{A}, \mathcal{M}, \Lambda)\), the weighted value problem is defined as the problem of finding a solution to:

\[
W^* := \max_{\pi \in \Pi} W(\pi) = \max_{\pi \in \Pi} \left\{ \sum_{m \in \mathcal{M}} \lambda_m \mathbb{E}^{\pi,P^m,\mu^m_1} \left[ \sum_{t=1}^{T} r^m_t(s_t,a_t) + r^m_{T+1}(s_{T+1}) \right] \right\}
\] (5)

and a set of policies \( \Pi^* := \{ \pi^* : W(\pi^*) = W^* \} \subseteq \Pi \) that achieve the maximum in (5).

The weighted value problem can be viewed as an interaction between the DM (who seeks to maximize the expected weighted value of the MMDP) and nature (an exogenous actor who acts according to a specified set of dynamics). In many robust formulations, nature is viewed as an adversary which represents the risk-aversion to ambiguity in model parameters. However, in the weighted value problem, nature plays the role of a neutral counterpart to the DM. In this interaction, the DM knows the complete characterization of each of the models and nature selects which model will be given to the DM by randomly sampling according to the probability distribution defined by \( \Lambda \in \mathcal{M}(\mathcal{M}) \). For a fixed model \( m \in \mathcal{M} \), there will exist an optimal policy for \( m \) that is Markov (i.e., \( \pi^*_m \in \Pi^M \)). We will focus on the problem of finding a policy that achieves the maximum in (5) when \( \Pi = \Pi^M \). We will refer to this problem as the non-adaptive problem because we are enforcing that the DM’s policy be based solely on the current state and she cannot adjust her strategy based on what sequences of states she has observed. As we will show, unlike traditional MDPs, the restriction to \( \Pi^M \) may not lead to an overall optimal solution. For completeness, we
will also describe an extension, called the adaptive problem, where the DM can utilize information about the history of observed states, however this extension is not the primary focus of this article.

3.1. The non-adaptive problem

The non-adaptive problem for MMDPs is an interaction between nature and the DM. In this interaction, the DM specifies a Markov policy, \( \pi \in \Pi^M \), a priori. In this case, the policy is composed of actions based only on the current state at each decision epoch. Therefore the policy is a distribution over the actions: \( \pi = \{ \pi_t(s_t) = (\pi_t(1 \mid s_t), \ldots, \pi_t(\vert A \vert \mid s_t)) \in \mathcal{M}(A) : a_t \in A, s_t \in S, t \in T \} \). In this policy, \( \pi_t(a_t \mid s_t) \) is the probability of selecting action \( a_t \in A \) if the MMDP is in state \( s_t \in S \) at time \( t \in T \). Then, after the DM has specified the policy, nature randomly selects model \( m \in \mathcal{M} \) with probability \( \lambda_m \). Now, nature selects \( s_1 \in S \) according to the initial distribution \( \mu_m^1 = \mathcal{M}(S) \) and the DM selects an action, \( a_1 \in A \), according to the pre-specified distribution \( \pi_1(s_1) \in \mathcal{M}(A) \). Then, nature selects the next state \( s_2 \in S \) according to \( p_m^1(\cdot \mid s_1, a_1) \in \mathcal{M}(S) \). The interaction carries on in this way where the DM selects actions according to the pre-specified policy, \( \pi \), and nature selects the next state according to the distribution given by the corresponding row of the transition probability matrix.

3.2. The adaptive problem

The adaptive problem generalizes the non-adaptive problem to allow the DM to utilize realizations of the states to adjust her strategy. In this problem, nature and the DM interact sequentially where the DM gets new information in each decision epoch of the MMDP and the DM is allowed to utilize the realizations of the states to infer information about the ambiguous problem parameters when selecting her future actions. In this setting, nature begins the interaction by selecting a model, \( m \in \mathcal{M} \), according to the distribution \( \Lambda \), and the model selected is not known to the DM. Nature then selects an initial state \( s_1 \in S \) according to the model’s initial distribution, \( \mu_m^1 \). Next, the DM observes the state, \( s_1 \), and makes her move by selecting an action, \( a_1 \in A \). At this point, nature randomly samples the next state, \( s_2 \in S \), according to the distribution given by \( p_m^1(\cdot \mid s_1, a_1) \in \mathcal{M}(S) \). The interaction continues by alternating between the DM (who observes the state and selects an action) and nature (who selects the next state according to the distribution defined by the corresponding row of the transition probability matrix).

In the adaptive problem, the DM considers the current state of the MMDP along with information about all previous states observed and actions taken. Because the history is available to the DM, the DM may be able to infer which model is most likely to correctly characterize the behavior...
of nature which the DM is observing. As we will formally prove later, in this context the DM will specify a history-dependent policy in general, \( \pi = \{ \pi_t(h_t) : h_t \in S \times A \times \ldots \times A \times S, t \in T \} \).

4. Analysis of MMDPs

In this section, we will analyze the weighted value problem as defined in (5). For both the adaptive and non-adaptive problems, we will describe the classes of policies that achieve the optimal weighted value, the complexity of solving the problem, and related problems that may provide insights into promising solution methods. All proofs are available separately, in Blinded (2018).

4.1. General properties of the weighted value problem

In both the adaptive and non-adaptive problems, nature is confined to the same set of rules. However, the set of strategies available to the DM in the non-adaptive problem is just a subset of the strategies available in the adaptive problem. Therefore, if \( W^*_N \) and \( W^*_A \) are the best expected values that the DM can achieve in the non-adaptive and adaptive problems, respectively, then it follows that \( W^*_N \leq W^*_A \).

**Proposition 1.** \( W^*_N \leq W^*_A \). Moreover, the inequality may be strict.

**Corollary 1.** It is possible that there are no optimal policies that are Markovian for the adaptive problem.

The results of Proposition 1 and Corollary 1 mean that the DM may benefit from being able to recall the history of the MMDP. This history allows for the DM to infer which model is most likely, conditional on the observed sample path and tailor the future actions to reflect this changing belief about nature’s choice of model. Therefore, the DM must search for policies within the history-dependent policy class to find an optimal solution to the adaptive MMDP. These results establish that the adaptive problem does not reduce to the non-adaptive problem in general. For this reason, we separate the analysis for the adaptive and non-adaptive problems.

4.2. Analysis of the adaptive problem

We begin by establishing an important connection between the adaptive problem and the partially-observable MDP (POMDP) (Smallwood and Sondik 1973):

**Proposition 2.** Any MMDP can be recast as a special case of a partially observable MDP (POMDP) such that the maximum weighted value of the MMDP is equivalent to the expected discounted rewards of the POMDP.

**Corollary 2.** There is always a deterministic policy that is optimal for the adaptive problem.
Figure 1  A Venn diagram illustrating the relationship between an MDP, MMDP, and POMDP. As shown in Proposition 2, any MMDP is a special case of a POMDP due to the structure of the transition matrix and observation conditional probabilities. Further, an MDP is a special case of an MMDP in which the MMDP only has one model.

The implication of Proposition 2 is illustrated in Figure 1 which displays the relationship between MDPs, MMDPs, and POMDPs. Given Proposition 2, we can draw on similar ideas proposed in the literature for solving POMDPs and refine them to take advantage of structural properties specific to MMDPs. However, we show that even though MMDPs have special structure on the observation matrix and transition probability matrix (see the proof of Proposition 2 in the appendix), we cannot expect any improvements in the complexity of the problem due to this structure.

**Proposition 3.** The adaptive problem for MMDPs is PSPACE-hard.

Although the adaptive problem is PSPACE-hard and we cannot expect to develop an algorithm whose solution time is bounded above by a function that is polynomial in the problem size, it is still possible to develop an exact algorithm for solving this problem; see Blinded (2018) for details. However, the focus of the remainder of this article is the non-adaptive problem.

4.3. Analysis of the non-adaptive problem

In this section, we analyze the non-adaptive problem for which restricts the DM’s policy is restricted to the class of Markov policies ($\Pi^M$). We begin by establishing the important result that there always exists a deterministic optimal policy for the special case of the non-adaptive problem. This result is important because searching among policies in the Markov deterministic policy class may be appealing for several reasons: First, each individual model is solved by a policy in this class and it could be desirable to find a policy with the same properties as the each model’s individual optimal policy. Second, Markov policies are typically easier to implement because they only require the current state to be stored rather than partial or complete histories of the MDP. Third, Markov deterministic policies are ideal for medical decision making, the motivating application for this article, because they can be easily translated to treatment guidelines that are based solely on the
information available to the physician at the time of the patient visit, such as the patient’s current blood pressure levels. For applications in medicine, such as the case study in Section 7, deterministic policies are a necessity since randomization is unlikely to be considered ethical outside the context of randomized clinical trials.

**Proposition 4.** For the non-adaptive problem, there is always a Markov deterministic policy that is optimal.

This result means that for the non-adaptive problem, the DM can restrict her attention to the class of Markov deterministic policies. This result may be surprising at first due to the result of Fact 2 in Singh et al. (1994) which states that the best stationary randomized policy can be arbitrarily better than the best stationary deterministic policy for POMDPs. While this result may seem to contradict Proposition 4, it is worth noting that Fact 2 of Singh et al. (1994) was derived in the context of an infinite-horizon MDP in which it is possible that the same state can be visited more than once. In the finite-horizon MMDP, it is not possible that any state \( s_t \in S \) could be visited more than once.

Even though the non-adaptive problem requires searching over a smaller policy class than for the adaptive problem (\( \Pi^{MD} \subset \Pi^{HD} \)), the non-adaptive problem is still provably hard.

**Proposition 5.** Solving the non-adaptive problem for an MMDP is NP-hard.

The result of Proposition 5 implies that we cannot expect to find an algorithm that solves the non-adaptive problem for all MMDPs in polynomial time. Still, we are able to solve the non-adaptive problem by formulating it as a mixed-integer program (MIP) as discussed in the following proposition.

**Proposition 6.** Non-adaptive MMDPs can be formulated as the following MIP:

\[
\begin{align*}
\max_{v, x} & \quad \sum_{m \in M} \lambda_m \sum_{s \in S} \mu_1^m(s)v_1^m(s) \\
\text{s.t.} & \quad v_{T+1}^m(s) \leq r_{T+1}^m(s), \quad \forall s \in S, m \in M, \\
& \quad v_t^m(s) \leq r_t^m(s,a) + \sum_{s' \in S} p_t^m(s'|s,a)v_{t+1}^m(s') + M(1 - x_{s,a,t}), \quad \forall m \in M, s \in S, a \in A, t \in T, \\
& \quad \sum_{a \in A} x_{s,a,t} = 1, \quad \forall s \in S, t \in T, \\
& \quad x_{s,a,t} \in \{0, 1\}, \quad \forall s \in S, a \in A, t \in T.
\end{align*}
\]

In this formulation, the decision variables, \( v_t^m(s) \in \mathbb{R} \), represent the value-to-go from state \( s \in S \) at time \( t \in T \) in model \( m \in M \). The binary decision variables, \( x_{s,a,t} \in \{0, 1\} \), take on a value of 1 if the optimal policy prescribes taking action \( a \in A \), in state \( s \in S \), at epoch \( t \in T \), and 0 otherwise.
It is well-known that standard MDPs can be solved using a linear programming (LP) formulation (Puterman 1994, §6.9). Suppose that \( v(s, a) \) represents the value-to-go from state \( s \in S \) using action \( a \in A \). The LP approach for solving MDPs utilizes a reformulation trick that finding \( \max_{a \in A} v(s, a) \) is equivalent to finding \( \min v(s) \) such that \( v(s) \geq v(s, a), \forall a \in A \). In this reformulation, the constraint \( v(s) \geq v(s, a) \) is tight for all actions that are optimal. The MIP formulation described in (6) relies on similar ideas as the LP formulation of an MDP, but is modified to enforce the constraint that the policy must be the same across all MDPs. The purpose of the big-M is to ensure that \( v^m_t(s) = v^m_t(s, a) \) only if \( x_{s,a,t} = 1 \) meaning that the value-to-go for this state-time pair in model \( m \in M \) corresponds to the policy that is being used in all models.

The formulation of the non-adaptive problem as a MIP may seem more natural after a discussion of the connections with two-stage stochastic programming (Birge and Louveaux 1997). If we view the non-adaptive problem through the lens of stochastic programming, the \( x_{s,a,t} \) binary variables that define the policy can be interpreted as the first-stage decisions of a two-stage stochastic program. Moreover, nature’s choices of model, \( M \), correspond to the possible scenarios which are observed according to the probability distribution \( \Lambda \). In this interpretation, the value function variables, \( v^m_t(s) \), can be viewed as the recourse decisions.

Our initial numerical experiments showed that moderate-sized MDPs can be solved using (6), but this approach may be too computationally intensive to solve large problems such as those that arise in the context of medical decision making. This motivated the development of an approximation algorithm that we describe in Section 5, subsequently test on randomly generated problem instances in Section 6, and then apply to a medical decision making problem in the case study in Section 7. The following relaxation of the non-adaptive problem allows us to quantify the performance of our approximation algorithm:

**Proposition 7.** For any policy \( \hat{\pi} \in \Pi \), the weighted value is bounded above by the weighted sum of the optimal values in each model. That is,

\[
\sum_{m \in M} \lambda_m v^m(\hat{\pi}) \leq \sum_{m \in M} \lambda_m \max_{\pi \in \Pi^{MD}} v^m(\pi), \quad \forall \hat{\pi} \in \Pi.
\]

The result of Proposition 7 allows us to evaluate the performance of any MD policy even when we cannot solve the weighted value problem exactly to determine the true optimal policy. We use this result to illustrate the performance of our approximation algorithm in Section 7.

Proposition 7 motivates several connections between robustness and the value of information. First, the upper bound in Proposition 7 is based on the well-known wait-and-see problem in stochastic programming that relaxes the condition that all models must have the same policy.
Second, the expected value of perfect information (EVPI) is the expected value of the wait-and-see solution minus the recourse problem solution:

$$EVPI = \left[ \sum_{m \in M} \lambda_m \max_{\pi \in \Pi_M} v^m(\pi) \right] - \max_{\pi \in \Pi_M} \left[ \sum_{m \in M} \lambda_m v^m(\pi) \right].$$  \(8\)

While the wait-and-see value provides an upper bound, it may prescribe a set of solutions, one for each model, and thus it often does not provide an implementable course of action. Another common approach in stochastic programming is to solve the mean value problem (MVP) which is a simpler problem in which all parameters take on their expected values. In the MMDP, this corresponds to the case where all transition probabilities and rewards are weighted as follows:

$$\bar{p}_t(s'|s,a) = \sum_{m \in M} \lambda_m p^m_t(s'|s,a), \quad \forall s \in S, a \in A, t \in T$$  \(9\)

and

$$\bar{r}_t(s,a) = \sum_{m \in M} \lambda_m r^m_t(s,a).$$  \(10\)

Solving the mean value problem will give a single policy, $\bar{\pi}$, which we will term the mean value solution, with the following expected rewards:

$$W(\bar{\pi}) = \sum_{m \in M} \lambda_m v^m(\bar{\pi}).$$  \(11\)

Thus, we can create a measure of robustness for an MMDP termed the value of the weighted value solution (VWV):

$$VWV = W^* - W(\bar{\pi}),$$  \(12\)

which parallels the well-known value of the stochastic solution (VSS) in stochastic programming (Birge and Louveaux 1997, §4.2). If VWV is low, this implies that there is not much value from solving the MMDP versus the MVP. On the other hand, if VWV is high, this implies that the DM will benefit significantly from solving the MMDP.

While the non-adaptive problem has connections to stochastic programming, it also has connections to POMDPs. The non-adaptive problem can be viewed as the problem of finding the best memoryless controller for this POMDP (Vlassis et al. 2012). Memoryless controllers for POMDPs are defined on the most recent observation only. For an MMDP, this would translate to the DM specifying a policy that is based only on the most recent observation of the state (recall that the DM gets no information about the model part of the state-model pair). Because no history is allowed to be incorporated into the definition of the policy, this policy is permissible for the non-adaptive problem. These connections between MMDPs and stochastic programs and POMDPs allow us to
better understand the complexity and potential solution methods for finding the best solution to the non-adaptive problem.

5. Solution methods

In this section, we will discuss how to leverage the results of Section 4 to solve the non-adaptive problem. Although the MIP formulation of Proposition 6 provides a viable way to exactly solve this class of problems, the result of Proposition 5 motivates the need for a fast approximation algorithm that can scale to large MMDPs.

5.1. Mixed-integer programming formulation

The big-M constraints are an important aspect of the MIP formulation of the weighted value problem. We discuss tightening of the big-M values in Blinded (2018); details are omitted here.

5.2. Weight-Select-Update Heuristic

Next, we discuss our Weight-Select-Update (WSU) heuristic, formalized in Procedure 1, which is a fast approximation algorithm for the non-adaptive problem. WSU generates decision-rules $\hat{\pi}_t \in \Pi^{MD}_t$ stage-wise starting at epoch $T$ and iterating backwards. At epoch $t \in \mathcal{T}$, the algorithm has an estimate of the value for this policy in each model conditioned on the state $s_{t+1}$ at epoch $t+1 \in \mathcal{T}$. This estimate is denoted $\hat{v}^m_{t+1}(s_{t+1})$, $\forall m \in \mathcal{M}, \forall s_{t+1} \in \mathcal{S}$. The algorithm weights the immediate rewards plus the value-to-go for each of the models and then the algorithm selects, for each state, an action that maximizes the sum of these weighted terms and denotes this action $\hat{\pi}_t(s_t)$. Next, the algorithm updates the estimated value-to-go for every state in each model according to the decision rule $\hat{\pi}_t$ at epoch $t \in \mathcal{T}$. This procedure iterates backwards stage-wise until the actions are specified for the first decision epoch.
**Procedure 1** Weight-Select-Update (WSU) heuristic for the non-adaptive problem (5)

**Input:** MMDP

Let $\hat{v}_m^{t+1}(s_{T+1}) = r_m^{t+1}(s_{T+1})$, $\forall m \in \mathcal{M}$

$t \leftarrow T$

while $t \geq 1$

for Every state $s_t \in \mathcal{S}$ do

$$\hat{\pi}_t(s_t) \leftarrow \arg\max_{a_t \in \mathcal{A}} \left\{ \sum_{m \in \mathcal{M}} \lambda_m \left( r_t^m(s_t, a_t) + \sum_{s_{t+1} \in \mathcal{S}} p_t^m(s_{t+1}|s_t, a_t) \hat{v}_t^{m+1}(s_{t+1}) \right) \right\}$$ (13)

end for

for Every model $m \in \mathcal{M}$ do

$$\hat{v}_t^m(s_t) \leftarrow r_t^m(s_t, \hat{\pi}_t(s_t)) + \sum_{s_{t+1} \in \mathcal{S}} p_t^m(s_{t+1}|s_t, \hat{\pi}_t(s_t)) \hat{v}_{t+1}^m(s_{t+1})$$ (14)

end for

$t \leftarrow t - 1$

end while

**Output:** The policy $\hat{\pi} = (\hat{\pi}_1, \ldots, \hat{\pi}_T) \in \Pi^{MD}$

Upon first inspection, it may not be obvious that WSU is not guaranteed to produce the optimal MD policy; however, this heuristic fails to account for the fact that, under a given policy, the likelihood of occupying a specific state could vary under the different models. The result of Proposition 8 shows that ignoring this could lead to sub-optimal selection of actions as illustrated in the proof.

**Proposition 8.** WSU is not guaranteed to produce an optimal solution to the non-adaptive weighted value problem.

Although WSU is not guaranteed to select the optimal action for a given state-time pair, this procedure is guaranteed to correctly evaluate the value-to-go in each model for the procedure’s policy, $\hat{\pi}$. This is because, although the action selection in equation (13) may be suboptimal, the update of the value-to-go in each model in (14) correctly evaluates the performance of this action in each model conditional on being in state $s_t$ at decision epoch $t$. That is, for a fixed policy, policy evaluation for standard MDPs applies to each of the models, separately.

**Proposition 9.** For $|\mathcal{M}| = 2$, if $\lambda_1^m > \lambda_2^m$, then the corresponding policies $\hat{\pi}(\lambda^1)$ and $\hat{\pi}(\lambda^2)$ generated via WSU for these values will be such that

$$v^m(\hat{\pi}(\lambda^1)) \geq v^m(\hat{\pi}(\lambda^2)).$$ (15)
Proposition 9 guarantees that the policies generated using WSU will have values in model $m \in \mathcal{M}$ that are non-decreasing model $m$’s weight, $\lambda_m$. This result is desirable because it allows DMs to know that placing more weight on a particular model will not result in a policy that does worse with respect to that model. Proposition 9 is also useful for establishing the lower bound in the following proposition:

**Proposition 10.** For $|\mathcal{M}| = 2$, any policy generated via WSU will be such that

$$W(\hat{\pi}(\lambda)) \geq \lambda v^1(\pi^2) + (1 - \lambda)v^2(\pi^1).$$

(16)

where $\pi^m$ is the optimal policy for model $m$.

Proposition 10 provides a lower bound on the weighted value of the policy obtained via WSU. While Proposition 7 provides an upper bound that applies for any policy $\pi \in \Pi_M$, the lower bound in Proposition 10 is specific to the policies obtained via WSU for the specific case of 2 models. The bound is generated by appropriately weighting the value obtained by model 1’s optimal policy in model 2 and the value obtained by model 2’s optimal policy in model 1. This establishes an easy way to obtain this bound because it involves solving the 2 models independently and then evaluating these policies.

### 6. Computational Experiments

In this section, we describe a set of computational experiments for comparing solution methods for the adaptive problem and the non-adaptive problem on the basis of run-time and quality of the solution. Our experiments were based on a series of random instances of MMDPs. To generate the random test instances, first the number of states, actions, models, and decision epochs for the problem were defined. Then, model parameters were randomly sampled. In all test instances, it was assumed that the sampled rewards were the same across models, the weights were uninformed priors on the models, and the initial distribution was a discrete uniform distribution across the states. The rewards were sampled from the uniform distribution: $r(s,a) \sim U(0,1), \forall (s,a) \in \mathcal{S} \times \mathcal{A}$. The transition probabilities were obtained by sampling from a uniform distribution so that $\tilde{p}^m(s'|s,a) \sim U(0,1)$. Then, for every $(m,s,a,s') \in \mathcal{M} \times \mathcal{S} \times \mathcal{A} \times \mathcal{S}$, the transition probabilities were normalized so that the row of the transition probability matrix had elements that sum to one:

$$p^m(s'|s,a) := \frac{\tilde{p}^m(s'|s,a)}{\sum_{s'' \in \mathcal{S}} \tilde{p}^m(s''|s,a)}.$$  

(17)

The non-adaptive problem was solved using WSU, MVP, and the MIP formulation. WSU and MVP were implemented using Python 3.5.2. All MIPs were solved using AMPL Version 20150815 and CPLEX 12.6.1.

For the non-adaptive problem, we compared the exact and heuristic solution methods on the basis of run-time and quality of solution. To do so, a base case problem size of 4 states, 4 actions, 4 models, and 4 decision epochs was defined. Then, the size of the problem was varied with respect to the number of states, actions, models, and decision epochs independently to determine the influence of growth in the problem size on the average- and worst-case run times and optimality gaps.

To evaluate the quality of the solutions obtained via the WSU and MVP heuristics, we will compare the weighted value policies obtained via the heuristics ($W_N(\hat{\pi})$) to the optimal value obtained by solving the MIP to within 1% of optimality, $W_N^*$:

$$\text{Gap} = \frac{W_N^* - W_N(\hat{\pi})}{W_N^*} \times 100\%,$$

where $\hat{\pi}$ is the policy obtained from either WSU or MVP. For each problem size tested, the WSU heuristic had a worst-case optimality gap of 1.0% and an average optimality gap being less than 0.01%. The performance of the MVP heuristic had a worst-case optimality gap of 51.9% and an average gap of 3.5%. These results indicate that the WSU heuristic is likely a better approximation method than MVP.

We also compared the time required to solve the instances using the WSU heuristic and the MIP (see Figure 3 in Blinded (2018)). The WSU heuristic was able to generate an policy relatively quickly on these test instances (under 1 CPU second on average) while the average time required to solve the MIP noticeably increases as the size of the problem increases, especially with respect to the number of decision epochs in the MMDP. These results suggest that heuristics may be needed to approximate solutions for larger MMDPs, such as the one presented as a case study in Section 7.

7. Case study: blood pressure and cholesterol management in type 2 diabetes

In this section, we present an MMDP to optimize the timing and sequencing of the initiation of blood pressure medications and cholesterol medications for patients with type 2 diabetes. Here, WSU was used to generate a policy that trades off conflicting estimates of cardiovascular risk from two well-established studies in the medical literature. We begin by providing some context about the problem, the MMDP model, and the parameter ambiguity that motivates its use.

Diabetes is one of the most common and costly chronic medical conditions, affecting more than 25 million adults, or 11% of the adult population in the United States (CDC 2011). Diabetes is associated with the inability to properly metabolize blood glucose (blood sugar) and other
metabolic risk factors that place the patient at risk of complications including coronary heart disease (CHD) and stroke. There are several types of diabetes including type 1 diabetes, in which the patient is dependent on insulin to live, gestational diabetes, which is associated with pregnancy, and type 2 diabetes in which the patient has some ability (albeit impaired) to manage glucose. In this case study we focus on type 2 diabetes, which accounts for more than 90% of all cases.

The first goal, glycemic control, is typically achieved quickly following diagnosis of diabetes using oral medications and/or insulin. Management of cardiovascular risk, the focus of this case study, is a longer term challenge with a complex tradeoff between the cost and harms of medication and the risk of future CHD and stroke events. Patients with diabetes are at much higher risk of stroke and CHD events than the general population. Well-known risk factors include total cholesterol (TC), high density lipids (HDL – often referred to as “good cholesterol”), and systolic blood pressure (SBP). Like blood glucose, the risk factors of TC, HDL, and BP are also controllable with medical treatment. Medications, such as statins and fibrates, can reduce TC and increase HDL. Similarly, there are a number of medications that can be used to reduce blood pressure including ACE inhibitors, ARBs, beta blockers, thiazide, and calcium channel blockers. All of these medications have side effects that must be weighed against the long-term benefits of lower risk of CHD and stroke. An added challenge to deciding when and in what sequence to initiate medication is due to the conflicting risk estimates provided by two well-known clinical studies: the Framingham Heart Study (Wolf et al. 1991, Wilson et al. 1998) and the ACC/AHA assessment of cardiovascular risk (Goff et al. 2014).

7.1. MMDP formulation

The MDP formulation of Mason et al. (2014) was adapted to create an MMDP based on the Framingham risk model (Wolf et al. 1991, Wilson et al. 1998) and the ACC/AHA risk model (Goff et al. 2014). These are the most well-known risk models used by physicians in practice. The state space of the MMDP is a finite set of health states defined by SBP, TC, HDL, and current medications. A discrete set of actions represent the initiation of the two cholesterol medications and 4 classes of blood pressure medications. The objective is to optimize the timing and sequencing of medication initiation based on a weighted reward function that considers QALYs and cost of treatment and complications. For this case study, we will assume that the rewards are the same in each of the models of the MMDP and that only the transition probabilities vary across models. Figure 3 provides a simplified example to illustrate the problem. In the diagram, solid lines illustrate the actions of initiating one or both of the most common medications (statins (ST), ACE inhibitors
(AI)), and dashed lines represent the occurrence of an adverse event (stroke or CHD event), or death from other causes. In each medication state, including the no medication state (∅), patients probabilistically move between health risk states, represented by L (low), M (medium), H (high), and V (very high). For patients on one or both medications, the resulting improvements in risk factors reduce the probability of complications. Treatment actions are taken at a discrete set of decision epochs indexed by $t \in T = \{0, 1, \ldots, T\}$ that correspond to ages 54 through 74 at one year intervals that represent annual preventive care visits with a primary care doctor. States can be separated into living states and absorbing states. Each living state is defined by the factors that influence a patient’s cardiovascular risk: the patient’s TC, HDL, and SBP levels, and medication state. We denote the set of the TC states by $\mathcal{L}_{TC} = \{L, M, H, V\}$, with similar definitions for HDL, $\mathcal{L}_{HDL} = \{L, M, H, V\}$, and SBP, $\mathcal{L}_{SBP} = \{L, M, H, V\}$. The thresholds for these ranges are based on established clinically-relevant cut points for treatment (Cleeman et al. 2001). The complete set of health states is indexed by $\ell \in \mathcal{L} = \mathcal{L}_{TC} \times \mathcal{L}_{HDL} \times \mathcal{L}_{SBP}$.

**Figure 3** An illustration of the state and action spaces of the MDP as illustrated in Mason et al. (2014). In the corresponding MMDP, when medications are initiated (solid lines denote actions), the risk factors are improved and the probability of an adverse event (denoted by the dashed lines) is reduced. The probabilities of adverse events may differ in the different models depending on the risk calculator that was used to estimate the probability.

The set of medication states is $\mathcal{M} = \{\tau = (\tau_1, \tau_2, \ldots, \tau_n) : \tau_i \in \{0, 1\}, \forall i = 1, 2, \ldots, 6\}$ corresponding to all combinations of the 6 medications mentioned above. If $\tau_i = 0$, the patient is not on medication $i$, and if $\tau_i = 1$, the patient is on medication $i$. The treatment effects for medication $i$ are denoted by $\omega_{TC}^{\tau}(i)$, for the proportional reduction in TC, $\omega_{HDL}^{\tau}(i)$, for the proportional change in HDL, and $\omega_{SBP}^{\tau}(i)$, for the proportional change in SBP, as reported in Mason et al. (2014). The living
states in the model are indexed by \((\ell, \tau) \in \mathcal{L} \times \mathcal{M}\). The absorbing states are indexed by \(d \in \mathcal{D} = \{\mathcal{D}_S, \mathcal{D}_{CHD}, \mathcal{D}_O\}\) represent having a stroke, \(\mathcal{D}_S\), having a CHD event, \(\mathcal{D}_{CHD}\), or dying, \(\mathcal{D}_O\). The action space depends on the history of medications that have been initiated in prior epochs. For each medication, at each epoch, medication \(i\) can be initiated \((I)\) or initiation can be delayed \((W)\):

\[
A(\ell, m_i) = \begin{cases} 
\{I_i, W_i\} & \text{if } \tau_i = 0, \\
\{W_i\} & \text{if } \tau_i = 1,
\end{cases}
\]

and \(A(\ell, \tau) = A(\ell, \tau_1) \times A(\ell, \tau_2) \times \ldots \times A(\ell, \tau_n)\). Action \(a \in A(\ell, \tau)\) denotes the action in state \((\ell, \tau)\).

If a patient is in living state \((\ell, \tau)\) and takes action \(a\), the new medication state is denoted by \(\tau'\), where \(\tau'_i\) is set to 1 for any medications \(i\) that are newly initiated by action \(a\); \(\tau'_i = \tau_i\) for all medications \(i\) which are not newly initiated. Once medication \(i\) is initiated, the associated risk factor is modified by the medication effects denoted by \(\omega^{TC}(i)\), \(\omega^{HDL}(i)\), and \(\omega^{SBP}(i)\), resulting in a reduction in the probability of a stroke or CHD event. Two types of transition probabilities are incorporated into the model: probabilities of transition among health states and the probability of events (fatal and nonfatal). At epoch \(t\), \(\bar{p}_t(d|\ell)\) denotes the probability of transition from state \((\ell, \tau) \in \mathcal{L} \times \mathcal{M}\) to an absorbing state \(d \in \mathcal{D}\). Given that the patient is in health state \(\ell \in \mathcal{L}\), the probability of being in health state \(\ell'\) in the next epoch is denoted by \(q_t(\ell'|\ell)\). The health state transition probabilities, \(q_t(\ell'|\ell)\), were computed from empirical data for the natural progression of BP and cholesterol adjusted for the absence of medication (Denton et al. 2009). We define \(p_t(i|\ell)\) to be the probability of a patient being in state \(j \in \mathcal{L} \cup \mathcal{D}\) at epoch \(t + 1\), given the patient is in living state \((\ell, \tau)\) at epoch \(t\). The transition probabilities can be written as:

\[
p_t(j|i) = \begin{cases} 
[1 - \sum_{d \in \mathcal{D}} \bar{p}_t(d|i)] q_t(j|i) & \text{if } i, j \in \mathcal{L}, \\
\bar{p}_t(j|i) & \text{if } i \in \mathcal{L}, j \in \mathcal{D}, \\
1 & \text{if } i = j \in \mathcal{D}, \\
0 & \text{otherwise.}
\end{cases}
\]

The two models of the MMDP represent the different cardiovascular risk calculators used to estimate the transition probabilities to the absorbing states: \(\bar{p}_t(i|d)\) for \(i \in \mathcal{L}, d \in \mathcal{D}\). We will refer to the model using the ACC/AHA study as model \(A\) and the model using Framingham study as model \(F\). We weight these models by \(\lambda_A \in [0,1]\) and \(\lambda_F := 1 - \lambda_A\) respectively. We estimate of all other cause mortality based take from the Centers for Disease Control and Prevention life tables (Arias et al. 2011). The reward \(r_t(\ell, \tau)\) for a patient in health state \(\ell\) at epoch \(t\) is:

\[
r_t(\ell, \tau) = Q(\ell, \tau),
\]
where $Q(\ell, \tau) = 1 - d^{\text{MED}}(\tau)$ is the reward for one quality-adjusted life year (QALY). QALYs are elicited through patient surveys, and are commonly used for health policy studies (Gold et al. 2002). The utility decrement factor, $d^{\text{MED}}(\tau)$, represents the estimated decrease in quality of life due to the side effects associated with the medications in $\tau$.

### 7.2. Results

Using the MMDP described above, we evaluated the performance of the solutions generated via WSU in terms of computation time and the objective function of QALYs until first event. We also discuss the policy associated with the solution generated using WSU when the weights are treated as an uninformed prior on the models. The MMDP had 4099 states, 64 actions, 20 decision epochs, and 2 models.

**Table 1** Time to approximate a solution to the weighted problem using the Weight-Select-Update algorithm and to solve each of the nominal models using standard dynamic programming, in CPU seconds.

<table>
<thead>
<tr>
<th>Model</th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSU with $\lambda_F = \lambda_A = 0.5$</td>
<td>10.98</td>
<td>11.08</td>
</tr>
<tr>
<td>Standard DP, Framingham Model</td>
<td>8.70</td>
<td>8.77</td>
</tr>
<tr>
<td>Standard DP, ACC/AHA Model</td>
<td>8.98</td>
<td>9.00</td>
</tr>
</tbody>
</table>

Table 1 shows the computation time required to run WSU with $\lambda_F = \lambda_A = 0.5$, as well as the time to required to solve the Framingham model and the ACC/AHA model using standard dynamic programming, for the female and male problem parameters. While WSU requires more computation time than standard dynamic programming for each of the individual models, WSU does not take more computation time than the total time for solving both of the nominal models.

Figure 4 shows the performance of the policies generated using WSU when evaluated in the ACC/AHA model and in the Framingham model, as well as the weighted value of these two models for the corresponding choice of the weight on the Framingham model, $\lambda_F$. When $\lambda_F = 100\%$, WSU finds the optimal policy for the Framingham model which is why the maximum the Framingham value is achieved at $\lambda_F = 100\%$. Of the WSU policies, the worst value in the ACC/AHA model is achieved at this point because the algorithm ignores the performance in the ACC/AHA model. Analogously, when $\lambda_F = 0\%$, WSU finds the optimal policy for the ACC/AHA model which is why the performance in the ACC/AHA model achieves its maximum and the performance in the Framingham model is at its lowest value at this point. For values of $\lambda_F \in (0, 1)$, WSU generates policies that trade-off the performance between these two models. As supported by Proposition 9, WSU has the desirable property that the performance in model $m$ is non-decreasing in $\lambda_m$. For women, using the Framingham model’s optimal policy leads to a severe degradation in performance with respect to the ACC/AHA model. In contrast, WSU is able to generate policies that do not sacrifice too
Figure 4  The performance of the policies generated using the Weight-Select-Update heuristic for the MMDP for treatment of men (Figure 4a) and women (Figure 4b). For each choice of the weight on the Framingham model in WSU, the graph shows the performance of these policies with respect to three different metrics: the performance in the ACC/AHA model (light grey), the performance in the Framingham model (dark grey), and the weighted value (black).
much performance in the ACC/AHA model in order to improve performance in the Framingham model. The results for women clearly illustrate why taking a max-min approach instead of the MMDP approach can be problematic in some cases. To see this, note that the Framingham model’s optimal policy is a solution to the max-min problem because $v^F(\pi^*_F) < v^A(\pi^*_F)$ and thus no policy will be able to achieve a better value than $\pi^*_F$ in the Framingham model. However, Figure 4(b) shows that this policy leads to a significant degradation in performance in the ACC/AHA model relative to that model’s optimal policy $\pi^*_A$. This demonstrates why taking a max-min approach, which is common in the robust MDP literature as pointed out in Section 2, can have the unintended consequence of ignoring the performance of a policy in all but one model in some cases.

By taking the weighted value approach with nontrivial weights on the models, the decision-maker is forced to consider the performance in all models. By generating policies using WSU by varying $\lambda_F \in (0, 1)$, the DM can strike a balance between the performance in the ACC/AHA model and the Framingham model.

Table 2 illustrates the effects of ambiguity in the cardiovascular disease MMDP when the model weights are treated as uninformed priors ($\lambda_A = \lambda_F = 0.5$). In this problem, the “wait-and-see” solution corresponds to the situation where the DM is allowed to observe which risk calculator is governing the cardiovascular risk of the patient before deciding how to treat the patient. The weighted value of the WSU solution is found by evaluating the policy obtained via the WSU heuristic in the weighted value problem. Similarly, the weighted value of the mean value solution ($W(\bar{\pi})$) is found by evaluating the policy obtained by solving the mean value problem in the weighted value problem. Because $WSU \leq W^*$, it follows that $VWV = W^* - W(\bar{\pi}) \leq WSU - W(\bar{\pi})$. For both men and women, the value of the weighted value solution, VWV, is very low when $\lambda_F = \lambda_A = 0.5$, suggesting that there is not much to be gained in solving the weighted value problem over the mean value problem in this case. The lower bound on EVPI in the table follows from $EVPI = WS - W^* \geq WS - WSU$. The lower bound on the EVPI suggests that at least 28 QALYs per 1000 men and 131.8 QALYs per 1000 women could be saved if there were no ambiguity in the cardiovascular risk
Figure 5  The percentage of patients who have not died or had an event by the specified age that will be on a medication under each of three different treatment policies: the ACC/AHA model’s optimal policy, the Framingham model’s optimal policy, and a policy generated via WSU with $\lambda_F = \lambda_A = 50\%$, as evaluated in the Framingham model.

of the patient. Estimates such as this provide insight into the value of future studies that could elucidate the ambiguity.
Figures 5(a) and 5(b) illustrate medication use for male and female patients, respectively, under three different policies: the ACC/AHA model’s optimal policy, the Framingham model’s optimal policy, and a policy generated via WSU with $\lambda_F = \lambda_A = 50\%$. For men, the optimal policy for Framingham and the optimal policy for ACC/AHA agree that all men should start statins immediately, which could be explained by the relatively low disutility and high risk reduction of statins in both models. However, the models disagree in the use of fibrates and the 4 classes of blood pressure medications. The optimal policy for the ACC/AHA model suggests that all men should start fibrates immediately, suggesting that cholesterol control is important in the ACC/AHA model. However, fibrates are less commonly prescribed under the Framingham model’s optimal policy with about two-thirds of men on this medication by age 65. The policy generated with WSU agrees with the ACC/AHA policy’s more extensive use of fibrates which may suggest that focusing on cholesterol control could be a good strategy in both models. Among the blood pressure medications, there are some disagreements between the optimal policies of the two models, with the most distinct being for the use of calcium channel blockers. This is likely to be due to the relatively high disutility (from side effects of calcium channel blockers) and low risk reduction associated with this medication. In the ACC/AHA model, the risk reduction of calcium channel blockers is worth the disutility in many cases, but in the Framingham model, there are few instances in which the disutility associated with this medication is worth the gain in QALYs. The policy generated with WSU generates a policy that strikes a balance between these two extremes. While the differences are not quite as extreme, WSU also generates a policy that balances the utilization of thiazides prescribed by each model’s optimal policy. For the other classes of blood pressure medications, both models agree that these medications should be commonly used for men, but disagree in the prioritization of these medications. The ACC/AHA model tends to utilize these medications more at latter ages, while the Framingham model starts more men on these medications early. Interestingly, WSU suggests that starting ACE/ARBs and beta blockers earlier is a good strategy in both models.

For women, the optimal policy for Framingham and the optimal policy for ACC/AHA agree that all women should be on a statin by age 57. The models mostly agree that relatively few women should start taking ACE/ARBs or calcium channel blockers. These results are not surprising as statins have low disutility and high risk reduction in both models, making them an attractive medication to use to manage a patients cardiovascular risk, while calcium channel blockers and ACE/ARBs are the two medications with lowest expected risk reduction in both models. The models disagree in how to treat women with thiazides, beta blockers, and fibrates. Beta blockers and thiazides have a higher estimated risk reduction in the Framingham model than in the
ACC/AHA model, which may be why these medications are considered good candidates to use in the Framingham model but not in the ACC/AHA model. WSU finds a middle ground between the use of thiazides and beta blockers in the two models, but suggests more use of ACE/ARBs for some women.

In summary, the results of this case study illustrate how the policy generated by WSU trades off performance in the ACC/AHA and Framingham models. This information could be useful for decision-makers who are tasked with designing screening and treatment protocols in the face of conflicting information from the medical literature.

8. Conclusions

In this article, we addressed the following research questions: (1) how can we improve stochastic dynamic programming methods so that they can account for parameter ambiguity? (2) how much benefit is there to mitigating the effects of ambiguity? To address the first question, we introduced the MMDP, which allows for multiple models of the reward and transition probability parameters. Although our complexity results establish that the MMDP model is computationally intractable, our analysis shows there is promising structure that can be exploited to create exact methods and fast approximation algorithms for solving the MMDP. To address the second research question, we established connections between concepts in stochastic programming and the MMDP that quantify the impact of ambiguity on an MDP. We showed that the non-adaptive problem can be viewed as a two-stage stochastic program in which the first-stage decisions correspond to the policy and the second-stage decisions correspond to the value-to-go in each model under the specified policy. This characterization provided insight into a formulation of the non-adaptive problem as an MIP corresponding to the deterministic equivalent problem of the aforementioned two-stage stochastic program. We showed the adaptive problem is a special case of a POMDP and described solution methods that exploit the structure of the belief space for computational gain.

We evaluated the performance of our solution methods using a large set of randomly-generated test instances and also an MMDP of blood pressure and cholesterol management for type 2 diabetes as a case study. The WSU heuristic performed very well across the randomly-generated test cases while the Mean Value Solution had some instances with large optimality gaps indicating that simply averaging multiple models should be done with caution. These randomly-generated test instances also showed that there was very little gain from adaptive optimization of policies over non-adaptive optimization for the problem instances considered.
In the case study, we solved an MMDP consisting of two models which were parameterized according to two well established, but conflicting, studies from the medical literature. The case study showed that WSU can be used to generate a policy for a large-scale problem instance that trades off performance between these two models, and showed that gaining more information about cardiovascular risk could lead to an increase in QALYs for both men and women. This case study also reinforced the finding from the smaller random instances that adaptive learning provides little benefit over the non-adaptive approach. For the most part, the policies generated via the WSU heuristic found a balance between the medication usage in each of the models. However, for men, the WSU heuristic suggested that more aggressive use of thiazides and ACE/ARBs would be allow for a better balance in performance in both models. For women, the WSU heuristic generated a policy that is more aggressive in cholesterol control than the Framingham model’s optimal policy and more aggressive in blood pressure control than the ACC/AHA model’s optimal policy.

There are open opportunities for future work that builds off of the MMDP formulation. In this article, we showed that the MMDP can provide a rigorous way to account for ambiguity in transition probability matrices in an MDP. We focused on ambiguity due to conflicting sources for model parameters in the case study. However, future work could study the performance of the MMDP formulation for addressing statistical uncertainty compared to other robust formulations that have attempted to mitigate the effects of this kind of uncertainty. Another opportunity is to apply this approach to other diseases, such as diabetes, breast cancer and prostate cancer, for which multiple models have been developed. Other future work might extend this concept to partially-observable MDPs and infinite-horizon MDPs, which are both commonly used for medical decision making. Further, the bounds developed for the WSU were in the context of a 2 model MMDP, but it would be valuable to develop bounds for WSU for $|\mathcal{M}| > 2$. Finally, the MMDP introduced in this article was limited to a finite number of models, however future work may consider the possibility of a countably infinite number of models.

In summary, the MMDP is a new approach for incorporating parameter ambiguity in MDPs. This approach allows DMs to explicitly trade off conflicting models of problem parameters to generate a policy that performs well with respect to each model while keeping the same level of complexity as each model’s optimal policy. The MMDP may be a valuable approach in many application areas of MDPs, such as medicine, where multiple sources are available for parameterizing the model.

References


