

# Long-Term Resource Allocation Fairness in Average Markov Decision Process (AMDP) Environment

Ganesh Ghalme\*

Technion Israel Institute of Technology  
Haifa, Israel  
ganeshg@campus.technion.ac.il

Vishakha Patil\*

Indian Institute of Science  
Bangalore, India  
patilv@iisc.ac.in

Vineet Nair\*

Technion Israel Institute of Technology  
Haifa, Israel  
vineetn90@gmail.com

Yilun Zhou\*

Massachusetts Institute of Technology  
Cambridge, MA, United States  
yilun@mit.edu

## ABSTRACT

Fairness has emerged as an important concern in automated decision-making in recent years, especially when these decisions affect human welfare. In this work, we study fairness in temporally extended decision-making settings, specifically those formulated as Markov Decision Processes (MDPs). Our proposed notion of fairness ensures that each state’s long-term visitation frequency is at least a specified fraction. This quota-based notion of fairness is natural in many resource-allocation settings where the dynamics of a single resource being allocated is governed by an MDP and the distribution of the shared resource is captured by its state-visitation frequency. In an *average*-reward MDP (AMDP) setting, we formulate the problem as a bilinear saddle point program and, for a generative model, solve it using a Stochastic Mirror Descent (SMD) based algorithm. The proposed solution guarantees a *simultaneous approximation* on the expected average-reward and fairness requirement. We give sample complexity bounds for the proposed algorithm and validate our theoretical results with experiments on simulated data.

## KEYWORDS

Fairness, Markov Decision Process, Reinforcement Learning

### ACM Reference Format:

Ganesh Ghalme\*, Vineet Nair\*, Vishakha Patil\*, and Yilun Zhou\*. 2022. Long-Term Resource Allocation Fairness in Average Markov Decision Process (AMDP) Environment. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 9 pages.

## 1 INTRODUCTION

Algorithms are increasingly used to make important decisions that impact human lives. While algorithmic decision-making frameworks offer increased efficiency, speed and scalability, their potential bias and unfairness have led to several concerns. For instance, studies have shown that the traditional algorithms may be unfair towards certain demographics of the population in recidivism prediction [13], loan and credit lending [5], online advertising [2],

and recommendation systems [35]. These concerns have led to a surge in research efforts aimed at ensuring fairness in algorithmic decision-making frameworks [4]. A large body of work in fair machine learning has focused on either one-shot settings such as classification [14, 22] or sequential but static settings such as multi-armed bandits where reward distributions are stationary [9, 27, 37]. However, in many real-world applications, the algorithm’s decisions may have long-term impact to the states and rewards of the system. The study of fairness in such temporally-extended decision-making settings, often modeled using the reinforcement learning framework [30], is still in its infancy.

We introduce the problem of state-visitation fairness in Markov Decision Processes (MDPs). Informally, our fairness notion requires that each state of the MDP be visited with a pre-specified minimum frequency. In particular, a policy  $\pi$  with stationary state distribution  $v^\pi$ , is called *fair* if  $v_s^\pi \geq \rho_s$  for every state  $s$ , where  $\rho_s \in [0, 1]$  specifies the fairness constraints and is given as input to the algorithm. Motivated by recent works on fairness in multi-armed bandits (MAB) [24, 27] that enforce minimum frequency on the selection of each arm, our *quota-based* notion of fairness is natural in dynamic resource allocation settings where the transition dynamics of the resource being allocated is governed by an MDP and an algorithm is required to *equitably* divide the shared resource. Although MABs can be considered a special case of MDPs with one action per arm on a shared dummy state, distinctively different techniques are needed to satisfy the analogously defined constraints on state visitation frequency for general MDPs due to long term implications of the taken actions.

As a concrete example, consider the task of scheduling the jobs of  $C$  clients to run on one shared server, where  $C$  is assumed to be fixed and known *a priori*. At every time step, each client submits some number of jobs, and the server decides how many jobs to run for each client. The maximum number of jobs that the server can run at every step is  $N$ . From the server’s perspective, this task can be modeled as an MDP [34]: a state is a vector of the form  $(n_1, \dots, n_C)$  representing the number of remaining jobs for the clients. At this state, the server can take an action  $a = (m_1, \dots, m_C)$ , where  $m_c \leq n_c$  is the number of jobs that the server runs for client  $i$  at the current time step, subject to the constraint that  $\sum_{c=1}^C m_c \leq N$ . The transition models the job execution and (potentially stochastic) job arrival. The server gets rewards depending on the number of finished jobs. In the simple case where client  $c$  pays  $r_c$  for each

\*Author names appear alphabetically. Vineet Nair is now at Google Research, India. Code and appendix are available at <https://yilunzhou.github.io/fair-average-mdp/>.

*Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022)*, P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online. © 2022 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

completed job, the total reward at each step is  $\sum_{c=1}^C m_c r_c$ , which obviously leads to a strict prioritization toward the highest paying client for a reward maximizing agent. However, in many cases we want to enforce some quality of service (QoS) to every client. Such a guarantee can be modeled with our notion of fairness by requiring minimum frequencies on states with low values of  $n_c$  (i.e. remaining jobs) for each low-paying client  $c$ .

Similar fairness requirements are present in other resource allocation settings such as taxi dispatching and postal service, and they can also be framed as state-visitation guarantees. In addition, since the service is “long-running”, the average reward captures the long term profitability of the service provided better than the discounted reward, and motivates us to consider the case of average-reward MDP, which we refer to as AMDP.

Last, our notion of fairness can also generalize demographic parity [7] into a temporally extended setting. Consider that a company wishes to ensure racial diversity in their employee base, for an extended period of time rather than for a specific hiring decision. We can represent the overall employee profile as a state in an MDP, from which we can compute diversity statistics. The company takes various human-resource decisions on a day-to-day basis that may cause the employee demographics to change. While the diversity objective may not be feasible to achieve at all time (e.g. due to random resignation decisions made by employees), the company still wants to or is required to maintain the diversity with high frequency, which can be naturally encoded as minimum frequency constraints on states that satisfy the diversity objective.

## 1.1 Our Contributions

In this paper, we make two contributions, a new notion of fairness as constraints on a MDP and an algorithm to solve for the optimal policy under the constraints. On the former, we introduce the fairness notion of *minimum resource allocation guarantee* in the MDP setting. This fairness notion is practically significant, capturing many real world applications and complementing existing notions of fairness in MDP such as approximate action-fairness guarantees [17] and demographic parity [33].

On the latter, our work contributes to the long line of literature on constrained MDPs [3], which has been mostly used to ensure the safety of exploration [1]. These formulations usually encode safety as upper bounds on state-dependent cost functions, while our desired fairness constraints are lower bounds on state-visitation frequencies. With sample access to the transition function (i.e. being able to sample the next state given the current state and action), we formulate the problem as a bilinear saddle point problem and present an algorithm (Algorithm 1 that uses the classical stochastic mirror descent (SMD) framework to simultaneously satisfy the fairness constraint and achieve reward maximization asymptotically (Theorems 1 and 2). Its running time depends on the required approximation threshold  $\epsilon$ , the fairness constraint parameter  $d_\rho$ , mixing time  $t_{mix}$ , number of states  $n$  and number of actions  $m$  as  $O(nm\epsilon^{-2}(1+d_\rho)^2 t_{mix}^2 \log(nm))$ .

Recently, Jin and Sidford [18] proposed the first algorithm with sample complexity bounds to compute an approximately-optimal policy for unconstrained AMDPs. Our algorithm is similar in spirit

to this unconstrained algorithm in [18] but requires novel analytical techniques to prove the simultaneous guarantee on fairness and reward. The main technical novelty in our work is how the primal variables are bounded. Since the fairness constraints introduce new primal variables in the linear program (Fair-LP (P) in Section 4.1), the analysis of [18] does not lend itself to a straightforward extension. Instead, Lemma 2 non-trivially use structure of the constraint matrix to bound the domain of primal variables in our algorithm. Another technical novelty is that we restrict the dual space by incorporating the fairness constraint explicitly in the domain, which simplifies the objective function (Eq. 5 to Eq. 6) and helps achieve the objective. In comparison, Jin and Sidford [18] only compute a feasible policy with no regard to optimality. To the best of our knowledge, our work is the first to achieve the simultaneous guarantee with sample complexity bounds.

## 2 RELATED WORK

Recently, there has been growing interest in studying fairness in sequential decision-making. For example, Creager et al. [10] propose causal modeling of dynamical systems to address fairness, Zhang et al. [38] study how algorithmic decisions impact the evolution of feature space of the underlying population modeled as an MDP, and D’Amour et al. [11] study the impact of feedback dynamics on long-term fairness via simulations.

The study of fairness in reinforcement learning (RL) was initiated by Jabbari et al. [17], who extend the *meritocratic fairness* notion defined by Joseph et al. [19] in the MAB setting to the MDP setting. Under this notion, a policy is fair if, with high probability, an action with a lower long-term reward is not favored over an action with a higher long-term reward. This notion of fairness can be classified as procedural fairness and is different from our outcome-based notion of fairness where the fairness guarantee can be quantified in terms of the state-visitation frequency. Doroudi et al. [12] study the problem of off-policy policy selection in RL under similar fairness constraints. In terms of the fairness constraints, the work closest to ours is that of Wen et al. [33], which models the agents as states of an MDP and study demographic parity with respect to reward to the agents. However, their work is different from ours in three key aspects: 1) they study discounted-reward MDPs in contrast to our average-reward MDPs, 2) they focus primarily on the setting where the model is known, whereas our main contribution is for the generative model, and 3) they model the constraints in terms of a reward to the agents in contrast to our fairness constraints which capture the absolute long-term state-visitation frequency.

The unconstrained AMDP problem has been extensively studied in the literature [6, 20, 25]. If the model is known and the MDP is unichain, then Altman [3] and Puterman [29] showed that an optimal policy can be computed by solving a linear program. With a generative model (i.e. a simulator that can sample from the transition function and compute the reward function [21]), the SMD approach [8, 26] was recently used by Wang [32] and Jin and Sidford [18] to compute an approximately optimal policy. Furthermore, Jin and Sidford [18] proposed an algorithm to compute a feasible, but not necessarily optimal, policy for constrained AMDPs.

We formulate MDPs with fairness guarantees as constrained MDPs (CMDPs), [3], which have been studied extensively in the

safety setting [1]. The policy search seeks to maximize the reward while ensuring certain upper bound frequency on some (high-risk or error) states [15, 31]. By contrast, our constraints are defined as lower bounds on states, for which existing techniques could not be adapted in a straightforward manner. Last, another line of literature studies the problem where the objective itself is to achieve a specified state-visitation frequency (or some function of it) in the absence of reward signals [16, 23].

In the rest of the paper, Section 3 introduces the Fair-AMDP problem. Section 4 formulates the solution as a linear program and provide necessary background on the SMD framework. Section 5 presents the concrete algorithm implementation and Section 6 presents the theoretical analysis. Section 7 presents experimental results to validate the proposed algorithm. Finally, Section 8 presents a discussion of the work and some future directions.

### 3 FAIR-AMDP MODEL

A discrete Markov Decision Process (MDP) is a sequential decision-making framework denoted by the tuple  $\langle \mathcal{S}, \mathcal{A}, \Gamma, \mathbf{r}, \mathbf{u} \rangle$ . At each step  $t \geq 1$ ,  $s_t \in \mathcal{S}$  denotes the state of the MDP at time  $t$ . A decision-maker takes an action  $a_t \in \mathcal{A}$ , receives a finite reward  $r_t = r_{s_t, a_t}$ , and the MDP transitions to a state  $s_{t+1}$  according to the transition probability function  $\Gamma : \mathcal{S} \times \mathcal{A} \rightarrow \Delta^{|\mathcal{S}|}$  where  $\Delta^{|\mathcal{S}|}$  is the simplex set of distributions over states. Without loss of generality, we assume that the rewards are non-negative and depend only on  $s_t$  and  $a_t$  and not on  $s_{t+1}$ . The initial state  $s_1$  is sampled from the initial-state distribution  $\mathbf{u} \in \Delta^{|\mathcal{S}|}$ .

Let  $\pi$  be a stochastic policy with  $\pi_{s,a}$  denoting the probability with which action  $a$  is taken in state  $s$ . Each policy  $\pi$  induces a stationary distribution over the state space denoted by  $\mathbf{v}^\pi \in \Delta^n$ . One of the popular optimization problems under an MDP framework [3, 29] is to find a policy that maximizes the long-term expected average-reward given by

$$\lim_{T \rightarrow \infty} \frac{1}{T} \cdot \sum_{t=1}^T \mathbb{E}_\pi [r_t]. \quad (1)$$

Throughout the paper, we consider a finite MDP with  $\mathcal{S} = \{1, 2, \dots, n\}$  and  $\mathcal{A} = \{1, 2, \dots, m\}$ . Let  $\ell = nm$  denote the total number of state-action pairs. For clarity, we often use  $(s, a)$  to index the  $((s-1)m + a)$ -th entry of vectors in  $\mathbb{R}^\ell$ . Thus, we can equivalently represent the transition function  $\Gamma$  as a matrix of dimension  $\ell \times n$ , where  $\Gamma_{(s,a),s'}$  is the probability of going to  $s'$  when taking  $a$  at  $s$ . Similarly, we can represent the reward function  $\mathbf{r}$  as a vector in  $\mathbb{R}^\ell$ , where  $r_{(s,a)}$  gives the reward of taking  $a$  at  $s$ . For notational convenience, we define a matrix form  $\Pi \in \mathbb{R}^{\ell \times n}$  for the policy  $\pi$  where  $\Pi_{(s,a),s} = \pi_{s,a}$  and  $\Pi_{(s,a),s'} = 0$  for all  $s \neq s'$ . Thus,  $\Gamma^\pi := \Pi^T \Gamma$  is the transition matrix of the Markov chain induced by  $\pi$ . In this work, we restrict ourselves to ergodic MDPs, defined below.

**Definition 1.** (Ergodicity) A Markov decision process is ergodic if Markov chain on the state induced by any policy is ergodic. A Markov chain is ergodic if there exists a positive integer  $T_0$ , such that for all pairs of states  $s_i, s_j$ , if the chain is started at  $s_i$ , the probability of being in state  $s_j$  is non-zero for all time after  $T_0$ .

Intuitively, the state transition in an ergodic MDP mixes across all states without showing any periodic oscillations. Next, we define its mixing time as follows.

**Definition 2.** (Mixing Time) The mixing time of a given MDP  $\langle \mathcal{S}, \mathcal{A}, \Gamma, \mathbf{r}, \mathbf{u} \rangle$  is given by  $t_{\text{mix}} = \max_{\pi} t_\pi$  where,

$$t_\pi = \arg \min_{t \geq 1} \left[ \max_{\mathbf{u}} \left[ \|\Gamma^{\pi^T} \mathbf{u} - \mathbf{v}^\pi\|_1 \leq 1/2 \right] \right].$$

The mixing time  $t_{\text{mix}}$  of an ergodic MDP captures how fast the Markov chain induced by any policy converges to its corresponding stationary distribution.

**Assumption 1.** The MDP instance  $\langle \mathcal{S}, \mathcal{A}, \Gamma, \mathbf{r}, \mathbf{u} \rangle$  is ergodic.

The ergodicity condition enables us to formulate the problem of finding an optimal policy for an AMDP problem as a linear program (Section 4.1). We consider a constrained version of it where the constraints are in terms of the minimum state-visitation frequency. In particular, we study the Fair-AMDP problem with the following notion of fairness.

**Definition 3.** Let  $\boldsymbol{\rho} \in [0, 1]^n$  such that  $\sum_s \rho_s \leq 1$ . Then, a policy  $\pi$  is called  $\boldsymbol{\rho}$ -fair if  $v_s^\pi \geq \rho_s$  for all  $s \in [n]$ .

A Fair-AMDP instance is denoted as  $\langle \mathcal{S}, \mathcal{A}, \Gamma, \mathbf{r}, \mathbf{u}, \boldsymbol{\rho} \rangle$ . We note that a Fair-AMDP instance may not even have a feasible policy. For example, consider an AMDP instance with  $n = 2, m = 1$  and  $\Gamma = (1 - \alpha, \alpha; 1 - \alpha, \alpha)$ . Here, the stationary distribution for the unique policy  $\pi$  is  $\mathbf{v}^\pi = (1 - \alpha, \alpha)$ . There does not exist a  $\boldsymbol{\rho}$ -fair policy for any  $\boldsymbol{\rho}$  with  $\rho_2 > \alpha$ . We restrict our attention to  $\boldsymbol{\rho}$  such that  $\rho_s < 1/n$  for all  $s \in \mathcal{S}$  unless otherwise specified. In this paper, we assume that the problem is feasible.

**Assumption 2.** There exists a  $\boldsymbol{\rho}$ -fair policy.

Ensuring feasibility and identifying infeasibility can be done with one technical addition to the generative model. We introduce a fair action  $a^*$  available at each state with a reward that is strictly lower than the one on any state-action pair. One choice of  $a^*$  is such that  $\Gamma((s, a^*), s') = 1/n$ . In other words, taking this action “resets” the agent to a state selected uniformly at random. We make the following additional remarks regarding the above assumption.

- (1) Given a Fair-AMDP with action  $a^*$ , a policy  $\pi$  that chooses action  $a^*$  in all the states  $s$  has  $\mathbf{v}^\pi = (1/n \dots 1/n)$  and hence  $\mathbf{v}^\pi > \boldsymbol{\rho}$  implying that the Fair-AMDP is strictly feasible. Further, if the Fair-AMDP instance is guaranteed to have a feasible solution without using the fair-action then there is an optimal policy that has zero probability of choosing action  $a^*$  at any state  $s$ .
- (2) For a particular  $\boldsymbol{\rho}$ , one may relax the assumption to the following: the fair-action  $a^*$  is such that  $\Gamma((s, a^*), s') > \max_s \rho_s$  for all  $s, s' \in [n]$ . Such an assumption is seemingly necessary to compute a fair-algorithm in the generative model where there is only stochastic access to the transition probability matrix via state-action queries.

## 4 SOLUTION APPROACH

As mentioned before, we assume access to a *generative model*, which can be used to sample the next state  $s'$  given the current state  $s$  and action  $a$  according to  $\Gamma$  and compute the reward  $\mathbf{r}_{s,a}$ .

#### 4.1 Linear Program for Fair-AMDP

We first recall the linear programming solution to solve the (unconstrained) AMDP when the transition and reward functions are known. The formulation is based on the Bellman equation for optimal policy, and is derived in detail in textbooks [e.g. 29]. Let  $\hat{\mathbf{I}} \in \mathbb{R}^{\ell \times n}$  be such that  $\hat{\mathbf{I}}((s, a), s') = 1$  if  $s = s'$  and 0 otherwise.

UC-LP (D)	UC-LP (P)
$\max_{\mathbf{x} \in \Delta^\ell} \mathbf{x}^T \mathbf{r}$	$\min_{\lambda > 0, \beta} \beta$
subject to $(\hat{\mathbf{I}} - \Gamma)^T \mathbf{x} = \mathbf{0}$	subject to $(\hat{\mathbf{I}} - \Gamma)\lambda + \mathbf{r} \leq \beta^T \mathbf{1}$

We focus our attention on UC-LP (D). From the optimal solution  $\mathbf{x}$ , we can derive the policy  $\pi$  as

$$\pi_{s,a} = \frac{x_{s,a}}{\sum_{a'=1}^m x_{s,a'}}$$

It is easy to verify the following:

- (1) For  $\mathbf{v}^\pi = (\sum_a x_{1,a}, \dots, \sum_a x_{n,a})$ , we have  $\mathbf{x} = \Pi \mathbf{v}^\pi$ ,
- (2)  $\mathbf{v}^\pi$  is the stationary state distribution corresponding to the  $\pi$ , as  $(\hat{\mathbf{I}} - \Gamma)^T \mathbf{x} = (\mathbf{I} - \Gamma^T)^T \mathbf{v}^\pi = \mathbf{0}$ , and
- (3)  $\hat{\mathbf{I}} \mathbf{x} = \mathbf{v}^\pi$ .

In particular, from (3) we conclude that  $x_{s,a}$  is the average state-action visitation probability of the state-action pair  $(s, a)$ . Note that Assumption 1 ensures that  $\mathbf{v}^\pi > \mathbf{0}$ , for every such  $\pi$ .

It follows that the desired fairness guarantee can be achieved by ensuring that  $\mathbf{x}$  satisfies  $\sum_a x_{s,a} \geq \rho_s \forall s$  such that  $\rho_s > 0$ . We assume, without loss of generality, that  $\rho_s > 0 \forall s$ . We state the FAIR-LP primal/dual below, where  $\mathbf{C} \in \mathbb{R}^{n \times \ell}$  is such that the  $(s', (s, a))$ -entry is  $1/\rho_s$  if  $s = s'$  and 0 otherwise. Assumption 2 guarantees their feasibility.

FAIR-LP (D)	FAIR-LP (P)
$\max_{\mathbf{x} \in \Delta^\ell} \mathbf{x}^T \mathbf{r}$	$\min_{\lambda > 0, \mu > 0, \beta} \beta - \mu^T \mathbf{1}$
subject to $(\hat{\mathbf{I}} - \Gamma)^T \mathbf{x} = \mathbf{0}, \mathbf{C}\mathbf{x} \geq \mathbf{1}$	subject to $(\hat{\mathbf{I}} - \Gamma)\lambda + \mathbf{r} + \mathbf{C}^T \mu \leq \beta^T \mathbf{1}$

**Proposition 1.** Let  $(\mathcal{S}, \mathcal{A}, \Gamma, \mathbf{r}, \mathbf{u}, \rho)$  be a Fair-AMDP instance and  $\pi$  be a policy with  $\mathbf{v}^\pi$  as the induced stationary distribution. Further, let  $\mathbf{x} = \Pi \mathbf{v}^\pi$ . Then,  $\pi$  is a  $\rho$ -fair policy if and only if  $\mathbf{x}$  is a feasible solution of FAIR-LP.

Proposition 1 establishes that if  $\Gamma$  and  $\mathbf{r}$  are known, we can compute an optimal  $\rho$ -fair policy by solving FAIR-LP. However, when the model parameters are unknown or partially known, even the problem of verifying the feasibility is difficult. For instance, if  $\alpha$  is not known in our example from Section 3, then without the fair-action, it is impossible to determine whether a policy is  $\rho$ -fair. Let  $(\lambda^*, \mu^*, \beta^*)$  be an (optimal) solution of FAIR-LP (P). First, we note that  $\lambda^*$  is not unique, as for every  $c \in \mathbb{R}$ ,  $(\lambda^* + c \cdot \mathbf{1}, \mu^*, \beta^*)$  is also an optimal solution. Hence, we may assume without loss of generality that  $\lambda^*$  is orthogonal to the stationary distribution  $\mathbf{v}^{\pi^*}$  of the optimal policy  $\pi^*$  of a given Fair-AMDP instance. We conclude this section by stating two important lemmas regarding the nature of  $\mu^*$  and  $\lambda^*$ . The presence of a strictly feasible solution is used in Claim 1, which is in turn used to prove Lemma 2.

**Claim 1.** Let  $\lambda^*, \mu^*, \beta^*$  be the solution to the FAIR-LP (P). Then  $\mu_s^* \leq (n\rho_s)/(1 - n\rho_s)$  for all  $s \in [n]$ .

**PROOF.** Since  $\lambda^*, \mu^*$ , and  $\beta^*$  satisfies the FAIR-LP primal, we have

$$(\hat{\mathbf{I}} - \Gamma)\lambda^* + \mathbf{r} + \mathbf{C}^T \mu^* \leq \beta^* \cdot \mathbf{1}.$$

Consider the policy  $\pi^f$  which only pulls the fair-action at every point. Then  $\mathbf{v}^f = (\frac{1}{n} \dots \frac{1}{n})$  is the stationary distribution corresponding to  $\pi^f$ . Since the reward for pulling the control action in any state is zero,  $(\Pi^f \mathbf{v}^f)^T \mathbf{r} = 0$ . Multiplying the above equation by  $(\Pi^f \mathbf{v}^f)^T$ , we have

$$\sum_s \frac{1}{n\rho_s} \mu_s^* \leq \beta^*$$

Further, using strong duality we have  $\mathbf{r}^T \mathbf{x}^* + \sum_s \mu_s^* = \beta^*$ . Hence,

$$\sum_s \frac{1}{n\rho_s} \mu_s^* \leq \mathbf{r}^T \mathbf{x}^* + \sum_s \mu_s^*.$$

Since  $\mathbf{r} \in [0, 1]^\ell$ , and  $\mathbf{x} \in \Delta^\ell$ ,

$$\sum_s \frac{1 - n\rho_s}{n\rho_s} \cdot \mu_s^* \leq \mathbf{r}^T \mathbf{x}^* \leq 1.$$

This implies for all  $s$ ,  $\mu_s^* \leq (n\rho_s)/(1 - n\rho_s)$  assuming  $\rho_s < 1/n$ .  $\square$

**Remark:** If we had prior knowledge that there exists a strictly feasible policy  $\pi$ , then it can be shown that  $\mu_s^* (\frac{\rho_s}{\rho_s} - 1) \leq 1 - (\Pi \mathbf{v})^T \mathbf{r}$ , where  $(\Pi \mathbf{v})^T \mathbf{r}$  is the reward of the strictly feasible policy  $\pi$  and is at least 0.

We first state the following useful lemma (Lemma 5 by Jin and Sidford [18]) here without a proof. This lemma is used in the proof of Lemma 2 and Theorem 2.

**Lemma 1.** Give a mixing AMDP with mixing time  $t_{\text{mix}}$ , a policy  $\pi$ , and its transition probability matrix  $\Gamma^\pi \in \mathbb{R}^{n \times n}$  and stationary distribution  $\mathbf{v}^\pi$ , the following holds:

$$\|(\mathbf{I} - \Gamma^\pi + \mathbf{1}\mathbf{v}^{\pi T})^{-1}\|_\infty \leq 2t_{\text{mix}}.$$

Our proposed algorithm's convergence time depends on the mixing time of the MDP via the following lemma.

**Lemma 2.** Let  $\lambda^*, \mu^*, \beta^*$  be the solution to the FAIR-LP (P). Then  $\|\lambda^*\|_\infty \leq M := 2t_{\text{mix}}(1 + d_\rho)$ , where  $d_\rho = \max_s \frac{n}{1 - \rho_s n}$ .

**PROOF.** Since  $\lambda^*, \mu^*$ , and  $\beta^*$  is the solution to the FAIR-LP primal,

$$(\hat{\mathbf{I}} - \Gamma)\lambda^* + \mathbf{C}^T \mu^* + \mathbf{r} \leq \beta^* \cdot \mathbf{1}. \quad (2)$$

Let  $\pi^*$  be the optimal policy corresponding to  $\mathbf{x}^*$ , and let  $\Pi^* \in \mathbb{R}^{\ell \times n}$  and  $\mathbf{v}^*$  be its corresponding matrix and stationary distribution respectively. Note that  $\Gamma^* = (\Pi^*)^T \cdot \Gamma$  is the probability transition matrix corresponding to the Markov chain induced by  $\pi^*$ . Multiplying Equation 2 by  $(\Pi^*)^T$  from the left, and using the KKT condition and that  $v_s^* > 0$  for all  $s$  (as the MDP is ergodic; Assumption 1) we have

$$(\mathbf{I} - \Gamma^*)\lambda^* + \mathbf{D}_\rho \mu^* + (\Pi^*)^T \mathbf{r} = \beta^* \cdot \mathbf{1}, \quad (3)$$

where  $\mathbf{D}_\rho$  is the  $n \times n$  diagonal matrix with its  $s$ -th entry being  $\frac{1}{\rho_s}$ . It is easy to see that  $(\Pi^*)^T \mathbf{C}^T = \mathbf{D}_\rho$ , and  $(\Pi^*)^T \mathbf{1} = \mathbf{1}$ . Denote

$(\Pi^*)^T \mathbf{r}$  as  $\mathbf{r}^*$ . Since  $\langle \boldsymbol{\lambda}^*, \mathbf{v}^* \rangle = 0$ , Equation 3 can be rewritten as follows

$$(\mathbf{I} - \Gamma^* + \mathbf{1}(\mathbf{v}^*)^T) \boldsymbol{\lambda}^* + \mathbf{D}_\rho \boldsymbol{\mu}^* + \mathbf{r}^* = \beta^* \cdot \mathbf{1}.$$

This implies

$$\begin{aligned} \boldsymbol{\lambda}^* &= (\mathbf{I} - \Gamma^* + \mathbf{1}(\mathbf{v}^*)^T)^{-1} (\beta^* \cdot \mathbf{1} - \mathbf{D}_\rho \boldsymbol{\mu}^* - \mathbf{r}^*) \\ \|\boldsymbol{\lambda}^*\|_\infty &\leq \|(\mathbf{I} - \Gamma^* + \mathbf{1}(\mathbf{v}^*)^T)^{-1}\|_\infty \|\beta^* \cdot \mathbf{1} - \mathbf{D}_\rho \boldsymbol{\mu}^* - \mathbf{r}^*\|_\infty \\ &\leq 2t_{\text{mix}} \|\beta^* \cdot \mathbf{1} - \mathbf{D}_\rho \boldsymbol{\mu}^* - \mathbf{r}^*\|_\infty \\ &\leq 2t_{\text{mix}} \left(1 + \max_s \frac{n}{1 - \rho_s n}\right) \end{aligned} \quad (4)$$

The first inequality in Equation 4 follows from Lemma 1, and the second follows by showing that  $\|\beta^* \cdot \mathbf{1} - \mathbf{D}_\rho \boldsymbol{\mu}^* - \Pi^T \mathbf{r}\|_\infty \leq (1 + \max_s \frac{n}{1 - \rho_s n})$ , which we argue next. By strong duality we have  $\beta^* - \sum_s \mu_s^* = \mathbf{r}^T \mathbf{x}^*$ . Since  $\mathbf{r} \in [0, 1]^\ell$  and  $\mathbf{x}^* \in \Delta^\ell$ ,  $0 \leq \beta^* \leq 1 + \sum_s \mu_s^*$ . Hence from Claim 1, we have for any  $s$

$$-1 - \sum_s \frac{n\rho_s}{1 - n\rho_s} \leq \beta^* - \frac{\mu_s^*}{\rho_s} - r_s^* \leq 1 + \max_s \frac{n}{1 - n\rho_s}.$$

Now as  $\sum_s \rho_s < 1$ ,  $1 + \sum_s \frac{n\rho_s}{1 - n\rho_s} \leq 1 + \max_s \frac{n}{1 - n\rho_s}$ . Hence,  $\|\beta^* \cdot \mathbf{1} - \mathbf{D}_\rho \boldsymbol{\mu}^* - \mathbf{r}^*\|_\infty \leq (1 + d_\rho)$ .  $\square$

The bound on  $\|\boldsymbol{\lambda}^*\|_\infty$  crucially restricts the search space of the primal variable  $\boldsymbol{\lambda}$  in the stochastic mirror descent (SMD) approach in Section 4.2 and 4.3 and enables us to give convergence guarantee for the proposed algorithm.

## 4.2 Minimax Formulation

We formulate Fair-AMDP as a bilinear saddle point problem using the Lagrangian of FAIR-LP,

$$h(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \beta) = \mathbf{r}^T \mathbf{x} + \boldsymbol{\lambda}^T (\hat{\mathbf{I}} - \Gamma) \mathbf{x} + \boldsymbol{\mu}^T (\mathbf{C}\mathbf{x} - \mathbf{1}) + \beta(1 - \mathbf{1}^T \mathbf{x}).$$

Let  $\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \beta^*$  be a solution to

$$\min_{\boldsymbol{\lambda} \in \mathbb{R}^n, \boldsymbol{\mu} \geq 0, \beta} \max_{\mathbf{x} \geq 0} h(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \beta). \quad (5)$$

From Assumption 2, we have that FAIR-LP problem has a feasible solution. Hence,  $\mathbf{x}^*$  is the solution to FAIR-LP (D), and  $\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*$  and  $\beta^*$  is the solution to FAIR-LP (P). Moreover, from the KKT optimality conditions,  $(\boldsymbol{\mu}^*)^T (\mathbf{C}\mathbf{x}^* - \mathbf{1}) = 0$ .

Let  $\Delta_\rho^\ell := \{\mathbf{x} \in \Delta^\ell \mid \sum_a x_{s,a} \geq \rho_s \text{ for all } s\}$ , and  $\mathbb{B}_{2M}^n := \{\boldsymbol{\lambda} \mid \|\boldsymbol{\lambda}\|_\infty \leq 2M\}$ , where  $M$  is as defined in Lemma 2. Then, for every  $\mathbf{x}^* \in \Delta_\rho^\ell$ , note the following: a)  $\beta(1 - \mathbf{1}^T \mathbf{x}) = 0$ , b)  $\mathbf{0} \in \arg \min_{\boldsymbol{\mu} \geq 0} \boldsymbol{\mu}^T (\mathbf{C}\mathbf{x} - \mathbf{1})$ . Hence, using Lemma 2 it follows that  $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \beta^*)$  is a solution to the optimization problem in Equation 5 if and only if  $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$  is the solution to the following problem

$$\min_{\boldsymbol{\lambda} \in \mathbb{B}_{2M}^n} \max_{\mathbf{x} \in \Delta_\rho^\ell} f(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{r}^T \mathbf{x} + \boldsymbol{\lambda}^T (\hat{\mathbf{I}} - \Gamma) \mathbf{x}. \quad (6)$$

Furthermore, we have

$$h(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \beta^*) = f(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \mathbf{r}^T \mathbf{x}^*.$$

Though  $\|\boldsymbol{\lambda}^*\|_\infty \leq M$ , increasing the domain of  $\boldsymbol{\lambda}$  from  $M$  to  $2M$  helps in the proof of Theorem 2 which provides simultaneous approximation guarantees on the optimality and fairness of the policy that is constructed by Algorithm 1.

Next, we define the Gap function, which quantifies the closeness of a given feasible solution  $(\mathbf{x}, \boldsymbol{\lambda})$  to the optimal solution.

**Definition 4.** The gap function  $\text{Gap} : \Delta_\rho^\ell \times \mathbb{R}^n \rightarrow \mathbb{R}_+$  is defined as

$$\text{Gap}(\mathbf{x}, \boldsymbol{\lambda}) = \max_{\mathbf{x}' \in \Delta_\rho^\ell} f(\mathbf{x}', \boldsymbol{\lambda}) - \min_{\boldsymbol{\lambda}' \in \mathbb{B}_{2M}^n} f(\mathbf{x}, \boldsymbol{\lambda}'). \quad (7)$$

It is easy to see that the Gap function is non-negative and  $\text{Gap}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = 0$ . We say that  $(\mathbf{x}, \boldsymbol{\lambda})$  is an  $\varepsilon$ -approximate solution to Equation 6 if  $\text{Gap}(\mathbf{x}, \boldsymbol{\lambda}) \leq \varepsilon$ .

## 4.3 Stochastic Mirror Descent (SMD)

In this section, we state the SMD framework and then briefly describe how we use it to compute an (expected)  $\varepsilon$ -approximate solution to the optimization problem in Equation 6. This is done using the ghost iterate technique [18] to compute the approximate solution using only stochastic query access to  $\Gamma$  and  $\mathbf{r}$  (i.e. the generative model). We begin with two useful definitions.

**Definition 5.** (Strong Convexity) Let  $\mathcal{X} \subset \mathbb{R}^n$  be a convex set. A differentiable function  $R : \mathcal{X} \rightarrow \mathbb{R}$  is said to be  $\alpha$ -strongly convex with respect to norm  $\|\cdot\|$  if  $R(\mathbf{y}) \geq R(\mathbf{x}) + \langle \nabla R(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\alpha}{2} \|\mathbf{y} - \mathbf{x}\|^2$  for all  $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ .

**Definition 6.** (Distance Generating Function) Let  $R : \mathcal{X} \rightarrow \mathbb{R}$  be a continuously differentiable, strongly convex, real-valued function on a convex set  $\mathcal{X} \subset \mathbb{R}^n$ . For any  $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ , the distance from point  $\mathbf{x}$  to  $\mathbf{y}$  is given by  $V_{\mathbf{x}}(\mathbf{y}) = R(\mathbf{y}) - R(\mathbf{x}) - \langle \nabla R(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$ .<sup>1</sup>

The SMD framework is a stochastic approximation approach of finding a solution to a stochastic convex program [26]. In particular, the SMD algorithm is a special type of stochastic gradient descent (SGD) algorithm where the updates are computed in the *mirrored* space. It provides an iterative procedure to select points from a convex space  $\mathcal{X}$  with stochastic query access to underlying parameters. First, a suitable strongly convex regularizer  $R$  catering to the geometry of  $\mathcal{X}$  is designed. Then, at each iterate  $t$ , an unbiased estimator of gradient  $\mathbf{g}$  given by  $\tilde{\mathbf{g}}_t$  and a step size  $\eta_t$  is computed. This estimator, along with the previous iterate value  $\mathbf{x}_t$  and step size  $\eta_t$ , is used to compute the next iterate as given below:

$$\mathbf{x}_{t+1} = \arg \min_{\mathbf{x} \in \mathcal{X}} \langle \eta_t \tilde{\mathbf{g}}_t, \mathbf{x} \rangle + V_{\mathbf{x}_t}(\mathbf{x}). \quad (8)$$

Since the regularizer is strictly convex and differentiable, the SMD update can be equivalently written in the (mirrored) gradient space as follows

$$\nabla R(\mathbf{x}_{t+1}) = \nabla R(\mathbf{x}_t) - \eta_t \tilde{\mathbf{g}}_t.$$

The strong convexity of the regularizer  $R$  implies the uniqueness of the mapping of  $\nabla R(\cdot)$ . For more details, the reader is referred to the works by Nemirovski et al. [26] and Carmon et al. [8].

In this paper, we use the SMD framework to obtain an approximate solution to the optimization problem defined in Equation 6. Note that, in our problem, there are two convex spaces:  $\Delta_\rho^\ell$  corresponding to  $\mathbf{x}$ , and  $\mathbb{B}_{2M}^n$  corresponding to  $\boldsymbol{\lambda}$ . First, we choose strongly convex regularizers for the respective spaces and initialize the *constant* step sizes  $\eta^{\mathbf{x}}$  and  $\eta^{\boldsymbol{\lambda}}$ , respectively. Then, at each iterate, we compute bounded estimators of the gradients  $\mathbf{g}^{\mathbf{x}}$  and  $\mathbf{g}^{\boldsymbol{\lambda}}$  given

<sup>1</sup>This function is also called as the Bregman divergence or the prox-function.

by  $\tilde{g}_t^x$  and  $\tilde{g}_t^\lambda$ , respectively. Here,  $g^x$  and  $g^\lambda$  are the gradients of  $f$  with respect to  $x$  and  $\lambda$ , respectively. Finally, we use the SMD update rule given in Equation 8 to compute the iterate value for the respective spaces. In this paper, we consider bounded gradient estimates of the gradients as defined below.

**Definition 7.** (Jin and Sidford [18]) Given following properties on mean, scale, and variance of an estimator  $\tilde{g}$  of the gradient  $g$ :

- (1) unbiasedness:  $\mathbb{E}[\tilde{g}] = g$ ,
- (2) bounded maximum entry:  $\|\tilde{g}\|_\infty \leq c$  with probability 1, and
- (3) bounded second moment  $\mathbb{E}[\|\tilde{g}\|^2] \leq v$ ,

we say that  $\tilde{g}$  is a  $(v, \|\cdot\|)$ -bounded estimator of  $g$  if it satisfies (1) and (2), and that it is a  $(c, v, \|\cdot\|_{\Delta_\rho^\ell})$ -bounded estimator if it satisfies all three with local norm  $\|\cdot\|_y$  for all  $y \in \Delta_\rho^\ell$ .

Next, we present our algorithm and the bounded gradient estimators and regularizers corresponding to the two convex spaces (mentioned above) used in the algorithm.

## 5 ALGORITHM AND ESTIMATORS

---

### Algorithm 1: Fair State-Visitation

---

**Input:** Desired accuracy  $\varepsilon$

**Output:** An  $\varepsilon$ -approx policy  $\pi^\varepsilon$

**Parameter:**  $\eta^\lambda \leq \varepsilon/16$ ,  $\eta^x \leq \varepsilon/(8\ell(24M^2 + 1))$ ,  
 $T \geq \max\left(8 \log \ell/(\eta^x \varepsilon), 32M^2 n/(\eta^\lambda \varepsilon)\right)$

```

1 for  $t = 1, \dots, T$  do
2   Get  $\tilde{g}_t^\lambda$  as a  $((4M + 1)\ell, (24M^2 + 1)\ell, \|\cdot\|_{\Delta^\ell})$ -bounded
   estimator of  $g^\lambda(x_t, \lambda_t)$ 
3   Get  $\tilde{g}_t^x$  as a  $(2, \|\cdot\|_2)$ -bounded estimator of  $g^x(x_t, \lambda_t)$ 
4   Let  $x_{t+1} = \arg \min_{x \in \Delta_\rho^\ell} \langle \eta^x \tilde{g}_t^x, x \rangle + V_{x_t}(x)$ 
5   Let  $\lambda_{t+1} = \arg \min_{\lambda \in \mathbb{B}_{2M}^n} \langle \eta^\lambda \tilde{g}_t^\lambda, \lambda \rangle + V_{\lambda_t}(\lambda)$ 
6 end
7 Let  $(x^\varepsilon, \lambda^\varepsilon) = \frac{1}{T} \sum_{t=1}^T (x_t, \lambda_t)$ 
8 return  $\pi^\varepsilon$  with  $\pi^\varepsilon(a|s) = x_{s,a} / \sum_{a'=1}^m x_{s,a'}$ 

```

---

In Section 4, we formulated Fair-AMDP as a bilinear saddle point problem using the Lagrangian function of FAIR-LP. We also showed how the SMD framework can be used to compute an expected approximate solution to this bilinear problem. Given input  $\varepsilon$ , Algorithm 1 computes  $(x^\varepsilon, \lambda^\varepsilon)$ , which is an expected  $\varepsilon$ -approximate solution to the optimization problem in Equation 6, i.e.  $\mathbb{E}[\text{Gap}(x^\varepsilon, \lambda^\varepsilon)] \leq \varepsilon$ . The details of the regularizers and estimators are provided in the next paragraph. Then, in Line 8, the algorithm uses  $x^\varepsilon$  to compute  $\pi^\varepsilon$ . Note that it is not immediately clear that  $\pi^\varepsilon$  satisfies the approximation guarantees (in expectation) as mentioned in Definition 8 even though  $x^\varepsilon$  is  $\varepsilon$ -approximate solution for the corresponding bilinear saddle point problem. In addition, note that Algorithm 1 is oblivious to the presence of the fair-action.

We use normalized entropic regularizer for  $\Delta_\rho^\ell$ , and  $\|\cdot\|_2^2$  regularizer for  $\mathbb{B}_{2M}^\ell$ , as shown in Table 1. The gradients of  $f$  with respect to  $x$ , and  $\lambda$  are given as

$$g^x = (\Gamma - \hat{\Gamma})\lambda - r,$$

$$g^\lambda = (\hat{\Gamma} - \Gamma)^T x.$$

Note that since we maximize with respect to  $x$ , we take the negative of the gradient with respect to  $x$ . To estimate  $\tilde{g}_t^x$  and  $\tilde{g}_t^\lambda$  of  $g^x$  and  $g^\lambda$ , we use the generative model as follows, where  $(s, a) \sim x_t$  denotes the state-action pair sampled from the distribution  $x_t \in \Delta_\rho^\ell$ .

- (1) For an  $x \in \Delta_\rho^\ell$ , the estimator  $\tilde{g}_t^\lambda$  for  $g^\lambda$  in Line 2 is computed:

$$(s, a) \sim x, \quad s' \sim \Gamma((s, a), s')$$

$$\tilde{g}_t^\lambda(x, \lambda) = e_s - e_{s'},$$

where  $e_s$  is the unit vector in  $\mathbb{R}^n$ . Finally,  $\tilde{g}_t^\lambda = \tilde{g}_t^\lambda(x_{t-1}, \lambda_{t-1})$ .

- (2) The estimator  $\tilde{g}_t^x$  of  $g^x$  in Line 3 is computed as:

$$(s, a) \sim [1/\ell], \quad s' \sim \Gamma((s, a), s')$$

$$\tilde{g}_t^x(x, \lambda) = \ell(\lambda_{s'} - \lambda_s - r_{s,a})e_{s,a},$$

where  $\lambda_s$  denotes the  $s$ -th entry of  $\lambda$ ,  $e_{s,a}$  is a unit vector in  $\mathbb{R}^\ell$ , and  $\tilde{g}_t^x = \tilde{g}_t^x(x_{t-1}, \lambda_{t-1})$ .

Next, we show that  $\tilde{g}_t^x$  and  $\tilde{g}_t^\lambda$  are bounded (Definition 7).

**Lemma 3.** The estimator  $\tilde{g}^x(x, \lambda)$  as constructed above is a  $((4M + 1)\ell, (24M^2 + 1)\ell, \|\cdot\|_{\Delta^\ell})$ -bounded estimator.

PROOF. From the definition of  $\tilde{g}^x(x, \lambda)$  it follows that

$$\mathbb{E}[\tilde{g}^x] = \sum_{s, s' \in [n], a \in [m]} \Gamma((s, a), s) (\lambda_{s'} - \lambda_s - r_{s,a}) e_{s,a}$$

$$= (\Gamma - \hat{\Gamma})\lambda - r.$$

Since  $\|\lambda\|_\infty \leq 2M$  and  $r \in [0, 1]^\ell$ ,  $|\lambda_{s'} - \lambda_s - r_{s,a}| \leq 4M + 1$ , we have  $\|\tilde{g}_t^x\|_\infty \leq (4M + 1)\ell$ . For the second moment, we have that for any  $x' \in \Delta^\ell$ ,

$$\mathbb{E}[\|\tilde{g}^x\|_{x'}^2] \leq \sum_{s,a} \frac{1}{\ell} x'_{s,a} (24M^2 + 1)\ell^2 = (24M^2 + 1)\ell.$$

□

**Lemma 4.** The estimator  $\tilde{g}^\lambda(x, \lambda)$  as constructed above is a  $(2, \|\cdot\|_2)$  bounded estimator.

The proof of Lemma 4 is in Appendix A. We note that if  $\rho = 0$ , then our algorithm is the same as that by Jin and Sidford [18] for the unconstrained AMDP. In Section 6, Theorem 1 shows that  $(x^\varepsilon, \lambda^\varepsilon)$  is such that  $\mathbb{E}\text{Gap}(x^\varepsilon, \lambda^\varepsilon) \leq \varepsilon$ . Then, Theorem 2 shows that the policy  $\pi^\varepsilon$  constructed at Line 8 is indeed  $(3\varepsilon, \varepsilon)$ -approximate in expectation, and thus is the main contribution of our work.

	$R$	$V_{x'}(x)$
$x$ - space	$\sum_{s,a} x_{s,a} \log(x_{s,a})$	$\sum_{s,a} x'_{s,a} \log(x'_{s,a}/x_{s,a})$
$\lambda$ - space	$\frac{1}{2}\ \lambda\ _2^2$	$\frac{1}{2}\ \lambda - \lambda'\ _2^2$

Table 1: SMD parameter setting

## 6 THEORETICAL RESULTS

Theorem 1 shows that  $(\mathbf{x}^\epsilon, \lambda^\epsilon)$  computed by Algorithm 1 at Line 5 satisfies  $\mathbb{E}[\text{Gap}(\mathbf{x}^\epsilon, \lambda^\epsilon)] \leq \epsilon$ .

**Theorem 1.** Given a state-visitation fairness vector  $\rho \in [0, 1/n]^n$ , desired accuracy  $\epsilon > 0$ , and bounded estimators  $\tilde{g}_t^{\mathbf{x}}$  and  $\tilde{g}_t^\lambda$  as given in Lemmas 3 and 4, Algorithm 1 with step sizes  $\eta^{\mathbf{x}} \leq \frac{\epsilon}{8\ell(24M^2+1)}$  and  $\eta^\lambda \leq \epsilon/16$ , at the end of  $T \geq \max\left(\frac{8\log \ell}{\eta^{\mathbf{x}}\epsilon}, \frac{32M^2n}{\eta^\lambda\epsilon}\right)$  rounds, at Line 5 computes  $(\mathbf{x}^\epsilon, \lambda^\epsilon)$  such that  $\mathbb{E}[\text{Gap}(\mathbf{x}^\epsilon, \lambda^\epsilon)] \leq \epsilon$ .

The expectation is over the stochasticity in both the state-action selection and the MDP transitions. The proof of Theorem 1 is in Appendix B. We make two remarks. First, the step size  $\eta^{\mathbf{x}}$  and time horizon  $T$  depend on the mixing time of an MDP. The problem of estimating (or providing an upper bound) on the mixing time of an MDP is an active research topic [28, 36]. We rely on these techniques to estimate the mixing time used to compute step size and time horizon.

Second, the lower bound on  $T$  is proportional to  $M$ , which becomes very large as  $\rho_s$  approaches  $1/n$ . Hence, if  $\rho_s$ 's are close to  $1/n$  then the search space of  $\lambda$  in Algorithm 1 increases and it converges slowly. Note that, if  $\rho_s \leq 1/2n$  then  $d_\rho \leq \max_s \frac{1}{\rho_s}$  then our algorithm yields similar convergence time (in order) as that of the constrained AMDP algorithm by [18] that only computes a feasible solution without reward maximization. This bound, without Assumption 2, can be obtained from a stationary distribution of (any) strictly feasible policy.

Next, we define the notion of simultaneous approximation guarantee of an solution to the Fair-AMDP problem and show that the the policy  $\pi^\epsilon$  computed by Algorithm 1 satisfies this notion in Theorem 2.

**Definition 8.** Let  $\mathbf{x}^*$  be the optimal solution to FAIR-LP (D). A policy  $\pi$  is called  $(\epsilon_1, \epsilon_2)$ -approximate for the Fair-AMDP problem if  $\mathbf{r}^T(\Pi \mathbf{v}^\pi) \geq \mathbf{r}^T \mathbf{x}^* - \epsilon_1$  and  $\mathbf{C}\mathbf{x} \geq \mathbf{1} - \epsilon_2$ .

**Theorem 2.** Given a state-visitation fairness vector  $\rho \in [0, 1/n]^n$ , desired accuracy  $\epsilon > 0$  and bounded estimators  $\tilde{g}_t^{\mathbf{x}}$  and  $\tilde{g}_t^\lambda$  as given in Lemmas 3 and 4, Algorithm 1 with step sizes and  $T$  as in Theorem 1 computes a policy  $\pi^\epsilon$  which is  $(3\epsilon, \epsilon)$ -approximate in expectation with sample complexity  $O(nm\epsilon^{-2}t_{mix}^2(1+d_\rho)^2 \log(nm))$ .

The proof of Theorem 2 is given in Appendix C. As stated earlier, Algorithm 1 is oblivious to the presence of fair-action. Assumption 2 is not necessary if strict feasibility (even for  $\rho \in [0, 1]^n$  such that  $\sum_s \rho_s < 1$ ) is known *a priori*. The convergence time  $T$  in this case would depend on the reward and the stationary distribution induced by the strictly feasible policy. We note an additional multiplicative factor of  $(1+d_\rho)^2$  in the sample complexity of our proposed algorithm when compared with the best known sample complexity result for the unconstrained problem (see Theorem 1 by Jin and Sidford [18]). In the proposed framework, this factor is required to ensure that the bounding box contains the optimal value of the primal variable  $\lambda$  and is the price we pay to compute a policy with simultaneous approximation guarantee on both fairness and reward. However, the sample complexity is not tight when  $\rho = \mathbf{0}$ . In this case  $(1+d_\rho)^2 = n^2$  and the sample complexity of the proposed algorithm becomes  $O(n^3 m \epsilon^{-2} t_{mix}^2 \log(nm))$  which is suboptimal.

Finding a better sample complexity for fair-AMDP problem is an interesting future work.

## 7 SIMULATIONS

In this section, we evaluate our proposed algorithm and validate its properties with experiments on simulated data. Figure 1 presents the MDP transition dynamics specification with three states,  $s_0, s_1, s_2$ , and two actions,  $a_0, a_1$ . The reward of taking  $a_0$  on  $s_0$  is 1, and 0.1 otherwise. It is easy to see that the optimal unconstrained policy is a deterministic policy with  $\pi(s_0) = a_0$ ,  $\pi(s_1) = a_1$ , and  $\pi(s_2) = a_0$ . This policy yields an average-reward of 0.526, with the three states  $s_0, s_1$ , and  $s_2$  being visited 47.4%, 43.5%, and 9.1% of times respectively. Also, since the MDP is feasible for  $\rho \leq 1/3$ , we do not use the fair-action. Recall,  $\mathbf{x}^\epsilon$  is the average of  $\mathbf{x}_t$ ,  $t \in [T]$ . Further, we use  $\mathbf{x}$  to denote the state-action frequency vector of policy  $\pi^\epsilon$  derived from  $\mathbf{x}^\epsilon$  (Step 8, Algorithm 1). Note that the reward of  $\pi^\epsilon$  is  $\mathbf{r}^T \mathbf{x}$ .

**Convergence of SMD:** We first empirically demonstrate the convergence of SMD. We choose a fairness constraint  $\rho = [0.1, 0.1, 0.25]$ , which is violated by the unconstrained optimal policy. We implemented Algorithm 1 to solve this problem with  $M = 100$  and  $\eta^{\mathbf{x}} = \eta^\lambda = 0.01$ . Figure 1 shows the average-reward and state-visitation frequencies for policies obtained after different number of gradient descent steps, up to a maximum of 20,000. The results are averaged over 100 runs, with the shaded regions representing the standard deviations. We also see that the reward  $\mathbf{r}^T \mathbf{x}$ , approaches the fair optimal value and the visitation frequency of  $s_2$  approaches the required value (25%). We note that  $\mathbf{r}^T \mathbf{x}$  can cross the optimal value as  $\mathbf{x}$  is only approximately fair. In Figure 1, we also plot  $\mathbf{r}^T \mathbf{x}^\epsilon$  along with  $\mathbf{r}^T \mathbf{x}$  for different values of  $T$ .

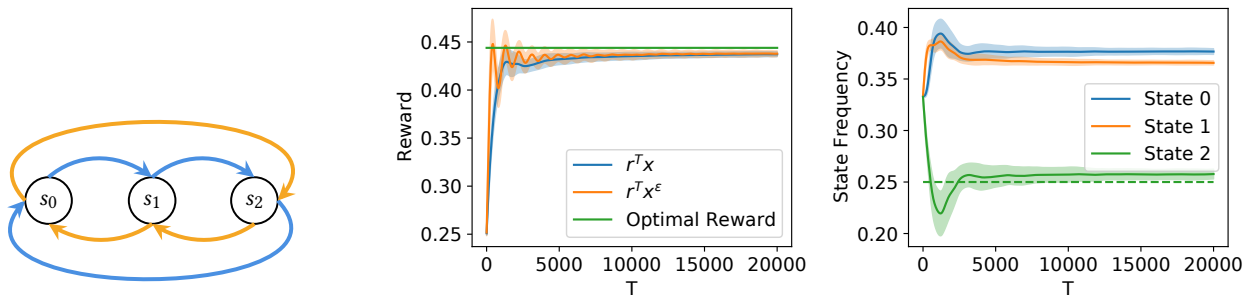
**Gap Function:** The theorems prove that the difference between  $\mathbf{r}^T \mathbf{x}$  and the optimal reward decreases with the value of the the gap function  $\text{Gap}(\mathbf{x}^\epsilon, \lambda^\epsilon)$ . Figure 2 (left) plots the reward difference as the gap decreases.

**Varying the Constraint  $\rho$ :** In this section, we vary the value of  $\rho_2 \in \{0.1, 0.15, 0.2, 0.25, 0.3\}$ . For each  $\rho$  parameter, we run SMD 10 times, with 10,000 steps each. Figure 2 plots the reward and state-visitation frequency on  $s_2$  for different  $\rho$ . The colors blue, orange, green, red, and purple indicate increasing values of  $\rho_2$ . As expected, all fairness constraints are satisfied, but as  $\rho_2$  increases, the optimal rewards decreases. This decrease captures the cost of fairness as the fairness constraints become more stringent.

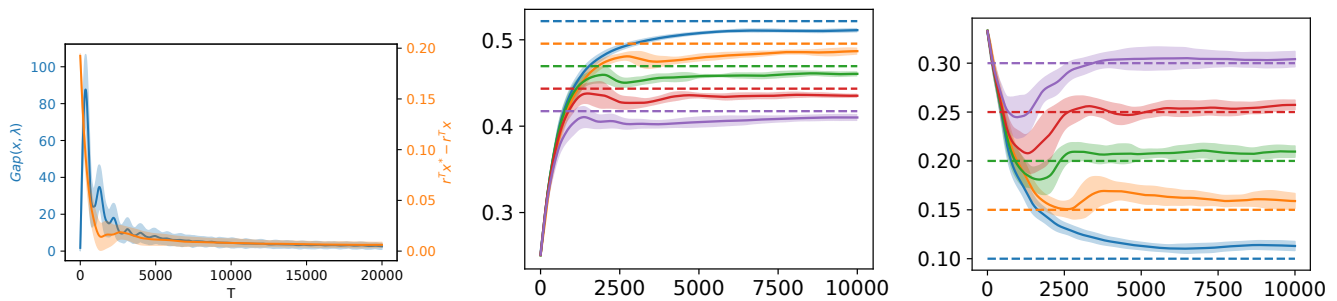
## 8 CONCLUSION AND FUTURE WORK

In this paper, we studied the Fair-AMDP problem and proposed an SMD-based algorithm to compute a policy with simultaneous approximation guarantee on average-reward and state-visitation fairness. We note that our results in this work can be extended to ensure state-action visitation fairness: for  $\rho \in [0, 1]^\ell$  a fair policy  $\pi$  with stationary distribution  $\mathbf{v}^\pi$  satisfies  $\Pi \cdot \mathbf{v}^\pi \geq \rho$ , provided there is a strictly feasible solution.

In Section 4.2, we determined the domain for the primal and dual variables for approximately solving the bilinear saddle point problem using the SMD framework. In particular, we restricted our domain of  $\mathbf{x}$  to  $\Delta_\rho^\ell$  and we removed  $\mu$  variables from the optimization problem. This in turn demands that at step 4 in Algorithm 1,  $\mathbf{x}_t$  is computed by projecting to  $\Delta_\rho^\ell$ . This approach can also be adapted



**Figure 1:** Left: the MDP specification, with three states and two actions. In each state, action  $a_0$  takes the blue transition with probability 0.9, and takes the yellow transition with probability 0.1; action  $a_1$  has the converse effect. Middle and right: reward and state-visitation frequency of a policy learned after  $T$  steps, averaged over 100 runs.



**Figure 2:** Left: gap function value (blue) and reward difference (orange) learned after  $T$  steps, averaged over 100 runs. Middle and right: reward and visitation frequency of  $s_2$  of policies learned after  $T$  steps, averaged over 10 runs, for each  $\rho_2 \in \{0.1, 0.15, 0.2, 0.25, 0.3\}$ . The colors of blue, orange, green, red, and purple indicate increasing values of  $\rho_2$ . Dashed lines indicate optimal rewards and visitation frequency constraints.

for computing a feasible solution for an AMDP with arbitrary linear constraints and no objective function [18].

For future work, the time complexity bound of our algorithm becomes large as  $\rho_s \rightarrow 1/n$ . Thus, an immediate direction is to improve the dependence of time complexity on  $\rho$ , either via a new algorithm or better analysis. Also observe that the introduction of fair-action ensures strict feasibility when  $\rho_s < 1/n$  for all  $s$ . It is interesting to see if there is an approach that ensures feasibility for a broader class of  $\rho$ .

We note that a convex program formulation for discounted-reward MDPs with state-visitation fairness constraints is not immediate, and we need novel techniques to solve the state-visitation fairness problem for discounted-reward MDPs. The introduction of fairness constraints may lead to a loss in the maximum average-reward that can be achieved. The difference in the average-reward of an unconstrained optimal policy and that of an optimal policy that satisfies fairness captures the *price of fairness* in the system. Further study of price of fairness in the Fair-AMDP setting is an interesting future direction.

## ACKNOWLEDGMENTS

We thank the anonymous reviewers for their reviews. Vineet Nair is thankful to be supported by the European Union’s Horizon 2020

research and innovation program under grant agreement No 682203-ERC-[Inf-Speed-Tradeoff]. Vishakha Patil is grateful for the support of a Google PhD Fellowship.

## REFERENCES

- [1] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. 2017. Constrained Policy Optimization. In *International Conference on Machine Learning (ICML)*. PMLR, 22–31.
- [2] Muhammad Ali, Piotr Sapiezynski, Miranda Bogen, Aleksandra Korolova, Alan Mislove, and Aaron Rieke. 2019. Discrimination Through Optimization: How Facebook’s Ad Delivery Can Lead to Biased Outcomes. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 1–30.
- [3] Eitan Altman. 1999. *Constrained Markov Decision Processes*. Vol. 7. CRC Press.
- [4] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2019. *Fairness and Machine Learning*. fairmlbook.org. <http://www.fairmlbook.org>.
- [5] James A Berkovec, Glenn B Canner, Stuart A Gabriel, and Timothy H Hannan. 1996. Mortgage Discrimination and FHA Loan Performance. *Cityscape* (1996), 9–24.
- [6] Ronen I Brafman and Moshe Tenenbholz. 2002. R-max – A General Polynomial Time Algorithm for Near-Optimal Reinforcement Learning. *Journal of Machine Learning Research (JMLR)* 3, Oct (2002), 213–231.
- [7] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. 2009. Building Classifiers with Independence Constraints. In *IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 13–18.
- [8] Yair Carmon, Yujia Jin, Aaron Sidford, and Kevin Tian. 2019. Variance reduction for matrix games. In *Advances in Neural Information Processing Systems (NeurIPS)*. 11381–11392.
- [9] L Elisa Celis, Sayash Kapoor, Farnood Salehi, and Nisheeth Vishnoi. 2019. Controlling Polarization in Personalization: An Algorithmic Framework. In *Conference on Fairness, Accountability, and Transparency (FAT\*)*. 160–169.



- [10] Elliot Creager, David Madras, Toniann Pitassi, and Richard Zemel. 2020. Causal Modeling for Fairness in Dynamical Systems. In *International Conference on Machine Learning (ICML)*. PMLR, 2185–2195.
- [11] Alexander D’Amour, Hansa Srinivasan, James Atwood, Pallavi Baljekar, D Sculley, and Yoni Halpern. 2020. Fairness is Not Static: Deeper Understanding of Long Term Fairness via Simulation Studies. In *Conference on Fairness, Accountability, and Transparency (FAT\*)*. 525–534.
- [12] Shayan Doroudi, Philip S Thomas, and Emma Brunskill. 2018. Importance Sampling for Fair Policy Selection. In *International Joint Conference on Artificial Intelligence (IJCAI)*. 5239–5243.
- [13] Julia Dressel and Hany Farid. 2018. The Accuracy, Fairness, and Limits of Predicting Recidivism. *Science Advances* 4, 1 (2018), eaao5580.
- [14] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness Through Awareness. In *Innovations in Theoretical Computer Science Conference (ITCS)*. 214–226.
- [15] Peter Geibel and Fritz Wysotzki. 2005. Risk-Sensitive Reinforcement Learning Applied to Control Under Constraints. *Journal of Artificial Intelligence Research (JAIR)* 24 (2005), 81–108.
- [16] Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. 2019. Provably Efficient Maximum Entropy Exploration. In *International Conference on Machine Learning (ICML)*. PMLR, 2681–2691.
- [17] Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. 2017. Fairness in Reinforcement Learning. In *International Conference on Machine Learning (ICML)*. PMLR, 1617–1626.
- [18] Yujia Jin and Aaron Sidford. 2020. Efficiently Solving MDPs with Stochastic Mirror Descent. In *International Conference on Machine Learning (ICML)*. PMLR, 4890–4900.
- [19] Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. 2016. Fairness in Learning: Classic and Contextual Bandits. *Advances in Neural Information Processing Systems (NIPS)* 29 (2016), 325–333.
- [20] Michael Kearns and Satinder Singh. 2002. Near-Optimal Reinforcement Learning in Polynomial Time. *Machine learning* 49, 2 (2002), 209–232.
- [21] Michael J Kearns, Yishay Mansour, and Andrew Y Ng. [n.d.]. Approximate Planning in Large POMDPs via Reusable Trajectories. In *Advances in Neural Information Processing Systems (NIPS)*.
- [22] Jon M. Kleinberg, Sendhil Mullainathan, and Manish Raghavan. 2017. Inherent Trade-Offs in the Fair Determination of Risk Scores. In *Innovations in Theoretical Computer Science Conference (ITCS)*, Vol. 67. 43:1–43:23.
- [23] Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. 2019. Efficient Exploration via State Marginal Matching. *arXiv preprint arXiv:1906.05274* (2019).
- [24] Fengjiao Li, Jia Liu, and Bo Ji. 2019. Combinatorial sleeping bandits with fairness constraints. *IEEE Transactions on Network Science and Engineering (T-NSE)* 7, 3 (2019), 1799–1813.
- [25] Sridhar Mahadevan. 1996. Average Reward Reinforcement Learning: Foundations, Algorithms, and Empirical Results. *Machine Learning* 22, 1-3 (1996), 159–195.
- [26] Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. 2009. Robust Stochastic Approximation Approach to Stochastic Programming. *SIAM Journal on Optimization* 19, 4 (2009), 1574–1609.
- [27] Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y Narahari. 2020. Achieving Fairness in the Stochastic Multi-Armed Bandit Problem. In *AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 34. 5379–5386.
- [28] Daniel Paulin. 2015. Concentration Inequalities for Markov Chains by Marton Couplings and Spectral Methods. *Electronic Journal of Probability* 20 (2015), 1–32.
- [29] Martin L Puterman. 2014. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons.
- [30] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement Learning: An Introduction*. MIT Press.
- [31] Aviv Tamar, Dotan Di Castro, and Shie Mannor. 2012. Policy Gradients with Variance Related Risk Criteria. In *International Conference on Machine Learning (ICML)*. 1651–1658.
- [32] Mengdi Wang. 2017. Primal-Dual  $\pi$  Learning: Sample Complexity and Sublinear Run Time for Ergodic Markov Decision Problems. *arXiv preprint arXiv:1710.06100* (2017).
- [33] Min Wen, Osbert Bastani, and Ufuk Topcu. 2021. Algorithms for Fairness in Sequential Decision Making. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 1144–1152.
- [34] DJ White. 1973. An Example of Loosely Coupled Stages in Dynamic Programming. *Management Science* 19, 7 (1973), 739–746.
- [35] Sirui Yao and Bert Huang. 2017. Beyond parity: Fairness objectives for collaborative filtering. In *Advances in Neural Information Processing Systems (NIPS)*, Vol. 30. 2921–2930.
- [36] Tom Zahavy, Alon Cohen, Haim Kaplan, and Yishay Mansour. 2020. Unknown Mixing Times in Apprenticeship and Reinforcement Learning. In *Conference on Uncertainty in Artificial Intelligence (UAI)*. PMLR, 430–439.
- [37] Xueru Zhang and Mingyan Liu. 2021. Fairness in Learning-Based Sequential Decision Algorithms: A Survey. In *Handbook of Reinforcement Learning and Control*, Springer, 525–555.
- [38] Xueru Zhang, Ruibo Tu, Yang Liu, Mingyan Liu, Hedvig Kjellstrom, Kun Zhang, and Cheng Zhang. 2020. How do Fair Decisions Fare in Long-Term Qualification?. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 33. 18457–18469.