Robust Control of the Multi-armed Bandit Problem

Felipe Caro* Aparupa Das Gupta†
UCLA Anderson School of Management

September 9, 2015
Forthcoming in *Annals of Operations Research*
http://dx.doi.org/10.1007/s10479-015-1965-7

Abstract

We study a robust model of the multi-armed bandit (MAB) problem in which the transition probabilities are ambiguous and belong to subsets of the probability simplex. We first show that for each arm there exists a robust counterpart of the Gittins index that is the solution to a robust optimal stopping-time problem and can be computed effectively with an equivalent restart problem. We then characterize the optimal policy of the robust MAB as a project-by-project retirement policy but we show that arms become dependent so the policy based on the robust Gittins index is not optimal. For a project selection problem, we show that the robust Gittins index policy is near optimal but its implementation requires more computational effort than solving a non-robust MAB problem. Hence, we propose a Lagrangian index policy that requires the same computational effort as evaluating the indices of a non-robust MAB and is within 1% of the optimum in the robust project selection problem.

Keywords: multiarmed bandit; index policies; Bellman equation; robust Markov decision processes; uncertain transition matrix; project selection.

1. Introduction

The classical Multi-armed Bandit (MAB) problem can be readily formulated as a Markov decision process (MDP). A traditional assumption for the MDP formulation is that the state transition probabilities are either known in advance or estimated from data. The optimal policy is computed ignoring any ambiguity in these transition probabilities. In practice, the transition probabilities are based on the judgement of the decision maker or estimated from historical data which inevitably has some associated noise rendering the probabilities ambiguous. When this ambiguity is taken into account in the optimization phase, a robust approach is needed.

In this paper we model the robust MAB (RMAB) problem as a game between the decision maker or controller and an adversary — which we call *nature* — such that the controller seeks to maximize the expected reward by selecting a project to work on, and in response nature chooses the worst possible expected reward by selecting the transition probability from a predefined ambiguity

*Email address: fcaro@anderson.ucla.edu
†Email address: aparupa@ucla.edu

set. Our main contributions are: 1) we show that the RMAB problem is *indexable* in the sense of Whittle [29] and the optimal policy is a *project-by-project retirement* policy; 2) we characterize an index for the RMAB problem, the robust index (RI), as an optimal stopping time that can be computed by solving a restart problem and show that it is suboptimal; 3) we propose the Lagrangian index (LI) policy that is computationally easier to evaluate than RI; 4) we propose a partial robust value iteration approach to approximately evaluate the worst-case expected reward of a policy; and 5) for the sequential project selection (SPS) problem we show that the suboptimality gaps of the LI and RI policies are comparable and near optimal. Overall, our work contributes to the nascent literature on approximate methods for robust MDPs which to date is a relatively unexplored area.

Many authors have addressed the issue of ambiguous transition probabilities of an MDP (see Satia and Lave [25], White and Eldeib [27], Givan et al.[13], Ng et al. [2], Nilim and El Ghaoui [20], Iyengar [15], Shapiro [26]). All these papers consider that the state transition probability lies in a given subset of the probability simplex (i.e. the ambiguity set) and they consider all the possible scenarios for the transition probability matrix within these pre-defined sets and seek a policy for the decision maker that performs best in the worst-case scenario. An approach which gives less conservative robust policies has been proposed by Paschalidis and Kang [22]. Delage and Mannor [9] use chance constraints for the same effect. Wiesemann et al. [30] introduced general class of ambiguity sets in which the transition probability chosen by nature for the same state but different actions can be dependent. All these papers provide general frameworks that are computationally intensive, if not untractable. There is a dearth of approximate methods, even for particular applications, to the point that Iyengar [15] concludes his paper calling for more research in this area. One of the few applications we are aware of is Dimitrov et al. [11] that study a robust MDP formulation for school budget allocation. Their approximate method is based on a Lagrangian decomposition that is similar to ours.

In contrast to the papers mentioned above, the focus of this paper is to study the RMAB problem. Our robust dynamic programming formulation follows Iyengar [15] and Nilim and El Ghaoui [20] so we assume that transitions are ambiguous but within certain sets. There is a separate stream of literature on sequential sampling of bandits in which the expected rewards depend on unknown parameters; see Lai and Robbins [19], Katehakis and Robbins [16], Burnetas and Katehakis [5]. These papers have a different objective than ours as they focus on minimizing regret by constructing adaptive index policies that possess optimal increase rate properties. This approach has been extended to finite state and action MDPs with incomplete information (Burnetas and Katehakis [6]) and to adversarial bandits that either make no assumption whatsoever on the

process generating the payoffs of the bandits (Auer et al. [1]) or bound its variation within a "variation budget" (Besbes et al. [4]). At the time of submission we became aware of the work by Kim and Lim [18] that also study the RMAB problem but with an alternative formulation in which deviations of the transition probabilities from their point estimates are penalized, so the analysis is essentially different from ours.

2. Model Formulation

We model the RMAB problem as a finite-state, infinite horizon robust MDP in which the payoffs are discounted by $\delta \in (0,1)$ in each period and the reward obtained for pulling arm n in state s_n is given by $R_n(s_n)$. There is a set $\mathcal{N} = \{1, .., N\}$ of available arms each having state space $\mathcal{S}_n$, $n \in \mathcal{N}$. The state space for the RMAB system is $\mathcal{S} = \mathcal{S}_1 \times .. \times \mathcal{S}_N$. The controller selects an action from the action space $\mathcal{A} = \left\{a \in \{0,1\}^N : \sum_{n\in\mathcal{N}} a_n = 1\right\}$ where $a_n = 1$ and $a_n = 0$ represent pulling and not pulling arm n, respectively. Note that as in the classical MAB problem, the controller is restricted to pull one arm at a time.

For arm n, let $p_n := \big(p_n(j)\big)_{j\in\mathcal{S}_n}$ denote a probability distribution on $\mathcal{S}_n$. Let $\Delta(\mathcal{S}_n) = \left\{q \in \Re_+^{|\mathcal{S}_n|} : \sum_{j\in\mathcal{S}_n} q(j) = 1\right\}$ be the probability simplex on $\mathcal{S}_n$. Let $\mathcal{U}_n(s_n, a_n) \subseteq \Delta(\mathcal{S}_n)$ be the uncertainty set for action a_n in state $s_n \in \mathcal{S}_n$. If $a_n = 1$, then arm n transitions as a Markov process to a new state from the current state s_n with a transition probability distribution $p_n \in \mathcal{U}_n(s_n, 1)$. If $a_n = 0$, then the arm does not undergo any state transition so $\mathcal{U}_n(s_n, 0) = \left\{q \in \Delta(\mathcal{S}_n) : q(j) = 0, \forall j \neq s_n\right\}$. The transition probability distribution for the system of N arms is given by $p := (p_n)_{n\in\mathcal{N}} \in \mathcal{U}(s,a) = \mathcal{U}_1(s_1, a_1) \times \ldots \times \mathcal{U}_N(s_N, a_N)$. When the context is unambiguous, we write $\mathcal{U}_n(s_n)$ instead of $\mathcal{U}_n(s_n, 1)$.

We assume that there is a single adversary that controls the transitions of all arms. Moreover, we allow the adversary to play dynamically in the sense that the choice of particular distribution $p \in \mathcal{U}(s,a)$ in a state-action pair (s,a) at a given point in time does not limit the choices of the adversary in the future. This last assumption is known as the *rectangularity assumption* and it provides the separability that is fundamental in order to establish the robust counterpart of the Bellman equation (see Iyengar [15] and Nilim and El Ghaoui [20]). For the controller, we focus on deterministic stationary policies that dictate which arm to pull in each state $s \in \mathcal{S}$. This policy restriction is without any loss of performance (see Theorem 3.1 in Iyengar [15]).

The objective of the controller is to find a (robust) policy that maximizes the worst-case expected reward. Let $V(s)$ be the optimal reward starting in state s. Under the assumptions stated above we can formulate the RMAB problem as a sequential game with perfect information between the

controller and nature. The robust dynamic programming recursion for this game is given by

$$V(s) = \max_{a \in \mathcal{A}} \Big\{ \sum_{n \in \mathcal{N}} R_n(s_n) a_n + \delta \inf_{p \in \mathcal{U}(s,a)} \sum_{s' \in \mathcal{S}} \prod_{n \in \mathcal{N}} p_n(s'_n) V(s') \Big\}, \quad \forall\, s \in \mathcal{S}. \tag{1}$$

We conclude this section with a few remarks. First, in our model we assume a single adversary. This assumption is standard in the robust MDP literature. One could think of an alternative game in which there are N replicas of nature, one for each arm, so the controller plays against arm specific adversaries. This alternative game turns out to be intractable since the uncertainty structure does not satisfy the rectangularity assumption so formulating the Bellman equation requires a state augmentation as in Shapiro [26]. A second remark is that we allow nature to play dynamically, but when the controller follows a stationary policy, nature's best-response is to also play a stationary policy (see Lemma 3.3 in Iyengar [15] or Theorem 4 in Nilim and El Ghaoui [20]). In other words, for a given state-action pair, nature chooses the same probability distribution at any point in time. Finally, the literature provides many different ways of specifying the ambiguity sets $\mathcal{U}(s,a)$ of the transition matrices. In this paper, we use the relative entropy $\mathcal{U}_n(s_n, 1) = \big\{ p \in \Delta(\mathcal{S}_n) : \sum_{j \in \mathcal{S}_n} p(j) \log \frac{p(j)}{q(j)} \leq \beta \big\}$ to model ambiguity where β is a fixed parameter and q is a point estimate of the transition probability distribution in state s_n.

3. Robust Index Policy

The classical MAB problem has an optimal solution given by the Gittins index policy which associates a dynamic index to each arm and then plays the arm with the highest index in each period (see Frostig and Weiss [12] for several proofs of this result). In this section we define and analyze an index policy for the RMAB model in Equation (1). For that, consider the "one and a half" or 1-1/2 RMAB problem. This problem comprises an arm n with state space $\mathcal{S}_n$ and a standard arm that does not change state so it always provides a constant reward λ. Since the standard arm has no transition, its state can be omitted. From Equation (1), the Bellman equation for the 1-1/2 RMAB problem corresponds to:

$$V(s_n) = \max \Big\{ R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{j \in \mathcal{S}_n} p_n(j) V(j),\ \frac{\lambda}{(1-\delta)} \Big\}, \quad \forall\, s_n \in \mathcal{S}_n. \tag{2}$$

where $V(s_n)$ is the worst-case expected reward obtained when starting in state s_n. The maximization in the right hand side of Equation (2) has two terms representing the reward of pulling arm n and the reward of pulling the standard arm. Since arm n remains in the same state when it is rested — i.e., when it is not pulled — once it is optimal to rest arm n, it is optimal to rest it forever. Hence, the reward for pulling the standard arm equals $\lambda/(1-\delta)$.

Let $D_n(\lambda) \subseteq \mathcal{S}_n$ be the set of states for which it is optimal to rest arm n when the reward of the standard arm is λ. Following Whittle [29], an arm is indexable if $D_n(\lambda)$ increases monotonically from $\emptyset$ to $\mathcal{S}_n$ as λ increases from $-\infty$ to $+\infty$. Indexibility implies that for each state $s_n \in \mathcal{S}_n$ there exists a value of λ that makes the controller indifferent between pulling or not the standard arm. This value of λ is defined as the index of arm n in state s_n. If all arms are indexable, then the RMAB problem is indexable, in which case λ induces a consistent ordering of the arms in the sense that any arm that is rested under a reward λ will also be rested under a higher reward $\lambda' > \lambda$. As intuitive as it might seem, indexibility should not be taken for granted in bandit problems (see, for instance, Caro and Yoo [7]). Hence, our first result is to show that the RMAB problem is indexable.

Proposition 1. *The RMAB problem is indexable.*

Proof. In this proof we will denote the expected worst case reward $V(s_n)$ of equation (2) as $V^\lambda(s_n)$, where λ is the constant reward from the standard arm. Consider the 1-1/2 RMAB problem for a fixed project $n \in \mathcal{N}$. To simplify the notation, we omit the subscript n from R_n, S_n, p_n, and $\mathcal{U}_n$ henceforth in the proof. For a given state $i \in S$, we consider the function

$$\Delta f_i(\lambda) = R(i) + \delta \inf_{p(i)\in\mathcal{U}(i)} \Big(\sum_{j\in\mathcal{S}} p(j) V^\lambda(j) \Big) - \frac{\lambda}{(1-\delta)}, \tag{3}$$

We now show that $\Delta f_i(\lambda)$ is a continuous and decreasing function of λ such that $\Delta f_i(-\infty) > 0$ and $\Delta f_i(+\infty) < 0$. If that holds true then it implies that the equation $\Delta f_i(\lambda) = 0$ has a root, which in turn means that the arms in RMAB problem are indexable. In what follows we use the convergence of the robust dynamic programming algorithm so $V^\lambda(i) = \lim_{k\to\infty} V_k^\lambda(i), \forall\, i \in S$, where $V_k^\lambda(i)$ is the k-th value iteration of the Bellman recursion (see Theorem 3.2 in Iyengar [15] or Theorem 3 in Nilim and El Ghaoui [20]).

Claim 1: $\Delta f_i(\lambda)$ is a continuous function of λ.

We first show that $V_k^\lambda(i)$ is a continuous function of λ. We apply an inductive argument on k. At k=1, $V_1^\lambda(i) = \max\{\lambda, R(i)\}$. Since the functions λ and $R(i)$ are continuous in λ, the function $V_1^\lambda(i)$ is also continuous in λ. This follows from the property that maximum of two continuous functions is continuous. Let this property be true for $k = m$ i.e., $V_m^\lambda(i)$ is a continuous function of λ. Then for $k = m+1$, we get

$$V_{m+1}^\lambda(i) = \max\left\{ \frac{\lambda(1-\delta^{m+1})}{(1-\delta)}, R(i) + \delta \inf_{p\in\mathcal{U}(i)} \Big(\sum_{j\in\mathcal{S}} p(j) V_m^\lambda(j) \Big) \right\}.$$

Clearly, the term $\frac{\lambda(1-\delta^{m+1})}{(1-\delta)}$ is continuous in λ. Since the weighted sum of continuous functions is continuous, for any $p \in \mathcal{U}(i)$ we have $\sum_{j\in\mathcal{S}} p(j) V_m^\lambda(j)$, that is continuous. Also, $\inf_{p\in\mathcal{U}(i)} \Big(\sum_{j\in\mathcal{S}} p(j) V_m^\lambda(j) \Big)$

is continuous since the infimum of continuous functions is also continuous over compact subsets of the probability simplex. Hence, $R(i)+\delta \inf_{p\in\mathcal{U}(i)} \Big(\sum_{j\in\mathcal{S}} p(j)V_m^{\lambda}(j)\Big)$ is continuous and therefore $V_{m+1}^{\lambda}(i)$ is a continuous function of λ. Therefore, by principle of induction this property holds true for all k. Hence, $\Delta f_i(\lambda)$, which is a sum of continuous functions of λ is also a continuous function of λ for any state $i \in \mathcal{S}$.

Claim 2: The function $\Delta f_i(\lambda)$ is a decreasing function of λ for all states $i \in \mathcal{S}$.

Let $\lambda_1 < \lambda_2$. We show that

$$\Delta f_i(\lambda_2) < \Delta f_i(\lambda_1), \forall i \in \mathcal{S}, \tag{4}$$

by applying induction on iteration step k. For k=1 , $\Delta f_i^1(\lambda_1) = R(i) - \lambda_1$ and $\Delta f_i^1(\lambda_2) = R(i) - \lambda_2$. Hence, $\Delta f_i^1(\lambda_2) < \Delta f_i^1(\lambda_1), \forall i \in \mathcal{S}$. Now, let us assume that (4) is true for all $k \leq m$. Therefore,

$$\Delta f_i^m(\lambda_2) < \Delta f_i^m(\lambda_1), \forall i \in \mathcal{S}. \tag{5}$$

For $k = m+1$ and a given $i \in S$ we have

$$\Delta f_i^{m+1}(\lambda_2) - \Delta f_i^{m+1}(\lambda_1) = \left\{ R(i) + \delta \inf_{p\in\mathcal{U}(i)} \Big(\sum_{j\in\mathcal{S}} p(j)V_m^{\lambda_2}(j)\Big) - \frac{\lambda_2(1-\delta^{m+1})}{(1-\delta)} \right\}$$
$$- \left\{ R(i) + \delta \inf_{p\in\mathcal{U}(i)} \Big(\sum_{j\in\mathcal{S}} p(j)V_m^{\lambda_1}(j)\Big) - \frac{\lambda_1(1-\delta^{m+1})}{(1-\delta)} \right\}.$$

Let NPULL represent the action of not selecting project n and let PULL represent the action of selecting project n. Then we can write for any k and any state $j \in \mathcal{S}$:

$$V_k^{\lambda}(j) = V_k^{\lambda,NPULL}(j) + \max\left\{0, V_k^{\lambda,PULL}(j) - V_k^{\lambda,NPULL}(j)\right\} = V_k^{\lambda,NPULL}(j) + \max\left\{0, \Delta f_j^k(\lambda)\right\}.$$

Therefore, for $k = m+1$:

$$\Delta f_i^{m+1}(\lambda_2) - \Delta f_i^{m+1}(\lambda_1) = \delta \inf_{p\in\mathcal{U}(i)} \sum_{j\in\mathcal{S}} p(j) \left[V_m^{\lambda_2,NPULL}(j) + \max\{0, \Delta f_j^m(\lambda_2)\} - \frac{\lambda_2(1-\delta^m)}{(1-\delta)} \right]$$
$$-\delta \inf_{p\in\mathcal{U}(i)} \sum_{j\in\mathcal{S}} p(j) \left[V_m^{\lambda_1,NPULL}(j) + \max\{0, \Delta f_j^m(\lambda_1)\} - \frac{\lambda_1(1-\delta^m)}{(1-\delta)} \right] + (\lambda_1 - \lambda_2)$$
$$= \delta \inf_{p\in\mathcal{U}(i)} \sum_{j\in\mathcal{S}} p(j) \left[\max\{0, \Delta f_j^m(\lambda_2)\}\right] - \delta \inf_{p\in\mathcal{U}(i)} \sum_{j\in\mathcal{S}} p(j) \left[\max\{0, \Delta f_j^m(\lambda_1)\}\right] + (\lambda_1 - \lambda_2), \tag{6}$$

where the equality follows from the fact that $V_m^{\lambda,NPULL}(j) = \frac{\lambda(1-\delta^m)}{(1-\delta)}$, for $\lambda = \lambda_1, \lambda_2$. From (5) we have $\max\{0, \Delta f_j^m(\lambda_2)\} \leq \max\{0, \Delta f_j^m(\lambda_1)\}, \forall j \in \mathcal{S}$. Therefore, $\sum_j p(j) \Big[\max\{0, \Delta f_j^m(\lambda_2)\}\Big] \leq \sum_j p(j) \Big[\max\{0, \Delta f_j^m(\lambda_1)\}\Big]$ for any $p \in \mathcal{U}(i)$. Hence, $\inf_{q\in\mathcal{U}(i)} \sum_j q(j) \Big[\max\{0, \Delta f_j^m(\lambda_2)\}\Big] \leq \sum_j p(j) \Big[\max\{0, \Delta f_j^m(\lambda_1)\}\Big]$ for any $p \in \mathcal{U}(i)$ and therefore $\inf_{q\in\mathcal{U}(i)} \sum_j q(j) \Big[\max\{0, \Delta f_j^m(\lambda_2)\}\Big] \leq$

$\inf_{p\in\mathcal{U}(i)} \sum_j p(j) \left[\max\{0, \Delta f_j^m(\lambda_1)\}\right]$. From (6) we therefore have $\Delta f_i^{m+1}(\lambda_2) - \Delta f_i^{m+1}(\lambda_1) < 0$. Since i was arbitrary, the property holds for all $i \in S$ and all k, so it also holds in the limit $k \to \infty$.

Since $\Delta f_i(+\infty) < 0$ and $\Delta f_i(-\infty) > 0$, $\Delta f_i(\lambda) = 0$ has a root (when the root is not unique, it is customary to take the smallest one). Therefore, the index $\lambda(i)$ is well-defined for any state i of any robust arm, which concludes the proof. ■

An important property of the (non-robust) Gittins index is that it can be characterized using stopping times. This property provides a probabilistic interpretation of the Gittins index and has been used to develop other exact methods to compute the indices (e.g., see Robinson [24]). In our case we show that the Gittins index amounts to a robust maximization over stopping times.

Consider the 1-1/2 RMAB formulated in Equation (2) for a given arm $n \in \mathcal{N}$. Let $Z_n(t)$ denote the state of arm n at time t. The stochastic process $Z_n(t)$ is governed by the collection $\mathbf{p} \in \mathcal{U}_n$ of — possibly time-varying — transition matrices chosen by nature, where $\mathcal{U}_n$ is the set of all admissible dynamic policies that can be constructed from the ambiguity sets $\mathcal{U}_n(i, 1), i \in \mathcal{S}_n$. Let τ_n be a $Z_n(t)$ stopping time and let $\nu_n(i, \tau_n)$ be the worst-case expected discounted reward per unit of discounted time when the initial state is i and arm n is operated for a duration τ_n. Formally,

$$\nu_n(i, \tau_n) = \inf_{\mathbf{p}\in\mathcal{U}_n} \frac{\mathbb{E}_{\mathbf{p}}\left\{\sum_{t=0}^{\tau_n-1} \delta^t R_n(Z_n(t)) \,\middle|\, Z_n(0) = i\right\}}{\mathbb{E}_{\mathbf{p}}\left\{\sum_{t=0}^{\tau_n-1} \delta^t \,\middle|\, Z_n(0) = i\right\}}, \tag{7}$$

where $\mathbb{E}_{\mathbf{p}}[\cdot]$ is the expectation when the dynamics of arm n are governed by the collection of transition matrices $\mathbf{p}$. We will refer to the Gittins index for RMAB as the robust Gittins index. We can now state our result.

Proposition 2. *The robust Gittins index is given by* $\nu_n(i) = \sup_{\tau_n>0} \nu_n(i, \tau_n),\ \forall\, i \in \mathcal{S}_n,\ n \in \mathcal{N}$.

Proof. The proof is for a given arm, so we omit the subindex n. Consider the 1-1/2 RMAB problem in which the controller can retire by pulling the standard arm and receives a lump sum M. The terminal payoff M plays an important role in the proof so we make it an explicit component of the state variable. Hence, we rewrite the Bellman equation (2) as:

$$V(i, M) = \max\left\{R(i) + \delta \inf_{p(i)\in\mathcal{U}(i)} \sum_{j\in\mathcal{S}} p(i, j)V(j, M), M\right\}, \quad \forall\, i \in \mathcal{S}. \tag{8}$$

Let $V^\pi(i, M)$ denote the discounted reward under an arbitrary policy π starting from state i and with terminal payoff M. From Theorem 2.1 in Iyengar [15], it follows that $V(i, M) = \sup_\pi V^\pi(i, M)$, where the supremum is with respect to all admissible policies.

Similar to Proposition 2.2 in Frostig and Weiss [12], for a given terminal payoff M we define:

$$\begin{aligned}
&\text{Strict stopping set} && S_M = \big\{i : M > R(i) + \delta \inf_{p\in\mathcal{U}(i)} \sum_{j\in S} p(j)V(j,M)\big\}\\
&\text{Strict continuation set} && C_M = \big\{i : M < R(i) + \delta \inf_{p\in\mathcal{U}(i)} \sum_{j\in S} p(j)V(j,M)\big\}\\
&\text{Indifferent states} && I_M = \big\{i : M = R(i) + \delta \inf_{p\in\mathcal{U}(i)} \sum_{j\in S} p(j)V(j,M)\big\}.
\end{aligned}$$

These sets are disjoint and any state $i \in \mathcal{S}$ must belong to S_M, C_M or I_M. Since the arm is indexable, c.f. Proposition 1, we know that any policy which continues to activate the non-standard arm while in C_M, acts arbitrarily in I_M and retires in S_M is optimal.

For any state $i \in \mathcal{S}$, let $M(i) = \inf\big\{M : V(i,M) = M\big\}$ and $\lambda(i) = (1-\delta)M(i)$. We now show that $\nu(i)$ equals the robust Gittins index $\lambda(i)$.

Claim 1: $\nu(i) \le \lambda(i)$.

Let $y < \nu(i)$ and $M = \frac{y}{(1-\delta)}$. Since $\nu(i)$ is the supremum over all stopping times, there exists a stopping time τ for which $\nu(i,\tau) > y$. Moreover, for any $\widetilde{\mathbf{p}} \in \mathcal{U}$,

$$y < \inf_{\mathbf{p}\in\mathcal{U}} \frac{\mathbb{E}_{\mathbf{p}}\left\{\sum_{t=0}^{\tau-1}\delta^t R(Z(t)) \,\middle|\, Z(0)=i\right\}}{\mathbb{E}_{\mathbf{p}}\left\{\sum_{t=0}^{\tau-1}\delta^t \,\middle|\, Z(0)=i\right\}} \le \frac{\mathbb{E}_{\widetilde{\mathbf{p}}}\left\{\sum_{t=0}^{\tau-1}\delta^t R(Z(t)) \,\middle|\, Z(0)=i\right\}}{\mathbb{E}_{\widetilde{\mathbf{p}}}\left\{\sum_{t=0}^{\tau-1}\delta^t \,\middle|\, Z(0)=i\right\}}. \tag{9}$$

Let π be the policy that plays the non-standard arm up to time τ and then stops to collect the reward M. Hence,

$$V^{\pi}(i,M) = \inf_{\mathbf{p}\in\mathcal{U}} \mathbb{E}_{\mathbf{p}}\left\{\sum_{t=0}^{\tau-1}\delta^t R(Z(t)) + \sum_{t=\tau}^{\infty}\delta^t y \,\middle|\, Z(0)=i\right\}.$$

For any $\epsilon > 0$, there exists $\mathbf{p}^\star$ such that

$$\begin{aligned}
V^{\pi}(i,M) &> \mathbb{E}_{\mathbf{p}^\star}\left\{\sum_{t=0}^{\tau-1}\delta^t R(Z(t)) + \sum_{t=\tau}^{\infty}\delta^t y \,\middle|\, Z(0)=i\right\} - \epsilon\\
&= \mathbb{E}_{\mathbf{p}^\star}\left\{\sum_{t=0}^{\tau-1}\delta^t R(Z(t)) \,\middle|\, Z(0)=i\right\} + \mathbb{E}_{\mathbf{p}^\star}\left\{\sum_{t=\tau}^{\infty}\delta^t y \,\middle|\, Z(0)=i\right\} - \epsilon\\
&> \mathbb{E}_{\mathbf{p}^\star}\left\{\sum_{t=0}^{\tau-1}\delta^t y \,\middle|\, Z(0)=i\right\} + \mathbb{E}_{\mathbf{p}^\star}\left\{\sum_{t=\tau}^{\infty}\delta^t y \,\middle|\, Z(0)=i\right\} - \epsilon\\
&= \mathbb{E}_{\mathbf{p}^\star}\left\{\frac{y}{(1-\delta)} - \sum_{t=\tau}^{\infty}\delta^t y \,\middle|\, Z(0)=i\right\} + \mathbb{E}_{\mathbf{p}^\star}\left\{\sum_{t=\tau}^{\infty}\delta^t y \,\middle|\, Z(0)=i\right\} = \frac{y}{(1-\delta)} - \epsilon = M - \epsilon,
\end{aligned}$$

where the second inequality follows from (9). Since ϵ can be arbitrarily small, $V^{\pi}(i,M) \ge M$, which implies $V(i,M) \ge M$ and $i \in C_M \cup I_M$. Therefore, $M(i) \ge M$ and $\lambda(i) \ge y$. Since $y < \nu(i)$ was arbitrary, $\lambda(i) \ge \nu(i)$.

Claim 2: $\nu(i) \ge \lambda(i)$.

Consider any $y < \lambda(i)$ and let $M = \frac{y}{(1-\delta)}$. We define the stopping time $\tau(i, M)$ as the first passage time from i into S_M. The stopping time $\tau(i, M)$ is optimal so its discounted reward is equal to $V(i, M)$. We have that $M < M(i)$ so $i \in C_M$ and $V(i, M) > M$. Now suppose that $\nu(i, \tau(i, M)) \leq y$. Consider any $\epsilon > 0$, then there exists $\mathbf{p}^*$ such that:

$$\frac{\mathbb{E}_{\mathbf{p}^*}\left\{\sum_{t=0}^{\tau(i,M)-1} \delta^t R(Z(t)) | Z(0) = i\right\}}{\mathbb{E}_{\mathbf{p}^\star}\left\{\sum_{t=0}^{\tau(i,M)-1} \delta^t | Z(0) = i\right\}} \leq y + \epsilon(1-\delta), \tag{10}$$

which in turn implies that

$$\begin{aligned} V(i, M) &= \inf_{\mathbf{p} \in \mathcal{U}} \mathbb{E}_{\mathbf{p}}\left\{\sum_{t=0}^{\tau(i,M)-1} \delta^t R(Z(t)) + \sum_{t=\tau(i,M)}^{\infty} \delta^t y \,\Big| Z(0) = i\right\} \\ &\leq \mathbb{E}_{\mathbf{p}^*}\left\{\sum_{t=0}^{\tau(i,M)-1} \delta^t R(Z(t)) + \sum_{t=\tau(i,M)}^{\infty} \delta^t y \,\Big| Z(0) = i\right\} \\ &\leq \mathbb{E}_{\mathbf{p}^*}\left\{\sum_{t=0}^{\tau(i,M)-1} \delta^t y + \sum_{t=\tau(i,M)}^{\infty} \delta^t y \,\Big| Z(0) = i\right\} + \epsilon = \frac{y}{1-\delta} + \epsilon = M + \epsilon. \end{aligned}$$

The first equality above follows from the optimality of $\tau(i, M)$ and the last inequality follows from (10). Since ϵ is arbitrarily small, $V(i, M) \leq M$. But this contradicts $V(i, M) > M$. Hence, $V(i, M) > M$ must imply $\nu(i, \tau(i, M)) > y$, which means that $\nu(i) > y$. Since $y < \lambda(i)$ was arbitrary, $\nu(i) \geq \lambda(i)$, and the proof is complete. ■

It can be shown that the stopping time $\tau_n(i, M)$ defined in the proof of Proposition 2 achieves the supremum in Equation (7). Moreover, the policy induced by $\tau_n(i, M)$ is stationary, so nature's best-response is stationary, which means that the minimization in Equation (7) can be restricted to stationary transition matrices without any loss of optimality.

From the proof of Proposition 2 it follows that the robust Gittins index $\lambda(i)$, $i \in S$, is equal to $(1-\delta)M(i)$, where $M(i) = \inf\{M : V(i, M) = M\}$ and $V(i, M)$ is defined in Equation (8). Hence, we can invoke *the restart problem* introduced in Katehakis and Veinott, Jr. [17] and Cowan and Katehakis [8] to compute the robust indices.[1] Indeed, one can show that for a fixed initial state $i_0 \in S$, $\lambda(i_0) = (1-\delta)J(i_0)$, where $J(i)$ is the solution to the following infinite horizon robust Bellman equation:

$$J(i) = \max\left\{R(i_0) + \delta \inf_{p \in \mathcal{U}(i_0)} \sum_{j \in S} p(j)J(j),\ R(i) + \delta \inf_{p \in \mathcal{U}(i)} \sum_{j \in S} p(j)J(j)\right\}. \tag{11}$$

From Theorem 5 in Nilim and El Ghaoui [20] we have that solving the restart problem (11) takes $O(S^2(\log(\frac{1}{\epsilon}))^2)$ computations for an arm with S states, where ϵ is the desired precision. Hence,

[1] We thank the Associate Editor for bringing the restart problem to our attention.

evaluating all the indices has a complexity $O(S^3 \log_2(\frac{R_{max}}{\epsilon})(\log(\frac{1}{\epsilon}))^2)$. In Section 6 we report the time required to compute the indices for a project selection problem.

The ambiguity level of the transition probabilities affects the index of a state. Similar to the worst-case expected reward (Paschalidis and Kang [22]), the index of a state varies inversely with the level of ambiguity corresponding to the transition probabilities. We formalize this observation in Proposition 3 here below. Note that an immediate corollary is that the robust Gittins index is more conservative than its non-robust counterpart.

Proposition 3. *For any arm $n \in \mathcal{N}$ and for each $i \in S_n$, let $\overline{\mathcal{U}}_n(i) \subseteq \mathcal{U}_n(i)$ be a pair of nested ambiguity sets. Then, the corresponding indices satisfy $\lambda_{\mathcal{U}}(i) \leq \lambda_{\overline{\mathcal{U}}}(i), \forall\, i \in \mathcal{S}_n$.*

Proof. The proof again is for a fixed arm so we can drop the subindex n. Consider the 1-1/2 RMAB given by Equation (2). For a state $i \in \mathcal{S}$, Paschalidis and Kang [22] show that $V_{\mathcal{U}}(i) \leq V_{\overline{\mathcal{U}}}(i)$. Hence, $\Delta f_i(\lambda) \leq \Delta \overline{f}_i(\lambda), \forall\, \lambda$, where $\Delta f_i(\lambda)$ and $\Delta \overline{f}_i(\lambda)$ are defined in Proposition 1 for the ambiguity sets $\mathcal{U}(i)$ and $\overline{\mathcal{U}}(i)$, respectively. As shown earlier both $\Delta f_i(\lambda)$ and $\Delta \overline{f}_i(\lambda)$ are continuous and decreasing functions of λ such that $\lim_{\lambda \to +\infty} \Delta \overline{f}_i(\lambda) < 0$ and $\lim_{\lambda \to -\infty} \Delta \overline{f}_i(\lambda) > 0$. Same holds true for $\Delta f_i(\lambda)$. Let $\Delta f_i(\lambda_{\mathcal{U}}(i)) = 0$ and $\Delta \overline{f}_i(\lambda_{\overline{\mathcal{U}}}(i)) = 0$. Therefore, $\Delta \overline{f}_i(\lambda_{\mathcal{U}}(i)) \geq \Delta f_i(\lambda_{\mathcal{U}}(i)) = \Delta \overline{f}_i(\lambda_{\overline{\mathcal{U}}}(i))$, which implies that $\lambda_{\mathcal{U}}(i) \leq \lambda_{\overline{\mathcal{U}}}(i)$ and the proof is complete. ■

Since we show that a real valued index can be assigned to each state of a project, the natural question that arises is how can we characterize the optimal policy for the RMAB problem in terms of the indices. Similar to the classical MAB problem, we show that a *project-by-project retirement* (PPR) policy is optimal for the RMAB problem. For that, we introduce a retirement option in the same fashion as the proof of Proposition 2 but now for the combined bandit problem with N arms. Hence, we rewrite the Bellman equation (1) as

$$V(s, M) = \max \left\{ M, \max_{a \in \mathcal{A}} \left\{ \sum_{n \in \mathcal{N}} R_n(s_n) a_n + \delta \inf_{p \in \mathcal{U}(s,a)} \sum_{s' \in \mathcal{S}} \prod_{n \in \mathcal{N}} p_n(s'_n) V(s', M) \right\} \right\}, \quad \forall\, s \in \mathcal{S}, \quad (12)$$

where M is the terminal payoff the controller receives if it retires. For each $s_n \in \mathcal{S}_n$ and $n \in \mathcal{N}$, let $\lambda_n(s_n)$ be the robust index and let $S_M^n = \{s_n : \lambda_n(s_n) < M(1-\delta)\}$ be arm n's *retirement set*. According to the PPR policy, at any state $s \in \mathcal{S}$ the controller should permanently retire arm n if $s_n \in S_M^n$ or should work on some arm if $s_{n'} \notin S_M^{n'}$ for some $n' \in \mathcal{N}$. The PPR policy does not specify which arm to select from those that have not been retired but it identifies the arms that should no longer be pulled. We next show that such policy is indeed optimal.

Proposition 4. *There exists an optimal PPR policy for the RMAB problem.*

Proof. The construction of the proof is similar to that of Proposition 1.5.2 in Bertsekas [3] for non-robust MAB. Let $f(s, s'_i) = (s_1, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n)$. The existence of a PPR policy is equivalent to having, for all $n \in \mathcal{N}$,

$$M > R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V\big(f(s, s'_n), M\big) \quad \forall\, s \text{ with } s_n \in S_M^n \tag{13}$$

$$M \leq R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V\big(f(s, s'_n), M\big) \quad \forall\, s \text{ with } s_n \notin S_M^n. \tag{14}$$

Let the expected worst-case reward obtained by working on project n only be given by

$$V^n(s_n, M) = \max\Big\{M, R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V^n(s'_n, M)\Big\}. \tag{15}$$

Since, the expected reward obtained from working on project n only is less than the expected reward obtained from working on any project including project n we have $V^n(s_n, M) \leq V(s, M), \forall s \in \mathcal{S}$. This implies for any fixed $s \in \mathcal{S}$ and $p'_n \in \mathcal{U}_n(s_n)$ we have the following $\sum_{s'_n \in \mathcal{S}_n} p'_n(s'_n) V^n(s'_n, M) \leq \sum_{s'_n \in \mathcal{S}_n} p'_n(s'_n) V(f(s, s'_n), M)$. Hence,

$$\inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V^n(s'_n, M) \leq \sum_{s'_n \in \mathcal{S}_n} p'_n(s'_n) V(f(s, s'_n), M).$$

Since this is true for any $p'_n \in \mathcal{U}_n(s_n)$, therefore we have

$$\inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V^n(s'_n, M) \leq \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V(f(s, s'_n), M).$$

Hence, if $s_n \notin S_M^n$ then

$$\begin{aligned} M &\leq R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V^n(s'_n, M) \\ &\leq R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V(f(s, s'_n), M). \end{aligned}$$

Thereby, we obtain (14). Without loss of generality we show (13) for project $n = 1$. Let $s_{-1} = (s_2, .., s_N)$, i.e. the state of all the projects except project 1 and we define $f(s_{-1}, s'_n) = (s_2, \ldots, s_{n-1}, s'_n, s_{n+1}, \ldots, s_N)$. We then define the expected reward obtained from all the projects except project 1 as

$$V(s_{-1}, M) = \max\left\{M, \max_{n \neq 1}\Big[R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) V(f(s_{-1}, s'_n), M)\Big]\right\}. \tag{16}$$

Clearly $V(s_{-1}, M) \leq V(s, M), \quad \forall s \in \mathcal{S}$. Next we show that $V(s, M) \leq V(s_{-1}, M)$ when $s_1 \in S_M^1$. In particular, we show the following

$$V(s, M) \leq V(s_{-1}, M) + \big(V^1(s_1, M) - M\big). \tag{17}$$

Note that for $s_1 \in S_M^1$, $V^1(s_1, M) = M$ and therefore $V(s, M) = V(s_{-1}, M)$ thereby establishing that when $s_1 \in S_M^1$ it is optimal to retire project 1 and select from the other projects (i.e., (13)). To show (17) we proceed by induction on the value iteration recursions. We use the convergence of the robust dynamic programming algorithm so $V(s, M) = \lim_{k\to\infty} V_k(s, M)$, $V^1(s_1, M) = \lim_{k\to\infty} V_k^1(s_1, M)$, $V(s_{-1}, M) = \lim_{k\to\infty} V_k(s_{-1}, M), \forall\, s \in \mathcal{S}$, where $V_k(s, M), V_k(s_{-1}, M)$ and $V_k^1(s_1, M)$ are the k-th value iteration of the Bellman recursion (see Theorem 3.2 in Iyengar [15] or Theorem 3 in Nilim and El Ghaoui [20]).

For $k = 0$ we initialize $V_0(s, M) = M$, $V_0(s_{-1}, M) = M$ and $V_0^1(s_1, M) = M, \quad \forall\, s \in \mathcal{S}$. Therefore (13) is satisfied for $k = 0$. Let us assume for $k = m$:

$$V_m(s, M) \leq V_m(s_{-1}, M) + \big(V_m^1(s_1, M) - M\big). \tag{18}$$

We now show that (17) holds for $k = m + 1$. We can re-write (12) as

$$V_{m+1}(s, M) = \max\{M, q_1, q_2\} \tag{19}$$

where

$$q_1 = R_1(s_1) + \delta \inf_{p_1 \in \mathcal{U}_1(s_1)} \sum_{s_1' \in \mathcal{S}_1} p_1(s_1') V_m\big(f(s, s_1'), M\big)$$

$$q_2 = \max_{n \neq 1} \Big(R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s_n' \in \mathcal{S}_n} p_n(s_n') V_m\big(f(s, s_n'), M\big)\Big).$$

From (18) we have for any n,

$$\inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s_n' \in \mathcal{S}_n} p_n(s_n') V_m\big(f(s, s_n'), M\big)$$

$$\leq \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s_n' \in \mathcal{S}_n} p_n(s_n') \Big(V_m^1(s_1', M) + V_m\big(f(s_{-1}, s_n'), M\big) - M\Big).$$

Therefore adding $R_n(s_n)$ on either side of above inequality we obtain for any n,

$$R_n(s_n) + \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s_n' \in \mathcal{S}_n} p_n(s_n') V_m\big(f(s, s_n'), M\big) \tag{20}$$

$$\leq R_n(s_n) + \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s_n' \in \mathcal{S}_n} p_n(s_n') \Big(V_m^1(s_1', M) + V_m\big(f(s_{-1}, s_n'), M\big) - M\Big).$$

Hence for $n = 1$,

$$q_1 \leq \max\Big[M, R_1(s_1) + \delta \inf_{p_1 \in \mathcal{U}_1(s_1)} \sum_{s_1' \in \mathcal{S}_1} p_1(s_1')\Big(V_m^1(s_1', M) + V_m\big(f(s_{-1}, s_1'), M\big) - M\Big)\Big], \tag{21}$$

and for $n \neq 1$,

$$q_2 \leq \max\Big[M, \max_{n \neq 1}\Big\{R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s_n' \in \mathcal{S}_n} p_n(s_n')\Big(V_m\big(f(s_{-1}, s_n'), M\big) + V_m^1(s_1', M) - M\Big)\Big\}\Big]. \tag{22}$$

Since $V_m(s_{-1}, M) - M$ and $V_m^1(s_1, M) - M$ are constants with respect to the inner optimization problem in (21) and (22) respectively and both $V_m(s_{-1}, M), V_m^1(s_1, M) \geq M, \forall\, s \in \mathcal{S}$, we can define

$$\tilde{q}_1 = \max\Big[M, R_1(s_1) + \delta \inf_{p_1 \in \mathcal{U}_1(s_1)} \sum_{s_1' \in \mathcal{S}_1} p_1(s_1') V_m^1(s_1', M)\Big] + \big(V_m(s_{-1}, M) - M\big)$$

$$\tilde{q}_2 = \max\Big[M, \max_{n \neq 1}\Big\{R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \sum_{s_n' \in \mathcal{S}_n} p_n(s_n') V_m\big(f(s_{-1}, s_n'), M\big)\Big\}\Big] + \big(V_m^1(s_1, M) - M\big).$$

Hence $q_i \leq \tilde{q}_i$ and $M \leq \tilde{q}_i, i = 1, 2$. Therefore,

$$V_{m+1}(s, M) \leq \max\{\tilde{q}_1, \tilde{q}_2\}$$

or, $V_{m+1}(s, M) \leq \max\Big[V_{m+1}^1(s_1, M) + \big(V_m(s_{-1}, M) - M\big), V_{m+1}(s_{-1}, M) + \big(V_m^1(s_1, M) - M\big)\Big]$. We know from the Bellman recursion in Equations (15)-(16) and the initial values $V_0(s, M) = M$, $V_0(s_{-1}, M) = M$ and $V_0^1(s_1, M) = M, \forall s \in \mathcal{S}$, that $V_m^1(s_1, M) \leq V_{m+1}^1(s_1, M)$, $V_m(s_{-1}, M) \leq V_{m+1}(s_{-1}, M), \forall s \in \mathcal{S}$. Therefore, we have $V_{m+1}^1(s_1, M) + \big(V_m(s_{-1}, M) - M\big) \leq V_{m+1}^1(s_1, M) + \big(V_{m+1}(s_{-1}, M) - M\big)$ and $V_{m+1}(s_{-1}, M) + \big(V_m^1(s_1, M) - M\big) \leq V_{m+1}(s_{-1}, M) + \big(V_{m+1}^1(s_1, M) - M\big)$. Hence, by principle of induction (17) holds for any k. ■

Proposition 4 shows that the robust Gittens indices are informative in the sense that they indicate which arms are the most promising (i.e., the arms that should not be retired). Then one could expect that a policy based on these indices should perform well. Let the robust index (RI) policy be the policy that chooses to play in each period the arm that has the highest robust Gittins index in its current state among all the other arms. In Section 6 we show that the RI policy is indeed near optimal. However, in contrast to the non-robust MAB, the RI policy in general is not optimal. The suboptimality arises from the fact that nature may choose a different transition probability distribution for the same state of an arm depending on the current state of the other arms. We demonstrate this by a counterexample. Let there be two projects each with three states as indicated in Figure 1.

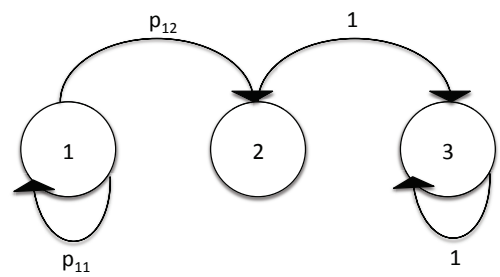

Figure 1: State transition diagram for arm 1

For simplicity we assume discrete ambiguity sets with two possible probability distributions. Let $\mathcal{U}_1(1) = \big\{(0.1, 0.9, 0), (0.7, 0.3, 0)\big\}$ and $\mathcal{U}_2(1) = \big\{(0.8, 0.2, 0), (0.9, 0.1, 0)\big\}$. The rewards are given as $R_1(1) = 1, R_1(2) = 6, R_1(3) = 0$ and $R_2(1) = 2$ and $R_2(2) = 3, R_2(3) = 0$. Let the discount factor be $\delta = 0.9$. The action space $a \in \big\{(1, 0), (0, 1)\big\}$, where $a = (1, 0)$ and $a = (0, 1)$ correspond to selecting arms 1 and 2, respectively.

The RI and optimal policies differ for states (1,1) and (1,2). The robust value functions for states (1,1) and (1,2) are given by,

$$V(1,1) = \max\Big[R_1(1) + \delta \inf_{p_1} \big\{p_1(1)V(1,1) + (1 - p_1(1))V(2,1)\big\},$$

$$R_2(1) + \delta \inf_{p_2} \big\{p_2(1)V(1,1) + (1 - p_2(1))V(1,2)\big\}\Big]$$

$$V(1,2) = \max\Big[R_1(1) + \delta \inf_{p_1} \big\{p_1(1)V(1,2) + (1 - p_1(1))V(2,2)\big\}, R_2(2) + \delta V(1,3)\Big].$$

The optimal probability distributions correspond to $p_1^\star = (0.7, 0.3, 0), p_2^\star = (0.8, 0.2, 0)$ for state $(1, 1)$ and $p_1^\star = (0.1, 0.9, 0)$ for state $(1, 2)$. Hence, nature's choice of the transition probability distribution for state 1 of arm 1 depends on the state of arm 2 leading to suboptimality of the index policy. Interestingly, the RI policy is suboptimal even in the alternative robust model in which the controller plays against N replicas of nature (see the discussion at the end of Section 2). The reason is that the adversaries have perfect information so their actions will internalize the state of the other arms, which again introduces dependencies across arms. If the RI policy is evaluated under maximum expected reward criterion instead of max-min criterion, then it reduces to the non-robust Gittins index policy for point estimates of the transition probability. The RI policy can still be used as a heuristic policy for the RMAB problem, but it is computationally intensive. In the next section we introduce the Lagrangian index policy which performs as well as the RI policy but can be computed more efficiently.

4. Lagrangian Index Policy

Evaluating robust indices is more expensive than evaluating non-robust indices. Hence, we introduce the Lagrangian index (LI) policy which chooses to play in each period the arm with the highest Lagrangian index that will be defined later in this section. The Lagrangian indices are evaluated by computing the Gittins indices of a classical MAB problem. The MAB problem is constructed by first relaxing the constraint that only one arm can be pulled at a time by the controller to obtain

$$L^{\lambda}(s) = \lambda + \max_{a \in \mathcal{A}} \Big\{ \sum_{n=1}^{N} (R_n(s_n) - \lambda) a_n + \delta \inf_{p \in \mathcal{U}(s,a)} \sum_{s' \in S} \prod_{n=1}^{N} p_n(s'_n) V(s') \Big\} \tag{23}$$

This relaxation collapses to a system with N loosely coupled arms. This is equivalent to each arm being played by the controller and an independent copy of nature. We note that this approach resembles the approximate method in [11]. More broadly, this Lagrangian technique has shown to be effective in solving weakly coupled MDPs (see for instance [14] and the references therein).

We can eventually show that solving the optimization problem given by equation (23) is equivalent to solving a system of N 1-1/2 RMAB problems for a given λ. Moreover, we obtain the following upper bound for the optimal reward starting in $s \in \mathcal{S}$:

$$V(s) \leq L^{\star}(s) = \min_{\lambda \geq 0} L^{\lambda}(s) = \min_{\lambda \geq 0} \frac{\lambda}{1-\delta} + \sum_{n=1}^{N} L_n^{\lambda}(s_n), \tag{24}$$

where

$$L_n^{\lambda}(s_n) = \max_{a_n \in \{0,1\}} \Big\{ (R_n(s_n) - \lambda) a_n + \delta \inf_{p_n \in \mathcal{U}_n(s_n, a_n)} \sum_{s'_n \in \mathcal{S}_n} p_n(s'_n) L_n^{\lambda}(s'_n) \Big\}.$$

The formal proof of Equation (24) is available from the authors and a similar proof can be found in Caro and Yoo [7].

Since nature's best-response is a stationary policy, for a given λ the above system assigns a transition distribution to each arm by solving the N 1-1/2 RMAB problems. We fix $\lambda^{\star} = \arg\min_{\lambda \geq 0} L^{\lambda}(s)$ and evaluate the policy of nature for each independent arm which gives a fixed state transition probability distribution $p^{\lambda^{\star}} \in \mathcal{U}(s,a)$ for the system of N arms. The Gittins index policy for the corresponding non-robust MAB with state transition probability distribution $p^{\lambda^{\star}}$ is the LI policy. If the maximum expected reward criterion is applied in place of max-min reward criterion, then the LI policy reduces to the Gittins index policy for point estimate of the transition probability. Since, evaluating the Lagrangian indices takes as much computational effort as required to evaluate the indices of a classical MAB, it is computationally faster to evaluate than the robust indices.

5. Partial Robust Value Iteration

Evaluating any policy for the RMAB problem poses the following challenges 1) the exponentially large state space prevents the exact computation of the robust value iteration and 2) ambiguous transition probability for any state action pair does not allow simulation of policies to estimate the expected reward. Hence, we introduce partial robust value iteration (PRV), that combines both the above ideas to obtain an approximate value of the expected reward for a given policy. In PRV we first simulate the policy for the point estimates of the transition probabilities (recall that for the ambiguity sets we assume the entropy model introduced in Section 2). The states that are visited during the simulation are identified as the high priority states. The remaining states are identified as the low priority states. The low priority states have their expected reward truncated to the immediate reward. The robust value iteration is applied only to the high priority states. Let $S_H \subset \mathcal{S}$ be the set of high priority states and $S_L \subset \mathcal{S}$ be the set of low priority states such that $S_H \cup S_L = \mathcal{S}$. If any state $s' \in S_L$ then $V(s') = \sum_{n \in \mathcal{N}} R_n(s'_n)a_n$. If $s' \in S_H$ then the robust Bellman equation is the same as (1) for a given $a \in \mathcal{A}$.

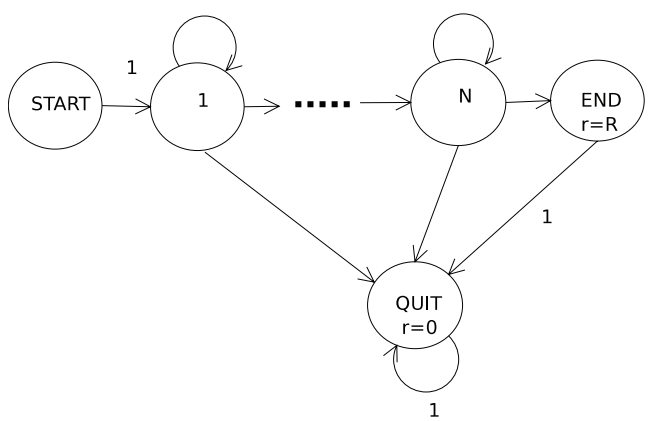

Figure 2: State transition diagram of a project

We illustrate the idea with an example of an RMAB problem where each arm has the state transition diagram given by Figure 2 with START (S) and QUIT (Q) as the starting and quitting states respectively for each project. The high priority states are identified as those states which appear in the sample path from the starting state $s_S = (S, .., S)$ to the quitting state $s_Q = (Q, .., Q)$. Depending on the probability distribution and the policy, different sample paths can be constructed between s_S and s_Q states. For instance from any state s_n of the n-th project the possible state transitions on selecting the project are to states $s_n, s_n + 1$ or Q. If $s \in S_H$, the expected reward

for action $a_n = 1$ is computed according to the Bellman recursion

$$R_n(s_n) + \delta \inf_{p_n \in \mathcal{U}_n(s_n)} \Big\{ p_n(s_n)V(s) + p_n(s_n+1)V(s_1,.,s_n+1,.,s_N) + p_n(Q)V(s_1,.,Q,.,s_N) \Big\},$$

whereas if $s \in S_L$, then the expected reward is approximated by the myopic reward $R_n(s_n)$. Hence, only the value of the high priority states are updated during value iteration.

Let $V^\pi(s)$ for $s \in \mathcal{S}$ be the worst-case expected reward under a policy π and let $v^\pi(s)$ be the value obtained through PRV for the fixed policy π. We now show that PRV provides a lower bound for the worst-case expected reward of a given policy.

Proposition 5. *For a given policy π, $v^\pi(s) \leq V^\pi(s), \forall\ s \in \mathcal{S}$*

Proof. The result is clearly true for $s \in S_L$. For $s \in S_H$ we use an inductive argument on the value iteration recursions. We use the convergence of the robust dynamic programming algorithm so $V^\pi(s) = \lim_{k\to\infty} V_k^\pi(s)$ and $v^\pi(s) = \lim_{k\to\infty} v_k^\pi(s), \forall\, s \in S$, where $V_k^\pi(s)$ and $v_k^\pi(s)$ are the k-th value iteration of the Bellman recursion for the fixed policy π (see Theorem 3.2 in Iyengar [15] or Theorem 3 in Nilim and El Ghaoui [20]). We initialize $v_0^\pi(s) = V_0^\pi(s) = \sum_{n\in\mathcal{N}} R_n(s_n)a_n, \forall s \in \mathcal{S}$. Hence, the result holds true for $k = 0$. Let us assume $v_k^\pi(s) \leq V_k^\pi(s), \forall s \in \mathcal{S}$. We show that the result holds true for $k+1$. For any fixed $q \in \mathcal{U}(s,a)$ we have

$$\sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} q_n(s'_n) v_k^\pi(s') \leq \sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} q_n(s'_n) V_k^\pi(s'), \forall s' \in \mathcal{S}.$$

Therefore for any action a,

$$\inf_{p\in\mathcal{U}(s,a)} \sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} p_n(s'_n) v_k^\pi(s') \leq \sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} q_n(s'_n) V_k^\pi(s'), \forall s' \in \mathcal{S},$$

This is true for any $q \in \mathcal{U}(s,a)$. Hence, it is also true for the optimal distribution, i.e.,

$$\inf_{p\in\mathcal{U}(s,a)} \sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} p_n(s'_n) v_k^\pi(s') \leq \inf_{p\in\mathcal{U}(s,a)} \sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} p_n(s'_n) V_k^\pi(s'), \forall s' \in \mathcal{S}.$$

Adding a constant, we have

$$\sum_n R_n(s_n)a_n + \inf_{p\in\mathcal{U}(s,a)} \sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} p_n(s'_n) v_k^\pi(s') \leq \sum_n R_n(s_n)a_n + \inf_{p\in\mathcal{U}(s,a)} \sum_{s'\in\mathcal{S}} \prod_{n\in\mathcal{N}} p_n(s'_n) V_k^\pi(s'), \forall s' \in \mathcal{S}.$$

Therefore, for a fixed policy π we have $v_{k+1}^\pi(s) \leq V_{k+1}^\pi(s), \forall s \in \mathcal{S}$. By principle of induction it is true for any k. ■

6. Numerical Experiments

We compare the performance of Robust Index policy (RI) and Lagrangian Index policy (LI) with the Non-robust Index policy (NRI) to evaluate the value of the robust approach. NRI is the index policy evaluated for the point estimates of the transition probability. Moreover, we consider two greedy approaches, the MYO and the CON policies. MYO is a myopic index policy where the index for a state action pair is the immediate reward obtained in the current state of the arm specified by the action. The comparison of LI and RI policies against the MYO policy shows the value of being forward-looking. The CON policy is further explained in the following subsection where we first describe the sequential project selection (SPS) problem followed by the performance of the heuristic policies for the SPS problem.

6.1 Sequential Project Selection Problem

Activities like research, development or exploration, progress sequentially in nature, i.e. at any given point of time the project can either move to the next stage towards completion or terminate altogether. The SPS model is loosely based on the ideas given in Roberts and Weitzman [23]. The controller has to choose a project from multiple projects at every decision epoch. Each project also has the possibility of making no progress and staying in the same state as it was in the previous period. If a project is abandoned in any intermediary stage then a small reward can be earned or a cost can be incurred. This reward is much smaller than the reward obtained on successful completion of the project. The intermediary rewards or cost indicate some partial benefit obtained or expense incurred by working on a project up to the intermediate stage.

Figure 2 represents a state transition diagram for a general SPS. We see from Figure 2 that the transition from states START, END and QUIT are known for sure. The ambiguity is only present in the transition probabilities between the intermediary states $(1, .., N)$ shown in the figure. Our objective is to evaluate the expected worst case profit for the state where all the projects are in the initial START state. In most practical scenarios the probability of transitioning from an intermediary state to any state in the following time period is not known accurately in advance so we formulate the SPS problem as an RMAB problem.

6.2 Experimental Setting

The objective of the numerical experiments is to compare the performance of the four heuristic policies, the RI, LI, NRI and MYO policies for the SPS problem. All the codes were written in

MATLAB 7.9.0 (R2009b). We will represent a problem instance by (N, S) which would imply N projects each having S states. For all our computational experiments the rewards for START and QUIT were zero and for END we randomly assigned $R \in [50, 100]$. The intermediary states were randomly assigned rewards $r \in [1, 10]$. The inner minimization problem for the robust Bellman equation (1) corresponding to an action is a convex optimization problem which can be solved by formulating its dual and applying the bisection algorithm on the dual problem (see Section 6, Nilim and El Ghaoui [20]). The accuracy of the bisection algorithm for solving the inner optimization problem was fixed as $\frac{(1-\delta)\epsilon}{2\delta}$ with $\epsilon = .001$ for our computations.

For the upper bound we use the fact that if we fix nature's policy, the optimal expected reward for the corresponding non-robust MAB system is an upper bound for the original robust MAB problem. Therefore the upper bound (UB) was evaluated by fixing nature's policy as $p^{\lambda^\star}$ (from §4) and evaluating the optimal reward for the corresponding non-robust MAB system. We used Monte Carlo simulation with relative estimation error equal to 0.02% and confidence level of 95% to estimate the expected reward for the optimal policy. The individual RI, LI, NRI, and MYO policies were evaluated by PRV. Any state appearing in any sample path was included in S_H (high priority states as described in §5). Figure 3 shows the relationship between the expected reward obtained from a policy π by PRV and the robust value iteration, the optimal reward for RMAB, and the upper bound. We vary three parameters of the model to compare the performance of different policies, the discount factor δ, the ambiguity level β, and the uncertainty level for the point estimate of transition probability γ, where $p(s_n, s_n) = \frac{\gamma}{2}$, $p(s_n, s_{n+1}) = 1 - \gamma$, $p(s_n, QUIT) = \frac{\gamma}{2}, \forall n, \gamma > 0$. We also compare the RI, LI and NRI policies with a conservative strategy CON, that selects the arm with the highest terminal reward (R) and plays it until it reaches END or QUIT and then plays the arm with second highest reward and so on. The CON policy is conservative in the sense that it finishes projects one by one.

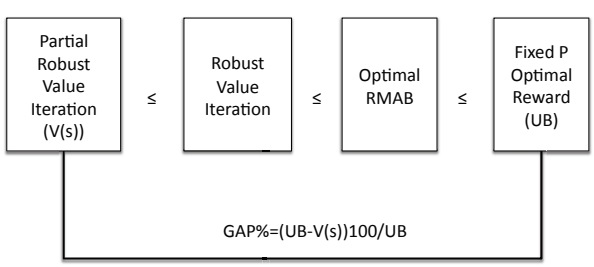

Figure 3: Suboptimality Gap

6.3 Computational Results

We begin this section by reporting the computational time required to compute all the robust indices of a single arm using the restart problem (11). The results are shown in Table 1. For comparison we include the computational time of an alternative method that computes the index of state $i \in S$ by finding the root of $\Delta f_i(\lambda) = 0$ using a bisection search.[2]. The bisection search takes $O(\log_2(\frac{R_{max}}{\epsilon}))$ iterations to converge, where R_{max} is the maximum reward among all the states of the arm. In contrast, the restart method solves a single robust MDP, and therefore it is about an order of magnitude faster. As expected, the run times increase with the number of states S and the discount δ but for the restart method they remain within an hour even for an instance with a hundred states. Note that $\beta = 0$ corresponds to a non-robust MAB and represents the computational time of the NRI and LI heuristics.

Table 1: Robust indices computational times in seconds ($\gamma = 0.33$)

S	δ	$\beta = 0$		$\beta = 0.5$		$\beta = 1$	
		restart	bisection	restart	bisection	restart	bisection
6	0.90	0.2	2.2	1.3	9.4	1.2	9.0
6	0.95	0.4	5.1	3.0	22.7	2.7	21.5
6	0.98	1.0	15.8	8.6	72.8	7.6	68.0
50	0.90	12.8	160.2	112.4	945.1	105.5	901.5
50	0.95	27.8	383.0	254.0	2264.6	240.3	2191.5
50	0.98	76.8	1132.3	713.9	6962.5	698.6	6527.8
100	0.90	52.2	636.0	460.8	3868.6	420.4	3502.8
100	0.95	114.3	1506.1	1019.4	8920.1	931.5	8554.8
100	0.98	311.4	4576.6	2904.1	28307.0	2881.3	27859.0

In order to study the effect of the discount factor δ we fix the ambiguity level $\beta = 0.5$ and the uncertainty level $\gamma = 0.33$ for all the intermediate states of all the arms. Then we vary $\delta = 0.98, 0.95, 0.90$ and report the performance gaps against the upper bound in Table 1. The heuristic with the smallest suboptimality gap for each instance is shown in boldface.. We observe that LI and RI policies outperform the NRI, CON, and MYO policies and have negligible difference between their performance gaps.

In Table 3 we report the performance gaps of the heuristic policies for various levels of ambiguity (β) with a fixed discount factor $\delta = 0.95$ and uncertainty level $\gamma = 0.33$. Note that the upper bound values decreases with increase in ambiguity. For higher β nature has more freedom to choose the worst case reward. For lower β the suboptimality gap of NRI policy is less since low β implies less ambiguity. From Table 3 we conclude that the LI and RI policies outperform the other heuristic

[2]From the proof of Proposition 1 it follows that $\Delta f_i(\lambda)$ defined in Equation (3) is monotone decreasing in λ

policies and are near optimal for varying levels of ambiguity β.

Table 2: % Gap ($\beta = 0.5$, $\gamma = 0.33$)

(N,S)	δ	LI (%)	RI (%)	NRI (%)	MYO (%)	CON (%)	UB
(10,6)	0.90	0.3187	**0.3138**	2.1629	82.8426	18.7111	36.2257
	0.95	**0.1571**	0.1631	1.3013	87.7161	11.0747	55.4729
	0.98	**0.0376**	0.0406	0.9029	90.4053	4.7795	74.9872
(20,6)	0.90	0.4906	**0.4897**	5.0769	78.9409	20.6150	42.1161
	0.95	0.3508	**0.3475**	3.5177	87.0391	15.7795	74.1790
	0.98	**0.1029**	0.1030	1.4344	91.6034	7.5408	120.1303
(30,6)	0.90	0.4819	**0.4712**	3.3284	94.4459	40.5325	40.7007
	0.95	0.5093	**0.4976**	3.2371	96.5627	32.3179	74.9461
	0.98	**0.1692**	0.1747	1.5960	97.8994	16.6486	132.6080
(40,6)	0.90	0.7551	**0.7403**	4.0805	79.3059	48.2339	41.2047
	0.95	0.7322	**0.7202**	3.3361	88.5751	38.0846	80.4769
	0.98	0.3525	**0.3517**	2.4118	94.0113	22.1388	160.5414

Table 3: % Gap ($\delta = 0.95$, $\gamma = 0.33$)

(N,S)	β	LI (%)	RI (%)	NRI (%)	MYO (%)	CON (%)	UB
(10,6)	0.1	0.2118	**0.2112**	0.2883	85.7835	6.0880	84.7780
	0.5	**0.1571**	0.1631	1.3013	87.7161	11.0747	55.4729
	1.0	**0.0453**	0.0491	2.5863	88.8776	14.5199	44.0211
(20,6)	0.1	**0.3508**	**0.3508**	1.1353	84.2952	10.3297	101.5785
	0.5	0.3507	**0.3475**	3.5177	87.0391	15.7795	74.1790
	1.0	**0.0592**	0.0608	9.4320	87.4193	23.9041	63.1585
(30,6)	0.1	0.4421	**0.4415**	1.0075	92.4825	22.7256	105.2099
	0.5	0.5093	**0.4976**	3.2371	96.5627	32.3179	74.9461
	1.0	0.0279	**0.0235**	7.8563	97.8817	40.3796	63.5097
(40,6)	0.1	0.4825	**0.4819**	0.9025	86.8301	24.5267	106.6997
	0.5	0.7322	**0.7202**	3.3361	88.5751	38.0846	80.4769
	1.0	**0.1678**	**0.1678**	8.5239	88.7917	50.9535	70.6248

We next compare the performance of all the heuristic policies by varying $\gamma = 0.67, 0.33, 0.1$. When $\gamma = \frac{2}{3}$, the point estimates of the transition probability distribution is uniform, whereas when γ is close to zero the distribution becomes deterministic. Table 4 shows how the performance of the indices changes with uncertainty (γ) for a given level of ambiguity (β) and discount factor (δ). We vary $\beta = 0.1, 0.5, 1$ to show low, medium, high ambiguity levels respectively. Table 4 shows that RI and LI policies outperform the NRI, MYO, and CON policies for different uncertainty and ambiguity levels. We find that the RI and LI policies perform near optimal for various levels of uncertainty (γ) and ambiguity (β). Whereas, the NRI policy performs better at low level of ambiguity for all levels of uncertainty.

Table 4: % Gap ($\delta = 0.95$)

(N,S)	β	γ	LI (%)	RI (%)	NRI (%)	MYO (%)	CON (%)	UB
(10,6)	0.1	0.10	**0.0492**	**0.0492**	0.3560	84.1284	2.3830	117.5610
		0.33	0.2118	**0.2112**	0.2883	85.7835	6.0880	84.7780
		0.67	**0.1459**	**0.1459**	0.6931	87.6012	13.0936	58.3055
	1.0	0.10	**0.1501**	**0.1501**	3.6890	87.3399	8.4946	61.8857
		0.33	**0.0453**	0.0491	2.5863	88.8776	14.5199	44.0211
		0.67	**0.8815**	**0.8815**	1.1497	89.8436	18.1786	38.0787
(20,6)	0.1	0.10	**0.0914**	**0.0914**	0.2202	85.7580	6.0463	134.2527
		0.33	**0.3754**	**0.3754**	0.6787	88.8946	10.7306	100.5750
		0.67	0.4255	**0.4196**	0.8363	92.1123	17.9561	72.9551
	1.0	0.10	**0.3153**	**0.3153**	3.2342	91.2877	13.0274	77.4274
		0.33	0.0164	**0.0133**	3.0316	93.5058	22.6043	57.8670
		0.67	**0.0305**	**0.0305**	9.8970	94.4082	40.0821	51.8860
(30,6)	0.1	0.10	**0.1597**	**0.1597**	0.4742	82.6907	9.5125	136.7363
		0.33	0.3608	**0.3607**	0.9270	84.9939	17.8691	105.3124
		0.67	0.7365	**0.7122**	1.7942	86.9983	30.2916	80.9258
	1.0	0.10	0.4471	**0.4406**	6.3301	87.3291	23.5304	84.4796
		0.33	0.0892	**0.0784**	6.0151	88.7002	37.3583	67.4664
		0.67	**0.3786**	**0.3786**	18.2438	89.2283	49.9370	60.3755

Overall since the gaps obtained from LI and RI are consistently less than 1% for all the instances, we can conclude that they are both near optimal heuristic policies for the robust SPS problem. Evaluating the Lagrangian indices is computationally less expensive than computing the Robust indices. Therefore, the LI policy is a suitable heuristic for the SPS problem.

7. Conclusion

The RMAB problem can be modeled as a game between the controller and nature. We see that the presence of nature as an adversary makes the transitions of arms dependent on the states of the other arms. We show that the RI policy is not optimal but performs better than the MYO or the NRI policies for the SPS problem. We propose the LI policy that is obtained by solving an MAB problem in which we relax the constraint that one arm has to be selected in every decision epoch for the RMAB problem. All the heuristics are evaluated empirically on randomly generated instances of the SPS problem. We find that the LI and RI policies are comparable in their performance and near optimal, but the RI policy is computationally more expensive than the LI policy. Hence, the LI policy would be the preferable heuristic policy for the SPS problem.

There are many possible extensions for the RMAB problem. These include all the variants of the classical MAB problem. For instance, resources might have to be allocated among more than one project at a time (Pandelis and Teneketzis [21]), new projects might arrive (Whittle

[28]), all projects may change state (Whittle [29]), or there might be constraints linking the bandits (Denardo, Feinberg and Rothblum [10]). Studying the results for these variants in a robust setting is an avenue for future work. On the other hand, our numerical results are based on the assumption that the ambiguity set is given by the Kullback-Liebler divergence from the point estimate of transition probabilities. Another possible extension could be to analyze the performance of the LI and RI policies under other ambiguity models. Finally, the robust formulation cares about the worst-case scenario, which might be regarded as an extreme case. Therefore, future research could focus on finding formulations and policies that balance the maximum expected reward and the worst-case.

References

[1] P. Auer, N. Cesa-Bianchi, Y. Freund, and R.E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2003.

[2] J.D. Bagnell, A. Y. Ng, and J. Schneider. Solving uncertain markov decision problems. Technical report, CMU-RI-TR-01-25, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA., 2001.

[3] D. Bertsekas. *Dynamic Programming and Optimal Control, Vol. II.* Athena Scientific, 2000.

[4] O. Besbes, Y. Gur, and A. Zeevi. Optimal exploration-exploitation in multi-armed-bandit problems with non-stationary rewards. 2013.

[5] A.N. Burnetas and M.N. Katehakis. Optimal adaptive policies for sequential allocation problems. *Advances in Applied Mathematics*, 17(2):122 – 142, 1996.

[6] A.N. Burnetas and M.N. Katehakis. Optimal adaptive policies for markov decision processes. *Mathematics of Operations Research*, 22(1):222–255, 1997.

[7] F. Caro and O.S. Yoo. Indexability of bandit problems with response delays. *Probability in the Engineering and Informational Sciences*, 24:349–374, 2010.

[8] W. Cowan and M.N. Katehakis. Multi-armed bandits under general depreciation and commitment. *Probability in the Engineering and Informational Sciences*, 29:51–76, 2015.

[9] E. Delage and S. Mannor. Percentile optimization for markov decision processes with parameter uncertainty. *Operations Research*, 58(1):203–213, 2010.

[10] E.V. Denardo, E.A. Feinberg, and U.G. Rothblum. The multi-armed bandit, with constraints. *Ann*, 208:37–62, 2013.

[11] N. Dimitrov, S. Dimitrov, and S. Chukova. Robust decomposable markov decision processes motivated by allocating school budgets. *European Journal of Operational Research*, 239:199–213, 2014.

[12] E. Frostig and G. Weiss. Four proofs of gittins multiarmed bandit theorem. *Annals of Operations Research*, DOI 10.1007/s10479-013-1523-0, 2014.

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

[14] Y. Gocgun and A. Ghate. Lagrangian relaxation and constraint generation for allocation and advanced scheduling. *Computers & Operations Research*, 39:2323–2336, 2012.

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

[16] M.N. Katehakis and H. Robbins. Sequential choice from several populations. *Proceedings of the National Academy of Sciences of the United States of America*, 92(19):8584–8585, 1995.

[17] M.N. Katehakis and A.F. Veinott, Jr. The multi-armed bandit problem: decomposition and computation. *Mathematics of Operations Research*, 22(2):262–268, 1987.

[18] M.J. Kim and A. Lim. Robust multi-armed bandit problems. *Management Science*, 2014.

[19] T.L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

[20] A. Nilim and L. El Ghaoui. Robust control of markov decision processes with uncertain transition matrix. *Operations Research*, 53(5):780–798, 2005.

[21] D.G. Pandelis and D. Teneketzis. On the optimality of the gittins index rule for multi-armed bandits with multiple plays. *Mathematical Methods of Operations Research*, 50:449–461, 1990.

[22] I.C. Paschalidis and S.C. Kang. A robust approach to markov decision problems with uncertain transition probabilities. In *Proceedings of the 17th IFAC World Congress*, pages 408–413, 2008.

[23] K. Roberts and M.L. Weitzman. Funding criteria for research, development, and exploration projects. *Econometrica*, 49(5):1261–1288, 1981.

[24] D.R. Robinson. Algorithms for evaluating the dynamic allocation index. *Operations Research Letters*, 1:72–74, 1982.

[25] J.K. Satia and R.E. Lave. Markovian decision processes with uncertain transition probabilities. *Operations Research*, 21(3):728–740, 1973.

[26] A. Shapiro. A dynamic programming approach to adjustable robust optimization. *Operations Research Letters*, 39:83–87, 2011.

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

[28] P. Whittle. Arm-acquiring bandits. *The Annals of Probability*, 9(2):284–292, 1981.

[29] P. Whittle. Restless bandits: Activity allocation in a changing world. *Journal of Applied Probability*, 25:287–298, 1988.

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