Risk-Sensitive Decision Making in the Presence of Model Uncertainty

Apoorva Sharma [1] **James Harrison** [2] **Marco Pavone** [1]

Abstract

Planning under model uncertainty is a fundamental problem across many applications of decision making and learning. For example, a robot interacting with items in the world wants to optimally disambiguate between physical models, generate good quality plans, and be robust to incorrect model beliefs. In this paper, we propose the Robust Adaptive Monte Carlo Planning (RAMCP) algorithm, which allows computation of risk-sensitive Bayes-adaptive policies that optimally trade off exploration and exploitation. RAMCP formulates the risk-sensitive planning problem as a two-player zero-sum game, where an adversary perturbs the agent's belief over the models. Importantly, the RAMCP algorithm converges to an optimal risk-sensitive policy without having to rebuild the search tree as the underlying belief over models is perturbed, so computation of these risk-sensitive policies is only marginally more expensive that computation of risk-neutral policies. We prove that the RAMCP algorithm asymptotically converges to an optimal risk-sensitive Bayes adaptive policy. We demonstrate the performance of this algorithm on a challenging n-pull multi-armed bandit problem, as well as a patient treatment scenario.

1. Introduction

Consider an autonomous vehicle driving down a highway, reasoning about and planning with respect to the future actions of nearby cars. In the case where the intent of a nearby vehicle is known, the autonomous vehicle may leverage probabilistic models of driver actions and vehicle dynamics, based on experience, to build a distribution over future states (see for example, Schmerling et al. (2017)). The intent of the other agent, however, is not known. The planning therefore must be over a set of possible models, where for each model, a stochastic transition function is known. Such problems are ubiquitous in planning and learning.

The Bayesian approach to dealing with this model ambiguity is to maintain a *belief distribution* over possible models. This belief distribution can be updated by employing Bayes' rule as the agent observes the state transitions in the environment. The agent may then act with respect to the posterior distribution over models. Simply optimizing the expectation over this posterior distribution, however, may be problematic. A policy with low cost in expectation, could still suffer a high cost on a model with low probability mass, leading to poor performance if this high-cost model was indeed the true underlying model of the system. This motivates the need to incorporate distributional robustness into the optimization, allowing a practitioner to trade off optimality in expectation to mitigate failures from inaccurate prior distributions. We wish to bring this notion of robustness to planning under model uncertainty, enabling an autonomous agent to disambiguate between possible models of the environment, while simultaneously acting robustly with respect to its subjective belief over these models.

Statement of Contributions In this work we present Robust Adaptive Monte Carlo Planning (RAMCP), an approach to risk-sensitive planning in discrete MDPs for which a collection of models is known, but the true underlying model in this set is not. This approach is based on adversarially perturbing the belief over models iteratively during the construction of a look-ahead tree. While naive perturbations of the belief may result in instability and non-convergence of the value function, we prove that RAMCP asymptotically converges to an optimal risk-sensitive, history-dependent policy. This policy optimally trades off exploration, to improve model knowledge, and exploitation. Moreover, because we are iteratively perturbing the belief during look-ahead tree construction, the tree does not have to be rebuilt for changing beliefs, and thus the RAMCP algorithm only results in a minor increase in computational cost relative to a risk-neutral look-ahead tree-based planning algorithm such as POMCP (Silver & Veness, 2010). We demonstrate the performance of RAMCP on a bandit problem and a patient treatment scenario.

[1]Department of Aeronautics and Astronautics, Stanford University, Stanford, CA, USA [2]Department of Mechanical Engineering, Stanford University, Stanford, CA, USA. Correspondence to: Apoorva Sharma <apoorva@stanford.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

2. Background

We wish to control an agent in a system defined by the MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, T, R, H)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $T(s'|s,a)$ is the transition function, $R(s,a,s')$ is the stage-wise reward function, and H is the problem horizon. We assume that the exact transition dynamics T depend on a parameter θ, and denote this dependence as T_θ. In this work, we consider problems in which there is a finite collection of parameters, $\Theta = \{\theta_i\}_{i=1}^M$. We will write our prior belief over Θ as b_{prior}. While limiting ourselves to discrete distributions over model parameters is somewhat restrictive, these simplifications are common in, for example, sequential Monte Carlo (Doucet et al., 2001). Indeed, computing continuous posterior distributions exactly is often intractable, so this discrete approximation is often necessary (Guez et al., 2014).

While acting within this MDP with uncertainty over models, an optimal policy will aim to simultaneously *explore*, to try to disambiguate between possible underlying models, and *exploit*, to maximize cumulative reward. As observed state transitions enable computing a posterior distribution over environments, an optimal policy will not necessarily be Markovian (Asmuth & Littman, 2012). Writing the history in an environment at time t as $h_t = (s_0, a_0, \ldots, s_t)$ and given a prior distribution b_{prior}, we can define optimal behavior in the Bayesian setting. Let $\mathcal{H}$ denote the set of possible histories for a given MDP. Then, we will write the set of stochastic history-dependent polices $\pi : \mathcal{H} \times \mathcal{A} \to [0,1]$ as Π. Let

$$V(h, \pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{H-1} R(s_t, a_t, s_{t+1}) | h_0 = h \right]$$

denote the value function associated with policy π, for history h. A history-dependent policy π^* is said to be Bayes-optimal with respect to the prior over models if its associated value function $V(\{s\}, \pi^*) = \sup_{\pi \in \Pi} V(\{s\}, \pi)$ (Martin, 1967).

By augmenting the state in the MDP $\mathcal{M}$ at time t with the history h_t, the *Bayes-Adaptive Markov Decision Process* (BAMDP) is formed (Duff & Barto, 2002). The optimal policy in this augmented MDP is the Bayes-optimal policy (Martin, 1967). In general, optimizing these policies is computationally difficult. Offline global value approximation approaches, typically based on offline POMDP solution methods, scale poorly to large state spaces (Ghavamzadeh et al., 2015). Online approaches (Wang et al., 2005; Guez et al., 2013; Chen et al., 2016) use tree search with heuristics to either simplify the problem or guide the search.

Most previous work in policy optimization for BAMDPs has focused on optimizing performance in expectation, and thus does not offer a notion of robustness to misspecified priors (Guez et al., 2013). Robust MDPs are posed as MDPs with uncertainty sets over state transitions, and approaches to this problem aim to optimize the worst-case performance over all possible transition models (Nilim & El Ghaoui, 2005). Directly applying this minimax approach to the Bayesian setting is overly conservative because their worst-case analysis does not consider the probability associated with each model. Moreover, because it is typically not desirable to collapse the belief over any model to zero in practice, a robust formulation would likely always act with respect to the worst performing model. In short, the issue arises because the safety guarantees of robust reinforcement learning are based on *sets* of models, but the Bayesian approach instead keeps *distributions* over models.

Tools from risk theory can be used to achieve tunable, distribution-dependent conservatism. A key concept in risk theory is that of a *coherent risk metric*. Given a random cost variable Z, a *risk metric* is a function $\rho(Z)$ that maps the uncertain cost to a real scalar, which encodes a preference model over uncertain outcomes where lower values of $\rho(Z)$ are preferred. Coherent risk metrics are defined as follows:

Definition 1 (Coherent Risk Metrics)**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and let $\mathcal{Z}$ be the space of random variables on Ω. A coherent risk metric (CRM) is a mapping $\rho : \mathcal{Z} \to \mathbb{R}$ that obeys the following four axioms. For all $Z, Z' \in \mathcal{Z}$:*
A1. Monotonicity: $Z \leq Z' \implies \rho(Z) \leq \rho(Z')$
A2. Translation invariance: $\forall a \in \mathbb{R}, \rho(Z+a) = \rho(Z)+a$
A3. Positive homogeneity: $\forall \lambda \geq 0, \rho(\lambda Z) = \lambda \rho(Z)$
A4. Subadditivity: $\rho(Z+Z') \leq \rho(Z) + \rho(Z')$

Note that this definition is stated in terms of costs, as opposed to rewards. These axioms, originally proposed in (Artzner et al., 1999), ensure a notion of rationality in risk assessments. We refer the reader to (Majumdar & Pavone, 2017) for a more thorough discussion of why coherent risk metrics are a useful tool in decision making. While expectation and worst-case are two possible coherent risk metrics, the set of CRMs is a rich class of metrics including the Conditional Value at Risk (CVaR) metric popular in mathematical finance (Majumdar & Pavone, 2017). Recently, the connections between risk sensitivity and robustness have been examined. Chow et al. (2015) showed that optimizing the CVaR metric on the total cost in an MDP can be thought of as robust planning while allowing multiplicative disturbances to the transition probabilities at every step, with the constraint that the product of the disturbances over the planning period is bounded.

3. Problem Statement

We aim to compute a history dependent policy π which is optimal according to a coherent risk metric over model uncertainty. To make this objective more concrete, let

$\tau = (s_0, a_0, \ldots, s_{H-1}, a_{H-1}, s_H)$ be particular trajectory realization. Notice that the probability of a given trajectory depends on both the choice of policy π and the transition dynamics of the MDP, T_θ. Let $q^\pi_{\theta_i}(\tau) = p(\tau|\theta_i, \pi)$ represent this conditional distribution over trajectories for a given policy π and dynamics model θ_i. The cumulative reward of a given trajectory $J(\tau)$ can be calculated by summing the stage-wise rewards

$$J(\tau) = \sum_{t=0}^{H-1} R(s_t, a_t, s_{t+1}). \tag{1}$$

Note that since the distribution over τ is governed by the stochasticity in each environment as well as the uncertainty over environments, the distribution of the total cost of a trajectory $J(\tau)$ is as well.

In this work, we focus our attention to risk-sensitivity with respect to the randomness from model uncertainty only. Concretely, we can write the objective as

$$\Pi^* = \arg\max_{\pi} \quad \rho\left(\mathbb{E}_{\tau\sim q^\pi_{\theta_i}}\left[J(\tau)\right]\right), \tag{2}$$

where the risk metric is with respect to the random variable induced by the distribution over models. Note that Π^* denotes the set of optimal policies.

The above objective, in which the risk sensitivity is over the expected cost of each model, is unusual in the risk-sensitive reinforcement learning literature. Typically, the objective is stated as the risk metric applied to the total reward random variable (Tamar et al., 2017), meaning directly to the value as opposed to the expected value for each model. We believe the objective (2) makes sense the BAMDP context. The robustness provided by risk sensitivity is primarily of value when the knowledge of the distribution is poor; that is to say that risk sensitivity induces distributional robustness. This is useful for beliefs over models, as the true distribution has mass concentrated entirely on one model, so effectively any belief will be incorrect. These beliefs are updated online and are thus susceptible to noise. Comparatively, the dynamics model for each θ_i is assumed to be well characterized and not updated online [JH: should polish this].

4. Robust Adaptive Monte Carlo Planning

4.1. Reformulation as a Zero-Sum, Two Player Game

Several recent works have viewed risk-sensitive policy optimization as a two-player game. In Robust Adversarial Reinforcement Learning (Pinto et al., 2017), the CVaR metric is used to motivate their proposed optimization scheme, which trains the agent while simultaneously training an adversary which can apply disturbances to hinder the performance of the agent. Chen & Bowling (2012) consider k-of-N measures, an approximation of the CVaR metric for continuous distributions, and mathematically formulate the optimization of such a metric as a two-player zero-sum game. In this work, we derive a similar game-theoretic formulation for the general class of coherent risk measures on discrete distributions.

Our reformulation of the objective stems from a universal representation theorem which all coherent risk metrics (CRMs) satisfy.

Theorem 1 (Representation Theorem for Coherent Risk Metrics (Artzner et al., 1999))**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω is a finite set with cardinality $|\Omega|$, $\mathcal{F}$ is a $\sigma-$algebra over subsets (i.e., $\mathcal{F} = 2^\Omega$), probabilities are assigned according to $\mathbb{P} = (p(1), \ldots, p(|\Omega|))$, and $\mathcal{Z}$ is the space of random variables on Ω. Denote by $\mathcal{C}$ the set of valid probability densities:*

$$\mathcal{C} := \left\{ \zeta \in \mathbb{R}^{|\Omega|} \mid \sum_{i=1}^{|\Omega|} p(i)\zeta(i) = 1, \zeta \geq 0 \right\}. \tag{3}$$

Define $p_\zeta \in \mathbb{R}^{|\Omega|}$ as $p_\zeta(i) = p(i)\zeta(i), i = 1, \ldots, |\Omega|$. A risk metric $\rho : \mathcal{Z} \to \mathbb{R}$ with respect to the space $(\Omega, \mathcal{F}, \mathbb{P})$ is a coherent risk metric if and only if there exists a compact convex set $\mathcal{B} \subset \mathcal{C}$ such that for any $Z \in \mathcal{Z}$:

$$\rho(Z) = \max_{\zeta\in\mathcal{B}} \mathbb{E}_{p_\zeta}[Z] = \max_{\zeta\in\mathcal{B}} \sum_{i=1}^{|\Omega|} p(i)\zeta(i)Z(i). \tag{4}$$

Again, note that the above is stated in terms of cost as opposed to reward. For the case of rewards, the maximization over ζ becomes a minimization. This theorem offers an interpretation of CRMs as a worst-case expectation over a set of densities $\mathcal{B}$, often referred to as the *risk envelope*. In this work we focus on *polytopic risk metrics*, for which the envelope $\mathcal{B}$ is a polytope (Chow & Pavone, 2014; Majumdar et al., 2017). For this class of metrics, the constraints on the maximization in Equation 4 become linear in the optimization variable ζ, and thus solving for the value of the risk metric becomes a tractable linear programming problem. Polytopic risk metrics constitute a broad class of risk metrics, encompassing risk neutrality, worst-case assessments, as well as the CVaR metric often used in financial applications.

Through the representation theorem for coherent risk metrics (Equation 4), we can understand Equation 2 as applying an adversarial reweighting ζ to the distribution over models b_{prior}. Let $b_{\text{adv}}(i) = b_{\text{prior}}(i)\zeta(i),\ i = 1, \ldots, M$ represent this reweighted distribution. Thus, the optimization problem we wish to solve may be written

$$\Pi^* = \arg\max_{\pi} \quad \min_{\zeta\in\mathcal{B}} \mathbb{E}_{\theta\sim b_{\text{adv}}}\left[\mathbb{E}_{\tau\sim q^\pi_\theta}\left[J(\tau)\right]\right], \tag{5}$$

where again Π^* denotes the set of optimal policies. We are generally interested simply in finding a policy within this set, as opposed to the full set. Note that this takes the form of a two player zero-sum game between the agent (the maximizer) and an adversary (the minimizer). One play of this game corresponds to the following three step sequence:

1. The adversary acts according to its strategy, choosing b_{adv} from the risk envelope.
2. Chance chooses $\theta \sim b_{\text{adv}}$.
3. The agent acts according to its strategy, or policy, $\pi(h)$ in the MDP with dynamics T_θ.

The action of the adversary is to choose a perturbation to the belief distribution from the polytope of valid disturbances that minimizes the expected performance of the agent. The agent seeks to compute a the best performing policy, taking into account this belief perturbation. The solution to Equation 5 is therefore the optimal Nash equilibrium of the two player game, which we denote as $(b^*_{\text{adv}}, \pi^*)$ In the next section, we will concretize this problem and present the RAMCP algorithm for computing this Nash equilibrium.

4.2. Tractable Methods for Approximately Computing Nash Equilibria

Having formulated our problem statement as a two-player zero-sum game, we leverage tools developed in algorithmic game theory to efficiently compute Nash equilibria. For two-player, zero-sum games, a Nash equilibrium can be directly computed by solving a linear program of size proportional to the strategy space of each player. In the context of our problem statement, the agent's strategy space is the space of all history dependent policies, and thus solving for a Nash equilibrium directly is computationally intractable. To compute the Nash equilibria of the game defined by Equation 2, we apply iterative techniques which converge to equilibria over repeated simulations of the game.

Work in algorithmic game theory has developed formal conditions on strategy updates which guarantee that such iterative schemes indeed converge to a Nash equilibrium. Fictitious Play (Brown, 1949) is a process in which players repeatedly play a game and update their strategies toward the best-response to the average strategy of their opponents. Leslie & Collins (2006) introduced Generalized Weakened Fictitious Play (GWFP), which allows for computation of approximate best-responses but maintains convergence guarantees, and thus has worked well for large-scale extensive form games (Heinrich et al., 2015). GWFP converges to Nash equilibria in several classes of games, including two-player zero-sum games such as Problem 5. In the risk-sensitive BAMDP, the GWFP updates to the adversary strategy b and agent strategy π are:

$$b_{k+1} = (1 - \alpha_{k+1})b_k + \alpha_{k+1}\text{BR}_\epsilon(\pi_k) \tag{6}$$

$$\pi_{k+1} = (1 - \alpha_{k+1})\pi_k + \alpha_{k+1}\text{BR}_\epsilon(b_k) \tag{7}$$

where $\text{BR}_\epsilon(\sigma)$ represents an ϵ-suboptimal best response to strategy σ, and α_{k+1} is an update coefficient chosen such that $\sum_{k=1}^{\infty} \alpha_k = \infty$ and $\lim_{k\to\infty} \alpha_k = 0$. We will defer a formal definition of an ϵ-best response to the supplementary material (Section A), but provide a brief description here. If an opposing player chooses strategy σ, then an ϵ-best response to σ is a strategy such that the player obtains a payoff (or cumulative reward) within ϵ of that of an optimal response (which is itself referred to as a best response). While $\text{BR}_\epsilon(\sigma)$ is typically used to refer to the set of ϵ-best responses, we will use this to refer to a strategy within this set in this work. In this work, we choose $\alpha_k = 1/k$, as in standard Fictitious Play. This choice has the consequence that the strategies of the represent running averages of the best-response strategies, a property we leverage in our algorithm. Initial values of the belief and policy may be chosen arbitrarily.

The adversarial best response $\text{BR}_\epsilon(\pi_k)$ can be computed by solving the linear program:

$$\begin{aligned} \min_{b,\zeta} \quad & \sum_{i=1}^{M} \hat{V}_{\pi_k}(i) b(i) \\ \text{s.t.} \quad & b_{\text{prior}}(i)\zeta(i) = b(i), \quad i = 1, \dots, M \\ & \zeta \in \mathcal{B} \end{aligned} \tag{8}$$

where $\mathcal{B}$ is the polytopic risk envelope, and $\hat{V}_{\pi_k}(i)$ is an estimate of $V_{\pi_k}(i) := \mathbb{E}_{\tau\sim q^{\pi_k}_{\theta_i}}[J(\tau)]$. The suboptimality of the solution of this LP is bounded by the error in $\hat{V}_{\pi_k}(i)$.

The agent's best response $\text{BR}_\epsilon(b_k)$ is a history dependent policy

$$\pi(h) = \arg\max_a \hat{Q}_{b_k}(h, a) \tag{9}$$

where $\hat{Q}_{b_k}(h, a)$ is an estimator of $Q^*_{b_k}(h, a)$, the value of taking action a at history h, then acting optimally when θ is drawn from b_k at the start of the episode.

4.3. Estimating $V_{\pi_k}(i)$ and $Q^*_{b_k}(h, a)$

Carrying out the GWFP process to solve this problem requires estimates $\hat{V}_{\pi_k}(i)$ and $\hat{Q}_{b_k}(h, a)$ at every iteration k. Computing these quantities exactly is intractable, and thus we turn to sampling based methods for estimation. To compute $\hat{V}_{\pi_k}(i)$, we can average the total reward accrued on multiple rollouts of policy π_k on model θ_i, obtaining a Monte Carlo estimate of $\mathbb{E}_{\tau\sim q^{\pi_k}_{\theta_i}}[J(\tau)]$. Computing $\hat{Q}_{b_k}(h, a)$ is equivalent to approximately solving the BAMDP induced by distribution b_k. Many sampling based methods exist to estimate the optimal Q function in a BAMDP.

Guez et al. (2013) showed that performing Monte-Carlo tree search where the dynamics parameter θ is drawn from

Algorithm 1 Robust Adaptive Monte Carlo Planning

1: **function** SEARCH(s_0, b_{prior})
2: $\hat{V}_\pi(i) \leftarrow 0$ for all $i = 1, \ldots, M$
3: $k \leftarrow 0$
4: $b_{\text{adv}} \leftarrow b_{\text{prior}}$
5: $\text{BR}_\epsilon(\pi) \leftarrow b_{\text{prior}}$
6: **while** within computational budget **do**
7: $k \leftarrow k + 1$
8: $w \leftarrow M \cdot \text{BR}_\epsilon(\pi)$
9: **for** $i = 1$ **to** M **do**
10: $J \leftarrow$ ESTIMATEV(s_0, θ_i, H, $w(i)$, c)
11: $\hat{V}_\pi(i) \leftarrow \hat{V}_\pi(i) + \frac{J - \hat{V}_\pi(i)}{k}$
12: **end for**
13: $\text{BR}_\epsilon(\pi) \leftarrow$ solution to linear program (8)
14: $b_{\text{adv}} \leftarrow b_{\text{adv}} + \frac{\text{BR}_\epsilon(\pi) - b_{\text{adv}}}{k}$
15: **end while**
16: **return** $\pi_{\text{avg}} =$ AVGACTION(h) for all h
17: **end function**

Algorithm 2 EstimateV

18: **function** ESTIMATEV(h, θ, d, w, c)
19: **if** $d <= 0$ **or** h is terminal **then**
20: **return** 0
21: **end if**
22: **if** $N(h) = 0$ **then**
23: $N(h) \leftarrow 0$
24: $V(h) \leftarrow 0$
25: **end if**
26: $\{Q_{ha_1}, \ldots, Q_{ha_K}\} \leftarrow$ ESTIMATEQ(h, θ, d, w, c)
27: $a^* \leftarrow$ GREEDYACTION(h)
28: $N(h) \leftarrow N(h) + 1$
29: $V(h) \leftarrow V(h) + \frac{wQ_{ha^*} - V(h)}{N(h)}$
30: $N_{\text{avg}}(h, a^*) \leftarrow N_{\text{avg}}(h, a^*) + 1$
31: **return** Q_{ha^*}
32: **end function**

b_k at the root of the tree at each iteration can accurately estimate the optimal value function $Q^*_{b_k}(h, a)$. These techniques suggest a naïve approach to solving for the optimal policy: at each iteration of the GWFP process, one could compute $\text{BR}_\epsilon(b_k)$ by running a tree search algorithm on b_k, and compute $\text{BR}_\epsilon(\pi_k)$ by rolling out policy π_k on each model θ_i to get $\hat{V}_{\pi_k}(i)$, and then solve the linear program (Equation 8). This is clearly impractical, as each iteration requires solving a new BAMDP. Furthermore, in order for the GWFP process to converge, the suboptimality of the best responses must go to zero as $k \to \infty$. Thus, each iteration of this naïve implementation would require a growing number of samples, increasing the computational challenges with this approach. Critically, we are able to leverage the structure of the GWFP process to obtain an algorithm that converges to the same result, iterating between performing FP iterations and building the tree. This approach requires growing only one tree, which results in substantial efficiency improvement.

4.4. RAMCP Outline

In the following subsection, we outline the RAMCP algorithm, and draw connections to the naïve implementation of GWFP described in the previous subsection. Algorithm 1 details the overall procedure. To compute a risk-sensitive plan for belief b_{prior} from state s_0, the agent calls the SEARCH function. For each model θ_i, the algorithm computes a weighting w_i as a function of the current adversarial belief. This weight is applied to the tree update such that the estimators are consistent with the true values had θ_i been drawn from the adversarial belief, rather than looped over deterministically (See lines 29 and 49). The proof of the consistency of this weighted update rule is similar to that of importance sampling, and is provided in the supplementary materials.

The call to ESTIMATEV (Algorithm 2) simulates all possible H-length action sequences under θ_i, sampling state transitions c times, and updating a tree of value function estimates, $\hat{V}(h)$ and $\hat{Q}(h, a)$. This tree of N, V, and Q values is a global variable that is incrementally updated over all iterations of the algorithm. ESTIMATEV returns the total reward for the trajectory that took actions according to $\pi^* = \arg\max_a \hat{Q}(h, a)$. The algorithm maintains a running average of these returns for each model θ_i in $\hat{V}_\pi(i)$. Finally, the procedure updates the adversary's best response according to the LP (8), and maintains a running average of these best responses in b_{adv}.

Notice that while iteration k of the naïve approach required building a tree where θ was drawn from b_k and estimating V_{π_k}, iteration k of RAMCP builds the tree as if θ was drawn from $\text{BR}_\epsilon(\pi_k)$, and obtains samples of $V_{\text{BR}_\epsilon(b_k)}$. By sharing information across iterations, RAMCP keeps running averages of these samples in $Q(h, a)$ and the vector $\hat{V}_\pi$ respectively. Since π_k and b_k in the GWFP process are themselves running averages over previous best-responses, RAMCP is able to obtain consistent estimates of $Q^*_{b_k}(h, a)$ and V_{π_k} with only a finite number of simulations $(M|\mathcal{A}|^H)$ at each iteration of fictitious play. In stark contrast to the naïve approach, RAMCP is able to carry out the GWFP process by only adding a small linear programming optimization step to a standard Monte-Carlo tree search procedure.

Note that π_k, the policy in GWFP that converges to the Nash equilibrium, is never explicitly computed. As this policy represents the average of all previous best responses, we simply log the number of times action a coincided with the best response policy over the learning procedure: $N_{\text{avg}}(h, a) = \sum_{j=1}^{k} \mathbb{1}\{a = \arg\max_a Q(h, a)\}$. Thus, π_k

Algorithm 3 EstimateQ

Require: Hyperparameter c
function ESTIMATEQ(h, θ, d, w, c)
  if $N(h,a) = 0$ **then**
    $N(h,a) \leftarrow 0$
    $N_{\text{avg}}(h,a) \leftarrow 0$
    $Q(h) \leftarrow 0$
  end if
  for all $a \in A$ **do**
    $Q_{ha} \leftarrow 0$
    for $j = 1$ **to** c **do**
      $s \leftarrow$ end of h
      $s' \sim T_\theta(\cdot \mid s, a)$
      $r \leftarrow R(s, a, s')$
      $V_{has'} \leftarrow$ ESTIMATEV(has', θ, $d-1$, w, c)
      $Q_{ha} \leftarrow \frac{1}{c}\left(r + \gamma V_{has'}\right)$
    end for
    $N(h,a) \leftarrow N(h,a) + 1$
    $Q(h,a) \leftarrow Q(h,a) + \frac{wQ_{ha} - Q(h,a)}{N(h,a)}$
  end for
  return $\{Q_{ha_1}, \ldots, Q_{ha_K}\}$
end function

can be computed as the stochastic policy where the probability of choosing action a at history h is proportional to $N_{\text{avg}}(h,a)$. Procedure AVGACTION(h) samples actions in this fashion. Thus, RAMCP implicitly carries out the GWFP process, and therefore converges to a Nash equilibrium. More formally, we have the following result, for which the proof is provided in the supplementary material:

Theorem 2 (Convergence of RAMCP)**.** *Let π_k denote the output of RAMCP (Algorithm 1) after k iterations of the outer loop. Then,* $\lim_{k\to\infty} \pi_k \in \Pi^*$.

5. Experiments

We present experimental results for a challenging n-pull multi-armed bandit problem, as well as for a patient treatment scenario. The bandit problem is designed to show the fundamental features of the RAMCP algorithm. In our experiments, we use the CVaR risk metric, which at a certain α-quantile corresponds to the expectation over the α fraction worst-case outcomes. For $\alpha = 1$, this corresponds to the expectation, and in the limit as $\alpha \to 0$, this corresponds to the worst-case metric. For this risk metric, the risk polytope $\mathcal{B}$ may be stated in closed form (Majumdar & Pavone, 2017).

5.1. Multi-armed Bandit

We consider a multi-armed bandit scenario to illustrate several properties of the RAMCP algorithm. Additionally, because we can compute true risk-sensitive optimal policies, we can examine the convergence rate in practice. In this problem there are four possible actions, $\{a_1, a_2, a_3, a_4\}$. There are two possible models, each with different transition probabilities, and these transition probabilities are known in the planning problem. Section C in the supplementary material shows the different reward functions for each model, as well as a description of the relelvent features of the environment. The prior belief is $(0.6, 0.4)$ for θ_1 and θ_2, respectively. In each cell in the table, the left number is the probability of obtaining the reward given in the column header for model θ_1, and the right number is the same for θ_2. An episode consists of two pulls (or two actions) in the environment.

Figure 1 shows the performance of the RAMCP algorithm under varying values of α. The top row shows the adversarially perturbed belief over iterations of the outer loop of RAMCP. The blue points show the solution to Equation 8. These may switch rapidly, as the value of a policy on one model increasing above the value for other models will result in the adversary placing as little probability mass on this high value model as possible. This can be seen for $\alpha = 0.375$. While the best response belief updates change rapidly, the running average adversarially perturbed belief converges.

In the second row, the value associated with the policy π, operating in an environment with each model is plotted (in orange and blue), as well as the expected value with respect to the prior belief. In the risk neutral case, the objective is to maximize this prior-weighted reward. In the risk-sensitive case, the agent will increasingly value increasing the performance in the worst-case model versus maximizing the prior-weighted reward. This can be seen in the figure. As the CVaR quantile α decreases, the expected reward with respect to the prior belief over models decreases, but the performance for the worst-case model improves. This illustrates the robustness of the RAMCP algorithm, as well as the tunability. As α approaches zero, the values associated with each model approach equal. Intuitively, if the value of the policy π on one of the models (say for example, θ_1) was larger than for other models, the adversary could perturb the belief such that the expected value of actions which yield higher reward on θ_2 than θ_1 are favorable. This then decreases the value of policy π operating in an environment with dynamics corresponding to θ_1. The reason this is not observed for all values of α is that the set of allowable perturbations by the agent is a polytope determined by α, and the perturbed belief is on the boundary.

The value for each model converges much more quickly than the average belief. For all values of α, a good value estimate is obtained after approximately 250 iterations. The case where $\alpha = 1$ is exactly the BAMCP algorithm (Guez et al., 2013) (except with uniform sampling as opposed to

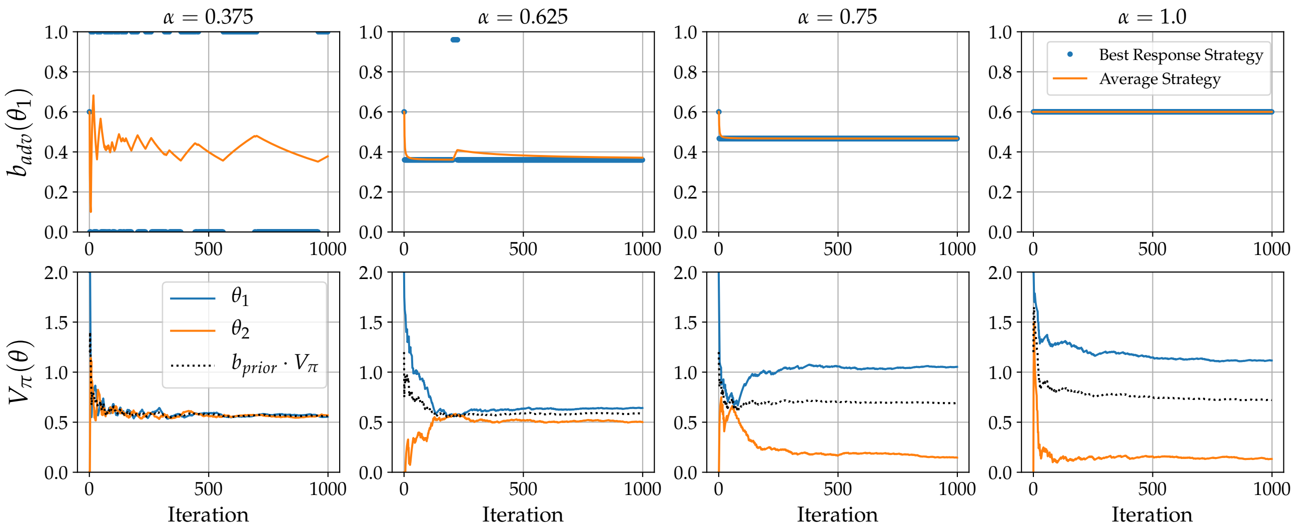

Figure 1. Convergence of the RAMCP algorithm for different CVaR quantiles (α). The upper row shows best response belief perturbations (blue) and the running average adversarial belief perturbation (orange) for various CVaR quantiles, α. The lower row shows the value function for the policy generated by RAMCP operating in an environment with dynamics parameterized by θ_1, and by θ_2 (blue and orange respectively). The dotted line denotes the expected value of the policy under the prior belief over models, which corresponds to the optimal value function in the risk-neutral case.

UCT). The value convergence rate in this case is roughly equal to the risk-sensitive case, and in some cases, the values for a risk-sensitive policy appear to approximately converge in fewer samples than the BAMCP algorithm. This suggests that, assuming the cost of solving the LP is small (there are comparatively few models), the added computational cost of RAMCP compared to BAMCP is negligible.

Figure 2 shows 95% confidence intervals for RAMCP for varying CVaR risk levels. Note that as expected, lower values of α result in policies with substantially lower variance over realized cost. However, this comes at the cost of a slight degredation in mean accrued reward. The non Bayes-adaptive policy (which is Markovian) is also plotted. Because it does not incorporate the information gained from previous transitions, it does not actively disambiguate between models and thus results in low mean reward.

5.2. Patient Treatment

The problem of developing patient treatment plans is one where being robust yet adaptive to model uncertainty is critical. Parameters governing the response of a patient to various treatment strategies may vary from patient to patient, so exploratory actions for model identification are often required. There are many such problems in medicine that have been formulated under the BAMDP or POMDP framework, including choosing drug infusion regimens (Hu et al., 1996), or HIV treatment plans, (Attarian & Tran, 2017). With human lives at stake, being robust towards this model uncertainty is also critical, and thus RAMCP offers an attractive solution approach. We demonstrate the effectiveness of RAMCP in such a domain by developing a simplified problem which captures the complexity of such tasks.

We consider a model where the state is the patient's health: $s \in \mathcal{S} = \{0, 1, \ldots, 6\}$, where $s = 0$ corresponds to death, and $s = 6$ corresponds to a full health level. The patient starts at $s = 2$. The action space is $\mathcal{A} = \{1, 2, 3, 4\}$, corresponding to different treatment options. We consider a uniform prior belief over four possible response profiles θ to the treatment. Each response profile assigns a probability mass to a relative change in patient health. Under the different response profiles, the same action may have a positive or negative effect on health, so rapid model identification is important. The rewards monotonically increase from -2 to $+1$ with the patient health. The exact transition probabilities and reward model is given in the supplementary materials.

Figure 3 shows the performance of the RAMCP algorithm on this problem, run for 3750 iterations with a search horizon $H = 4$, 50 times for each quantile. Notice that the risk-neutral algorithm yields a solution which the 95% confidence bound includes scenarios with patient death. RAMCP allows practitioners to adjust their degree of risk sensitivity and instead converge to a class of policies for which the 95% confidence bound is entirely positive. Again, notice the variance decrease as α decreases. In this case however, there is no noticeable decrease in mean performance.

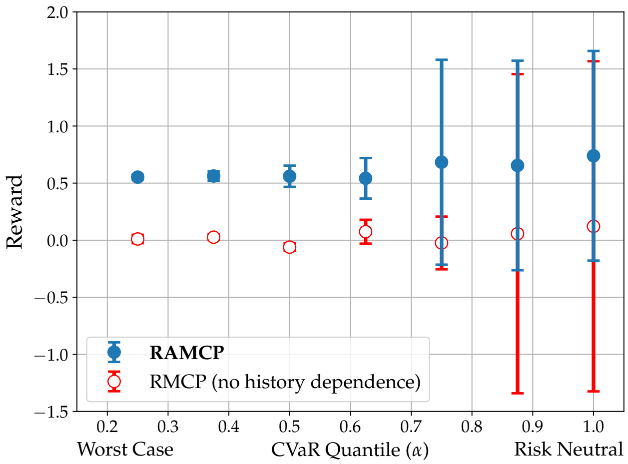

Figure 2. Performance of the RAMCP algorithm on a bandit problem for varying CVaR quantiles (α). For each quantile, 500 trials were performed. In this plot, the error bars denote the 95% confidence bound corresponding to variation in performance due to uncertainty over the underlying model. Applying this same risk-sensitive optimization method on a state dependent policy, yields the RMCP policy, which performs consistently worse than the history dependent RAMCP policy, highlighting the importance of planning with adaptation in mind.

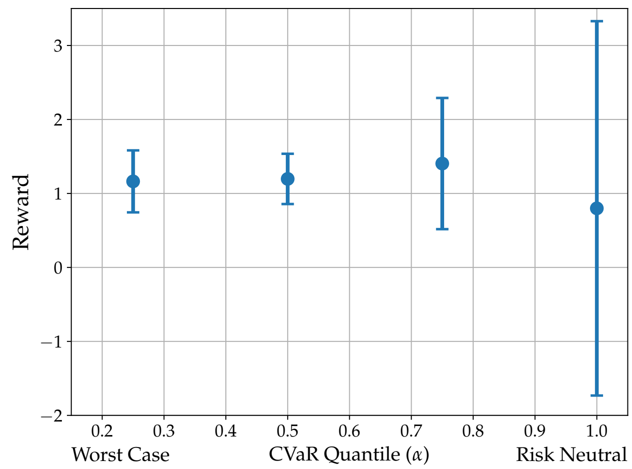

Figure 3. Performance of the RAMCP algorithm on a patient treatment problem for varying CVaR quantiles (α). For each quantile, 500 trials were performed. In this plot, the error bars denote the 95% confidence bound corresponding to variation in performance due to uncertainty over the underlying model.

6. Related Work

[JH: expand on Bayes adaptive MDP methods; can move content from below]

[JH: expand on other risk methods]

[JH: edit this] The notion of robust policy selection subject to model uncertainty in Markov Decision Processes is captured by the *Robust MDP* framework, and there is a substantial body of literature on this problem. Typically, these problems are concerned with uncertainty in the transition probabilities (Nilim & El Ghaoui, 2005). More specifically, work in this area aims to learn policies that maximize expected reward subject to the worst-case disturbances in some uncertainty set, with no consideration for the associated probabilities of these disturbances. This framework typically leads to extremely conservative policies. Furthermore, computing robust policies for general uncertainty sets is NP-hard (Bagnell et al., 2001). Moreover even for rectangular uncertainty sets, computing the policy for the static case (in which the perturbed model is the same every time the state-action pair (s, a) is visited) is NP-hard (Iyengar, 2005).

While Robust MDPs consider the worst-case expected reward, the *Risk-Sensitive MDP* framework instead considers replacing the expected reward of standard MDPs with a risk measure (Howard & Matheson, 1972). This allows a naturally tunable notion of conservatism. However, both the risk-sensitive and robust MDP frameworks do not consider that the uncertainty over dynamics models may be reduced as the agent interacts with the environment, still leading to unnecessarily conservative policies that do not consider the value of information gathering actions.

The field of adaptive control generally aims to estimate parameters in a model while safely controlling the system (Aström & Wittenmark, 2013). However, guarantees are typically only available for specific types of systems (e.g. linear with Gaussian noise) and estimation methods (e.g. linear parametric models), and robustness is typically in the worst-case sense (Ioannou & Sun, 1996). Yu et al. (2017) develop a model in which they train a policy which is dependent on dynamics parameters, as well as a system identification model that maps from observed transitions in the environment to dynamics parameters, but no convergence guarantees are established and errors could compound. That work, as well as traditional adaptive control approaches typically do not consider the value of information in their action selection. Blackmore & Williams (2006) developed an algorithm for model discrimination with constraints on task satisfaction for aircraft, but they do not simultaneously consider reward maximization. This exploration-exploitation trade-off was addressed in (Bai et al., 2013), in which the authors model the problem as a POMDP and generates offline plans, but the work does not consider robustness.

7. Discussion and Conclusions

Planning with uncertainty over models is a pervasive problem in autonomous decision making, from human robot interaction, to patient treatment, to allocation problems in finance and operations research. It is desirable to generate policies in these problems that take into account the value of information, and so may plan to gather knowledge of the underlying model. However, it is also desirable to be robust to model uncertainty, as beliefs over models are approximate and typically inaccurate. Worst-case robustness results in over-conservatism, so we leverage the concept of risk-sensitivity instead. We have developed the RAMCP algorithm, which has tunable risk-sensitivity, and generates asymptotically optimal Bayes-adaptive policies. This algorithm relies on building look-ahead trees in response to adversarial belief perturbations, and critically, is only marginally more computationally expensive than risk-neutral Bayes-adaptive planning. We have also proved that the RAMCP algorithm converges asymptotically, whereas naively approaches of belief perturbation may result in oscillation or instability of the generated policy. This algorithm is general to Bayes-Adaptive Markov Decision Processes, and therefore is applicable to a wide variety of domains in which robustness to model belief is desirable.

Acknowledgements

Apoorva Sharma was supported in part by the Stanford School of Engineering Fellowship. James Harrison was supported in part by the Stanford Graduate Fellowship and the National Sciences and Engineering Research Council (NSERC). The authors were partially supported by the Office of Naval Research, Science of Autonomy Program, under Contract N00014-15-1-2673

References

Artzner, Philippe, Delbaen, Freddy, Eber, Jean-Marc, and Heath, David. Coherent measures of risk. *Mathematical finance*, 9(3):203–228, 1999.

Asmuth, John and Littman, Michael L. Learning is planning: near bayes-optimal reinforcement learning via monte-carlo tree search. *arXiv preprint arXiv:1202.3699*, 2012.

Aström, Karl J and Wittenmark, Björn. *Adaptive control*. Courier Corporation, 2013.

Attarian, Adam and Tran, Hien. An optimal control approach to structured treatment interruptions for hiv patients: a personalized medicine perspective. *Applied Mathematics*, 8(7):934–955, 2017.

Bagnell, J Andrew, Ng, Andrew Y, and Schneider, Jeff G. Solving uncertain markov decision processes. 2001.

Bai, Haoyu, Hsu, David, and Lee, Wee Sun. Planning how to learn. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 2853–2859. IEEE, 2013.

Blackmore, Lars and Williams, Brian. Finite horizon control design for optimal discrimination between several models. In *Decision and Control, 2006 45th IEEE Conference on*, pp. 1147–1152. IEEE, 2006.

Brown, George W. Some notes on computation of games solutions. Technical report, RAND CORP SANTA MONICA CA, 1949.

Chen, Katherine and Bowling, Michael. Tractable objectives for robust policy optimization. In *Advances in Neural Information Processing Systems*, pp. 2069–2077, 2012.

Chen, Min, Frazzoli, Emilio, Hsu, David, and Lee, Wee Sun. Pomdp-lite for robust robot planning under uncertainty. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pp. 5427–5433. IEEE, 2016.

Chow, Yin-Lam and Pavone, Marco. A framework for time-consistent, risk-averse model predictive control: Theory and algorithms. In *American Control Conference (ACC), 2014*, pp. 4204–4211. IEEE, 2014.

Chow, Yinlam, Tamar, Aviv, Mannor, Shie, and Pavone, Marco. Risk-sensitive and robust decision-making: a cvar optimization approach. In *Advances in Neural Information Processing Systems*, pp. 1522–1530, 2015.

Doucet, Arnaud, De Freitas, Nando, and Gordon, Neil. An introduction to sequential monte carlo methods. In *Sequential Monte Carlo methods in practice*, pp. 3–14. Springer, 2001.

Duff, Michael O'Gordon and Barto, Andrew. *Optimal Learning: Computational procedures for Bayes-adaptive Markov decision processes*. PhD thesis, University of Massachusetts at Amherst, 2002.

Ghavamzadeh, Mohammad, Mannor, Shie, Pineau, Joelle, Tamar, Aviv, et al. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.

Guez, Arthur, Silver, David, and Dayan, Peter. Scalable and efficient bayes-adaptive reinforcement learning based on monte-carlo tree search. *Journal of Artificial Intelligence Research*, 48:841–883, 2013.

Guez, Arthur, Heess, Nicolas, Silver, David, and Dayan, Peter. Bayes-adaptive simulation-based search with value function approximation. In *Advances in Neural Information Processing Systems*, pp. 451–459, 2014.

Heinrich, Johannes, Lanctot, Marc, and Silver, David. Fictitious self-play in extensive-form games. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 805–813, 2015.

Howard, Ronald A and Matheson, James E. Risk-sensitive markov decision processes. *Management science*, 18(7): 356–369, 1972.

Hu, Chuanpu, Lovejoy, William S, and Shafer, Steven L. Comparison of some suboptimal control policies in medical drug therapy. *Operations Research*, 44(5):696–709, 1996.

Ioannou, Petros A and Sun, Jing. *Robust adaptive control*, volume 1. PTR Prentice-Hall Upper Saddle River, NJ, 1996.

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

Kearns, Michael, Mansour, Yishay, and Ng, Andrew Y. A sparse sampling algorithm for near-optimal planning in large markov decision processes. *Machine learning*, 49 (2-3):193–208, 2002.

Leslie, David S and Collins, Edmund J. Generalised weakened fictitious play. *Games and Economic Behavior*, 56 (2):285–298, 2006.

Majumdar, Anirudha and Pavone, Marco. How should a robot assess risk? towards an axiomatic theory of risk in robotics. *arXiv preprint arXiv:1710.11040*, 2017.

Majumdar, Anirudha, Singh, Sumeet, Mandlekar, Ajay, and Pavone, Marco. Risk-sensitive inverse reinforcement learning via coherent risk models. In *Robotics: Science and Systems*, 2017.

Martin, James John. *Bayesian decision problems and Markov chains*. Wiley, 1967.

Nilim, Arnab and El Ghaoui, Laurent. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

Pinto, Lerrel, Davidson, James, Sukthankar, Rahul, and Gupta, Abhinav. Robust adversarial reinforcement learning. 2017. URL https://arxiv.org/abs/1703.02702.

Schmerling, Edward, Leung, Karen, Vollprecht, Wolf, and Pavone, Marco. Multimodal probabilistic model-based planning for human-robot interaction. *arXiv preprint arXiv:1710.09483*, 2017.

Silver, David and Veness, Joel. Monte-carlo planning in large pomdps. In *Advances in neural information processing systems*, pp. 2164–2172, 2010.

Tamar, Aviv, Chow, Yinlam, Ghavamzadeh, Mohammad, and Mannor, Shie. Sequential decision making with coherent risk. *IEEE Transactions on Automatic Control*, 62 (7):3323–3338, 2017.

Wang, Tao, Lizotte, Daniel, Bowling, Michael, and Schuurmans, Dale. Bayesian sparse sampling for on-line reward optimization. In *Proceedings of the 22nd international conference on Machine learning*, pp. 956–963. ACM, 2005.

Yu, Wenhao, Liu, C Karen, and Turk, Greg. Preparing for the unknown: Learning a universal policy with online system identification. *arXiv preprint arXiv:1702.02453*, 2017.

A. Proof of Theorem 2

In this section we prove Theorem 2, and provide necessary background material. We begin by introducing Generalized Weakened Fictitious Play (GWFP) (Leslie & Collins, 2006), and restating the core convergence results from that work.

Generally, we consider a repeated N-player normal-form game. We will write the pure strategy set of player i as A^i, and the mixed strategy set as Δ^i, where a mixed strategy is a distribution over pure strategies. Let $r^i : \times_{i=1}^N \Delta^i \to \mathbb{R}$ denote the bounded reward function of player i (where $\times_{i=1}^N$ denotes the Cartesian product). Then, letting the π^{-i} denote a set of mixed strategies for all players but player i, let $r^i(\pi^i, \pi^{i-1})$ denote the expected reward to player i selecting strategy π^i, if all other players select π^{i-1}. We will write the best response of player i to π^{i-1} as

$$BR^i(\pi^{i-1}) = \arg\max_{\pi^i \in \Delta^i} r^i(\pi^i, \pi^{i-1}),$$

and thus, we can write the collection of best responses as

$$BR(\pi) = \times_{i=1}^N BR^i(\pi^{i-1}).$$

Finally, in a similar fashion, we will define ϵ-best response of player i to be the set

$$BR^i_\epsilon(\pi^{-i}) = \\ \left\{\pi^i \in \Delta^i : r^i(\pi^i, \pi^{-i}) \geq r^i(BR^i(\pi^{-i}), \pi^{-i}) - \epsilon\right\}.$$

The set of ϵ-best strategies is defined in the same way as above, as is written $BR_\epsilon(\pi)$. We may now formally define the GWFP update process.

Definition 2 (Generalized Weakened Fictitious Play (GWFP) Process (Leslie & Collins, 2006))**.** *A GWFP process is any process $\{\sigma_n\}_{m\geq 0}$, with $\sigma_n \in \times_{i=1}^N \Delta^i$, such that*

$$\sigma_{n+1} \in (1-\alpha_{n+1})\sigma_n + \alpha_{n+1}(BR_{\epsilon_n}(\sigma_n) + M_{n+1}) \quad (10)$$

with $\alpha_n \to 0$ and $\epsilon_n \to 0$ as $n \to \infty$,

$$\sum_{n\geq 1} \alpha_n = \infty,$$

and $\{M_n\}_{n\geq 1}$ is a sequence of perturbations such that, for any $T > 0$,

$$\lim_{n\to\infty} \sup_k \left\{ \left\| \sum_{i=n}^{k-1} \alpha_{i+1} M_{i+1} \right\| : \sum_{i=n}^{k-1} \alpha_{i+1} \leq T \right\} = 0.$$

Equation 10 is written as an inclusion, and thus can be thought of as defining a set of possible next strategy profiles. In practice, we only compute one strategy profile which lies in the set $BR_\epsilon(\pi)$, and assign σ_{n+1} to the corresponding mixed profile. Thus, the reader should think of this equation as defining an iterative updating of the strategy profiles.

Critically, this definition allows us to establish guarantees on convergence in certain cases.

Lemma 3 (Convergence of GWFP Processes)**.** *Any GWFP process will converge to the set of Nash equilibria in two-player zero-sum games, in potential games, and in generic $2 \times m$ games.*

Proof. Restatement of Corollary 5 in (Leslie & Collins, 2006). □

We will now prove several results on the convergence of the look-ahead tree estimates in RAMCP, which are used to prove Theorem 2.

Lemma 4 (Convergence of Tree Q Estimates)**.** *Let $Q^*_{b_k}(h,a)$ be the Q value associated corresponding the Bayes-optimal policy under belief b_k. Let $\hat{Q}_k(h,a)$ be the estimate that is obtained by performing k tree updates according to line 49 in Algorithm **??**, sampling θ from b^*_k at each iteration. Then, for all h shorter than the search horizon H, and for all actions a, $\mathbb{E}[\hat{Q}_k(h,a)] = Q^*_{b_k}(h,a)$, and $\hat{Q}_k(h,a)$ converges to $Q^*_{b_k}(h,a)$ in probability.*

Proof. We prove this lemma by first showing that the theorem holds for leaf nodes of the look-ahead tree, where histories h are of length H. At these nodes,

$$Q^*_{b_k}(h,a) = \mathbb{E}[\mathbb{E}[R(s,a,s') \mid s' \sim p(s'|s,a,\theta)] \mid \theta \sim b_k(\theta)].$$

Let $\bar{R}(h,a,\theta) = \mathbb{E}[R(s,a,s') \mid s' \sim p(s'|s,a,\theta)]$.

The estimates in the algorithm are calculated as

$$N_k(h,a)\hat{Q}_k(h,a) = \sum_{j=1}^k \mathbb{1}_{h_j=h} R_j \quad (11)$$

$$N_k(h,a) = \sum_{j=1}^k \mathbb{1}_{h_j=h} \quad (12)$$

where R_j is the reward observed in the trajectory sampled at iteration j. Taking the expectation of Equation 11,

$$\mathbb{E}[N_k(h,a)\hat{Q}_k(h,a)] = \sum_{j=1}^{k} \mathbb{E}[\mathbb{1}_{h_j=h} R_j \mid \theta \sim b_j^*(\theta)] \tag{13}$$

$$= \sum_{j=1}^{k}\sum_{i=1}^{M} b_j^*(\theta_i)\mathbb{E}[\mathbb{1}_{h_j=h} R_j \mid \theta_j = \theta_i] \tag{14}$$

$$= \sum_{i=1}^{M}\sum_{j=1}^{k} b_j^*(\theta_i) p(h \mid \theta_i)\bar{R}(h,a,\theta) \tag{15}$$

$$= \sum_{i=1}^{M} p(h \mid \theta_i)\bar{R}(h,a,\theta)\sum_{j=1}^{k} b_j^*(\theta_i) \tag{16}$$

$$= \sum_{i=1}^{M} p(h \mid \theta_i)\bar{R}(h,a,\theta) k b_k(\theta_i) \tag{17}$$

$$= \sum_{i=1}^{M} p(h \mid \theta_i)\bar{R}(h,a,\theta)\sum_{j=1}^{k} b_k(\theta_i) \tag{18}$$

$$= \sum_{j=1}^{k} \mathbb{E}[\mathbb{1}_{h_j=h} R_j \mid \theta \sim b_k(\theta)] \tag{19}$$

$$= \mathbb{E}[N_k^*(h,a)\hat{Q}_k^*(h,a)] \tag{20}$$

where $\hat{Q}_k^*(h,a)$, $N_k^*(h,a)$ are the quantities that would be attained had $\theta \sim b_k$ at every iteration. Equation 17 was obtained by substituting the definition of b_k in the smoothed belief and policy updating. A very similar analysis shows that $\mathbb{E}[N_k(h,a)] = \mathbb{E}[N_k^*(h,a)]$. Note that if action selection is uniform during the tree building, then the number of realizations of $N_k(h,a)$ should be independent of the value $\hat{Q}_k(h,a)$, meaning $\mathbb{E}[N_k(h,a)\hat{Q}_k(h,a)] = \mathbb{E}[N_k(h,a)]\mathbb{E}[\hat{Q}_k(h,a)]$. Together, these facts imply that $\mathbb{E}[\hat{Q}_k(h,a)] = \mathbb{E}[\hat{Q}_k^*(h,a)]$. Guez et al. (2013) show that the estimates obtained from tree search when sampling θ at the root of the tree from a fixed belief b_k converge to the Bayes-optimal value, $Q_{b_k}^*(h,a)$. Thus, we have shown that for the leaf nodes in our tree, for all $\epsilon > 0$

$$\lim_{k\to\infty} P(|\hat{Q}_{k-1}(h,a) - Q_{b_{k-1}}^*(h,a)| > \epsilon) = 0 \tag{21}$$

Kearns et al. (2002) give an inductive argument showing that convergence at the leaves of such a search tree leads to convergence at nodes higher up the tree. □

For the purposes of the following proof, we assume without loss of generality that the total rewards are bounded within $[0,1]$. Furthermore, throughout this section we assume the b_k and π_k are the belief and agent policy that follow the iterative averaging process that is analogous to the GWFP process defined above. For ease of notation, we refer to the ϵ-best responses of the adversary as $b_k^* \in BR_\epsilon(\pi_{k-1})$, and those of the agent as $\pi_k^* \in BR_\epsilon(\pi_{k-1})$.

Lemma 5 (Convergence of Model Value Estimates)**.** *Let $V_{\pi_k}(\theta_i)$ correspond to the expected reward of policy π_k on model θ_i. Let $\hat{V}_k(\theta_i)$ be the estimate calculated by the iterative update in line 11 of Algorithm 1. $\mathbb{E}[\hat{V}_k(\theta_i)] = V_{\pi_k}(\theta_i)$, and the error between the two converges in probability as*

$$P(|\hat{V}_k(\theta_i) - V_{\pi_k}(\theta_i)| \geq \epsilon) \leq 2\exp(-2k^2\epsilon^2) \tag{22}$$

Proof. To show that the iterative estimates $\hat{V}_k(\theta_i)$ serve as an unbiased estimator for $V_{\pi_k}(\theta_i)$, we leverage the fact that π_k is a moving average

$$\pi_k = (1 - \frac{1}{k})\pi_{k-1} + \frac{1}{k}\pi_k^*$$
$$= \frac{1}{k}\sum_{i=1}^{k}\pi_i^*.$$

Note that this average over policies represents to a mixing of strategies, i.e. π_k is the mixed strategy that randomly picks between the k past best-response policies, and plays that policy. This means that the value of this mixed strategy will be itself a moving average:

$$V_{\pi_k}(\theta_i) = (1 - \frac{1}{k})V_{\pi_{k-1}}(\theta_i) + \frac{1}{k}V_{\pi_k^*}(\theta_i)$$

Note that π_k^* is the ϵ-best response to belief b_{k-1}. The approximate Q value $\hat{Q}_{k-1}(h,a)$ is an estimator of the true optimal value function for b_{k-1}, $Q_{b_{k-1}}^*(h,a)$, for which

$$\lim_{k\to\infty}\left(\hat{Q}_{k-1}(h,a) - Q_{b_{k-1}}^*(h,a)\right) = 0 \tag{23}$$

by Lemma 4. This estimator defines the ϵ-best response $\pi_k^*(h) = \arg\max_a \hat{Q}_{k-1}(h,a)$. Given this policy, we can perform rollouts under model θ_i obtain a Monte Carlo estimate of $V_{\pi_k^*}(\theta)$, which we denote $\hat{V}_{\pi_k^*}(\theta_i)$ (see line 10 of Algorithm 1). Note that $\hat{V}_{\pi_k^*}(\theta_i)$ is random variable with expectation is $V_{\pi_k^*}(\theta)$, that takes on values between $[0,1]$, given our assumption regarding the range of rewards. If we compute $\hat{V}_{\pi_k}(\theta_i)$ as the empirical mean of these estimates, as in line 11 of Algorithm 1, then the error in the estimate will converge in probability by Hoeffding's inequality, resulting in Inequality 22. □

Now, based on the above, we may prove Theorem 2.

Proof of Theorem 2. We will begin by rewriting Equation 10 in the specific notation of Algorithm 1. Let b_k denote the average belief over model parameters after k iterations. Therefore, Equation 10 is equivalent to

$$b_{k+1} = (1 - \alpha_{k+1})b_k + \alpha_{k+1}BR_\epsilon(\pi_k) \tag{24}$$

$$\pi_{k+1} = (1 - \alpha_{k+1})\pi_k + \alpha_{k+1}BR_\epsilon(b_k), \tag{25}$$

where we set $M_k = 0$ for all k. We will show that the belief and policy updates that are being performed in Algorithm 1 satisfy the above. We will first look at the policy update. By Lemma 4, the approximate Q values computed from the tree converge in probability to the optimal Q function for the averaged set of beliefs. This implies the policy is an ϵ-best response, with $\epsilon \to 0$ as $k \to \infty$. Similarly, for the belief update, the Value estimate converges in probability to the optimal value for each model. Thus the computed solution to Equation 2 converges to the best response as $k \to \infty$. Therefore both the belief update and the policy update satisfy the definition of a GWFP.

By Lemma 3, any GWFP process will converge to the set of Nash Equilibria in zero-sum, two-player games. Note that since the infinite action set for the adversary, $\mathcal{B}$, is convex, any Nash equilibrium will correspond to a solution of Equation 5 by the Minimax theorem. This implies that as $k \to \infty$, the computed belief and policy converge to the optimal Nash equilibrium, meaning $\lim_{k\to\infty} \pi_k \in \Pi^*$. □

B. Weighted Tree Updates

When we sample θ at the root of the tree, and then follow any policy up to history h, the distribution of samples of θ at a node h will be distributed according to $p(\theta_i|h)$. We refer to Lemma 1 in Guez et al. (2013) for a proof. We require that θ_i be sampled uniformly to get consistent estimates of $V_{\theta_i}(\pi)$ for every θ. However, we would like to update the tree's estimates to reflect values corresponding to simulation with $\theta_i \sim b_j(\theta)$, the adversarially chosen distribution.

Let $Q_q(h, a)$ represent a sample of $Q(h, a)$ obtained at node (h, a) where θ was sampled from from $q(\theta)$ at the root. The standard Monte Carlo updates for the estimator are:

$$\hat{N}_q^{(k+1)}(h,a) = \hat{N}_q^{(k)}(h,a) + 1$$
$$\hat{Q}_q^{(k+1)}(h,a) = \hat{Q}_q^{(k)}(h,a) + \frac{(Q_q^k(h,a) - \hat{Q}_q^{(k)}(h,a))}{\hat{N}_q^{(k)}(h,a)} \tag{26}$$

with

$$\hat{N}_q^{(0)}(h,a) = 0 \tag{27}$$
$$\hat{Q}_q^{(0)}(h,a) = 0. \tag{28}$$

Define the weighted estimators $\hat{Q}_{q,w}^{(k)}(h,a)$ with the update equation as

$$\hat{N}_{q,w}^{(k+1)}(h,a) = \hat{N}_{q,w}^{(k)}(h,a) + 1$$
$$\hat{Q}_{q,w}^{(k+1)}(h,a) = \hat{Q}_{q,w}^{(k)}(h,a) + \frac{(w(\theta_k)Q_q^k(h,a) - \hat{Q}_q^{(k)}(h,a))}{\hat{N}_q^{(k)}(h,a)} \tag{29}$$

with

$$\hat{N}_{q,w}^{(0)}(h,a) = 0 \tag{30}$$
$$\hat{Q}_{q,w}^{(0)}(h,a) = 0. \tag{31}$$

With the correct choice of weighting, we can sample θ from one distribution q, while obtaining estimates of the Q values as if θ was sampled from a different distribution p.

Theorem 6 (Correctness of Weighted Tree Updates)**.** *If $w = p/q$, then $\lim_{k\to\infty} \hat{Q}_p^{(k)}(h,a) = \lim_{k\to\infty} \hat{Q}_{q,w}^{(k)}(h,a)$, i.e. the estimators converge in expectation.*

Proof. Note that the normal recurrence relation (26) corresponds to the explicit formula

$$\hat{Q}_p^{(k)}(h,a) = \frac{1}{N_p^{(k)}(h,a)} \sum_{i=1}^{N_p^{(k)}(h,a)} Q_p^i(h,a).$$

This is a Monte Carlo estimate of $Q_p^i(h,a)$, and thus will converge to

$$\mathbb{E}[Q_p^i(h,a)] = \int_{\theta_i} \mathbb{E}[Q_p^i(h,a)|\theta_i] p(\theta_i|h) d\theta_i \tag{32}$$
$$= \int_{\theta_i} \frac{\mathbb{E}[Q_p^i(h,a)|\theta_i] p(h|\theta_i) p(\theta_i)}{p(h)} d\theta_i, \tag{33}$$

where Equation 32 follows from the definition of the expectation, and Equation 33 was obatined by application of Bayes' rule.

Similarly, the weighted recurrence corresponds to the explicit formula

$$\hat{Q}_{q,w}^{(k)}(h,a) = \frac{1}{N_{q,w}^{(k)}(h,a)} \sum_{i=1}^{N_{q,w}^{(k)}(h,a)} w(\theta_i) Q_q^i(h,a).$$

This is a Monte Carlo estimate of $w(\theta_i)Q_q^i(h,a)$, which converges to the expected value of $w(\theta_i)Q_q^i(h,a)$:

$$\mathbb{E}[w(\theta_i)Q_q^i(h,a)] = \int_{\theta_i} \mathbb{E}[w(\theta_i)Q_q^i(h,a)|\theta_i] p(\theta_i|h) d\theta_i \tag{34}$$
$$= \int_{\theta_i} \frac{\mathbb{E}[Q_q^i(h,a)|\theta_i] w(\theta_i) p(h|\theta_i) q(\theta_i)}{p(h)} d\theta_i \tag{35}$$
$$= \int_{\theta_i} \frac{\mathbb{E}[Q_q^i(h,a)|\theta_i] p(h|\theta_i) p(\theta_i)}{p(h)} d\theta_i \tag{36}$$
$$= \int_{\theta_i} \frac{\mathbb{E}[Q_p^i(h,a)|\theta_i] p(h|\theta_i) p(\theta_i)}{p(h)} d\theta_i \tag{37}$$
$$= \mathbb{E}[Q_p^i(h,a)]. \tag{38}$$

From Equation 34 to Equation 35, we use the fact that $w(\theta_i)$ is constant within the conditional expectation. Equation 36 is obtained by substituting $w = q/p$. Equation 37 follows because conditioned on θ_i, the estimates of Q_p^i and Q_q^i should have the same distribution, since the only difference between them is in the sampling of θ. Since the two update formulas are Monte Carlo estimates of quantities with the same expectation, their value as $N(h, a) \to \infty$ will be the same. Since the number of times a node is simulated goes to infinity with the number of iterations, these estimators must converge to the same value in the limit as $k \to \infty$ as well. □

C. Bandit Reward Model

Table 1. Bandit model for the n-pull bandit experiment. Each cell lists $p(R_i \mid \theta_1)/p(R_i \mid \theta_2)$, the probabilities of obtaining the reward R_i under each of the two models, where the reward R_i is listed in the column heading. Each column corresponds to a certain reward, each row corresponds to a certain action.

R	-1.0	-0.5	-0.1	0.0	0.5	1.0
a_1	0 / 0	0 / 0	1 / 0	0 / 1	0 / 0	0 / 0
a_2	0 / 0	0 / 1	0 / 0	0 / 0	1 / 0	0 / 0
a_3	.2 / .8	0 / 0	0 / 0	0 / 0	0 / 0	.8 / .2
a_4	.8 / .2	0 / 0	0 / 0	0 / 0	0 / 0	.2 / .8

There are several features to note about this bandit problem. First, the rewards for a_1 and a_2 are not stochastic within each model. The important result of this is that taking one of these actions completely disambiguates between the models. For example, if an agent takes action a_1 and receives a reward of -0.1, the posterior probability of θ_1 is 1, and the probability of θ_2 is 0. This is important, as on an n-pull bandit problem, an agent action optimally in a Bayes-adaptive sense will aim to trade off exploration and exploitation. The difference between the two actions is the magnitudes of the reward for each model. If an agent is acting in a risk-sensitive fashion with respect to models, given for example a prior belief over models of 0.5 for each model, the agent may prefer a_1, which has a lower expected reward but also a lower variance. In the risk-neutral case, a Bayes-adaptive agent will prefer a_2, which has higher expected reward but also higher variance, and lower worst-model performance.

For actions a_3 and a_4, the reward distributions will have the same mean, variance, and worst-model performance for a uniform prior over models. However, given better knowledge of the model, the expected value of one of the two actions increases. Therefore, in the case where the agent takes action a_1 or a_2 to disambiguate between the models (thus setting the posterior probability of one of the models to 1), the expected reward under either a_3 or a_4 will be 0.6. Therefore, in this environment, a Bayes-adaptive optimal agent will disambiguate between models by taking action a_1 or a_2, and then exploit by taking either a_3 or a_4. The specific actions and the associated value of the optimal Bayes-adaptive policy depend on the risk metric chosen.

D. Patient Treatment Experimental Details

The reward model and transition models used in the patient treatment experiment.

Table 2. Reward Model

State	0	1	2	3	4	5	6
Reward	-2	-0.1	0.0	0.2	0.5	0.7	1.0

Table 3. Tranistion Model

$s' - s$	-2	-1	0	1	2
$p(s' - s \vert \theta_1, a_1)$	0.9	0.1	0	0	0
$p(s' - s \vert \theta_1, a_2)$	0	0.9	0.1	0	0
$p(s' - s \vert \theta_1, a_2)$	0	0	0.1	0.9	0
$p(s' - s \vert \theta_1, a_3)$	0	0	0	0.1	0.9
$p(s' - s \vert \theta_2, a_1)$	0	0.5	0.5	0	0
$p(s' - s \vert \theta_2, a_2)$	0.5	0.5	0	0	0
$p(s' - s \vert \theta_2, a_3)$	0	0	0	0.5	0.5
$p(s' - s \vert \theta_2, a_4)$	0	0	0.5	0.5	0
$p(s' - s \vert \theta_3, a_1)$	0	0	0	0.1	0.9
$p(s' - s \vert \theta_3, a_2)$	0	0	0.1	0.9	0
$p(s' - s \vert \theta_3, a_3)$	0	0.9	0.1	0	0
$p(s' - s \vert \theta_3, a_4)$	0.9	0.1	0	0	0
$p(s' - s \vert \theta_4, a_1)$	0	0	0.5	0.5	0
$p(s' - s \vert \theta_4, a_2)$	0	0	0	0.5	0.5
$p(s' - s \vert \theta_4, a_3)$	0.5	0.5	0	0	0
$p(s' - s \vert \theta_4, a_4)$	0	0.5	0.5	0	0