

# Towards Return Parity in Markov Decision Processes

Jianfeng Chi<sup>1</sup>, Jian Shen<sup>2</sup>, Xinyi Dai<sup>2</sup>, Weinan Zhang<sup>2</sup>, Yuan Tian<sup>1</sup>, Han Zhao<sup>3</sup>

<sup>1</sup>University of Virginia

<sup>2</sup>Shanghai Jiao Tong University

<sup>3</sup>University of Illinois at Urbana-Champaign

## Abstract

Algorithmic decisions made by machine learning models in high-stakes domains may have lasting impacts over time. Unfortunately, naive applications of standard fairness criterion in static settings over temporal domains may lead to delayed and adverse effects. To understand the dynamics of performance disparity, we study a fairness problem in Markov decision processes (MDPs). Specifically, we propose return parity, a fairness notion that requires MDPs from different demographic groups that share the same state and action spaces to achieve approximately the same expected time-discounted rewards. We first provide a decomposition theorem for return disparity, which decomposes the return disparity of any two MDPs into the distance between group-wise reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions induced by the group policies. Motivated by our decomposition theorem, we propose algorithms to mitigate return disparity via learning a shared group policy with state visitation distributional alignment using integral probability metrics. We conduct experiments to corroborate our results, showing that the proposed algorithm can successfully close the disparity gap while maintaining the performance of policies on two real-world recommender system benchmark datasets.

## 1 Introduction

The increasing use of automated decision-making systems trained with real-world data has raised serious concerns with potential unfairness caused by biased data, learning algorithms, and models. Decisions made by these systems have lasting and diverse effects on different social groups. For example, in predictive policing (Lum and Isaac, 2016), each time, the decisions about the locations of future crimes are made by the predictive models. As a result, the discovered crime rates in different communities might change dynamically and interactively as feedback to the decisions being made by the models. Thus, the error rates of the predictive models for different communities might change over time, and error gaps among communities could possibly exacerbate in the long run. Similar feedback loops could also exist in the applications such as recommender systems (*e.g.*, the user satisfactions from different demographic groups diverge over time), temporal resource allocation systems (*e.g.*, the uneven resource allocation become more skew over one group than the others), etc. This interplay between algorithmic decisions and the heterogeneous reactions caused by the decisions further complicates the analysis of (un)fairness problems in the dynamic environment.

Most prior works mainly focus on studying static fairness notions (*e.g.*, demographic parity (Dwork et al., 2012) and equalized odds (Hardt et al., 2016)) in the settings of classification (Hardt et al., 2016; Zafar et al.,

2017; Zhao and Gordon, 2019; Jiang et al., 2020) and regression (Komiyama et al., 2018; Agarwal et al., 2019; Chzhen et al., 2020; Chi et al., 2021). In a seminal work, Liu et al. (2018) show that somewhat contrary to the common belief, enforcing static fairness constraints could do harm to the minority group even in a one-step feedback model. Motivated by this observation, a line of work aims to extend static fairness notions in the settings of online learning (Blum et al., 2018; Bechavod et al., 2019) and multi-armed bandit (Joseph et al., 2016; Patil et al., 2020; Chen et al., 2020). However, these works do not take into account the interactions between the models and the environment: the decisions made by the models could potentially influence the state of our environment as well, as demonstrated by the predictive policing example. Other works study the interplay between (static) fairness notions and the population dynamics under simplified temporal dynamic models (Hu and Chen, 2018; Ensign et al., 2018; Hashimoto et al., 2018; Mouzannar et al., 2019; Zhang et al., 2019; Elzayn et al., 2019; Liu et al., 2020b). However, those simplified temporal dynamic models explicitly make task-specific assumptions on temporal dynamics and might not be able to precisely characterize the complex dynamics of a more general changing environment.

In this work, we study a fairness problem in Markov decision processes (MDPs) to understand the dynamics of performance disparity in the changing environments, taking into account the feedback loop caused by policies to the environments. Specifically, we propose *return parity*, a novel fairness notion that requires MDPs from different demographic groups that share the same state and action spaces to achieve approximately the same expected time-discounted rewards. To the best of our knowledge, our work is the first of this kind, in the sense that we study the long-term impact of policy functions in general MDPs. First, we formally show exact return parity cannot always be satisfied for any two MDPs and provide sufficient conditions that ensure return parity in terms of transitions, initial distributions, and the reward functions. Next, we derive a decomposition theorem for return disparity that decomposes it into the distance between group-wise reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions induced by the group policies. Motivated by the decomposition theorem, we propose algorithms to mitigate return disparity via learning a shared group policy with state visitation distributional alignment using integral probability metrics. We conduct experiments on two real-world benchmark datasets to corroborate the effectiveness of our proposed algorithms in reducing return disparity. Experimental results demonstrate that our proposed algorithms help to mitigate return disparity while maintaining the performance of policies.

## 2 Preliminaries

**Notation and Problem Setup** Throughout the paper, we mainly focus on discrete MDPs, where both the state and action spaces are finite.<sup>1</sup> A Markov decision process is a tuple  $(\mathcal{S}, \mathcal{A}, \mu, T, r, \gamma)$ , where  $\mathcal{S}$  and  $\mathcal{A}$  are the state space and the action space, respectively;  $\mu$  is initial state distribution;  $T \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{A}| \times |\mathcal{S}|}$  is the state transition function where  $T(s' | s, a) = \Pr[S_{t+1} = s' | S_t = s, A_t = a]$ ;  $r \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{A}|}$  is the reward function where  $r(s, a)$  represents the reward obtained when taking action  $a$  in state  $s$ . Throughout the paper, we assume that the reward function is uniformly bounded, *i.e.*,  $\exists R > 0, \|r\|_\infty \leq R$ . We use  $\gamma \in (0, 1)$  to denote the discount factor. Given a policy  $\pi \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{A}|}$  (*i.e.*,  $\pi(a | s) = \Pr[A_t = a | S_t = s]$ ), the induced state transition under  $\pi$  is  $P^\pi \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$  where  $P^\pi(s' | s) = \sum_{a \in \mathcal{A}} \pi(a | s) T(s' | s, a)$ . The induced distribution over states under the policy  $\pi$  at time step  $t$  is

$$\mu^{(\pi, t)} = \begin{cases} \mu & \text{if } t = 0 \\ P^\pi \mu^{(\pi, t-1)} & \text{otherwise.} \end{cases}$$

---

<sup>1</sup>The main results in Section 3 could be extended to continuous state space and action space.

The state visitation distribution (*i.e.*, time-discounted distribution over states) is  $\mu^\pi = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \mu^{(\pi,t)} = (1 - \gamma) \sum_{t=0}^{\infty} (\gamma P^\pi)^t \mu$  and the occupancy measure (*i.e.*, time-discounted distribution over state-action pairs) is  $\rho^\pi(s, a) = \mu^\pi(s) \pi(a | s)$ . We then define the value function w.r.t. the reward function  $r$  under the policy  $\pi$  as  $v^\pi(s) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r_t | S_0 = s]$  and the Q-function as  $q^\pi(s, a) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r_t | S_0 = s, A_0 = a]$ , where  $r_t$  is the immediate reward at time step  $t$ . With all the notation defined above, the goal of reinforcement learning is to find a policy to maximize the value (expected return) under the initial state distribution:

$$\eta^\pi := \mathbb{E}_{s \in \mu}[v^\pi(s)] = \frac{1}{1 - \gamma} \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r(s, a) \rho^\pi(s, a) = \frac{1}{1 - \gamma} \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r(s, a) \mu^\pi(s) \pi(a | s).$$

Let  $|\mathcal{S}| = m$  and  $|\mathcal{A}| = n$ . In practice, each state  $s \in \mathcal{S}$  might represent features of an individual and the action enforced on the individual could lead to the change of features of the individual. We also assume there are two Markov decision processes that represent two different demographic groups (*e.g.*, male/female, white/non-white) and the two MDPs share the same state space, action space, and discount factor but might differ in initial distributions, transitions, and reward functions. We use the subscript  $g \in \{0, 1\}$  to denote the two groups. For example,  $\mu_0$  and  $\mu_1$  are the initial distributions of the two groups, respectively. Note that in our paper, we mainly discuss the setting there are two different demographic groups, but the underlying theory and algorithms could easily be extended to the case with finite  $K > 2$  groups.

Next, we define  $\epsilon$ -return parity as a fairness criterion to ask that different demographic groups share approximately the same long-term rewards under a policy:

**Definition 2.1** ( $\epsilon$ -Return Parity). For  $g \in \{0, 1\}$ , two MDPs satisfy  $\epsilon$ -return parity if  $\Delta_{\text{Ret}} := |\mathbb{E}_{s \in \mu_0}[v^{\pi_0}(s)] - \mathbb{E}_{s \in \mu_1}[v^{\pi_1}(s)]| \leq \epsilon$ .

In different scenarios, return parity could have different implications: In predictive policing, return parity asks for approximately equalized prediction accuracy of models over time; In recommender systems, return parity might seek similar users' satisfaction across different demographic groups. The complex temporal dynamic of the above scenarios could be modeled by MDPs. The goal in our setting is then to find two policies that maximize the weighted combination of expected returns of two MDPs respectively while satisfying  $\epsilon$ -return parity:

$$\max_{\pi_0, \pi_1} \lambda \eta^{\pi_0} + (1 - \lambda) \eta^{\pi_1}, \quad \text{s.t.} \quad \Delta_{\text{Ret}} \leq \epsilon,$$

where  $\lambda \in [0, 1]$  represents the proportion of group 0 over the entire population.

**Integral Probability Metrics** The integral probability metrics (IPMs) are a class of distance measures on probability distributions over the same probability space (Müller, 1997). Formally, given two probability distributions  $\mathcal{P}$  and  $\mathcal{Q}$ , the IPMs are defined as  $d_{\mathcal{F}}(\mathcal{P}, \mathcal{Q}) = \sup_{f \in \mathcal{F}} |\int f d\mathcal{P} - \int f d\mathcal{Q}|$ , where  $\mathcal{F}$  is a class of real-valued bounded measurable functions on the space where the distributions are defined on. Different choices of  $\mathcal{F}$  recover different distance metrics: If we choose  $\mathcal{F} = \{f : \|f\|_L \leq 1\}$  where  $\|\cdot\|_L$  denotes the Lipschitz semi-norm, then  $d_{\mathcal{F}}(\mathcal{P}, \mathcal{Q})$  becomes *Wasserstein-1 distance*  $W_1(\mathcal{P}, \mathcal{Q})$ ; If we choose  $\mathcal{F} = \{f : \|f\|_{\mathcal{H}} \leq 1\}$  where  $\|\cdot\|_{\mathcal{H}}$  denotes the norm in a reproducing kernel Hilbert space (RKHS), then  $d_{\mathcal{F}}(\mathcal{P}, \mathcal{Q})$  becomes *maximum mean discrepancy*  $\text{MMD}(\mathcal{P}, \mathcal{Q})$ .

### 3 Analysis of Return Parity

In this section, we first show that return parity cannot always be satisfied between two MDPs that share the same state and action spaces and provide sufficient conditions under which return parity is possible. Then, we provide more insights into return disparity by proving an upper bound of the return gap between two MDPs. We defer all the detailed proofs to Appendix A due to the space limit.

#### 3.1 Is Return Parity Always Possible?

Before we provide a rigorous analysis of return disparity in MDPs, it is vital to ask whether the exact return parity is always achievable. The following proposition gives a negative answer to this question:

**Proposition 3.1.** For any constant  $c > 0$ , there exist two MDPs that share the same state and action spaces, such that  $\forall \pi_0, \pi_1 \in \Pi$ , the return disparity  $\Delta_{\text{Ret}} \geq c$ .

For example, consider two MDPs share two states  $s_1$  and  $s_2$ . Let  $r(s_1, a) = c(1 - \gamma) > 0$  and  $r(s_2, a) = 0$ ,  $\forall a \in \mathcal{A}$ ,  $T(s_2 | s_1, a) = T(s_1 | s_2, a) = 0$  and  $T(s_1 | s_1, a) = T(s_2 | s_2, a) = 1$ ,  $\forall a \in \mathcal{A}$ . Given  $\mu_0 = [1, 0]^T$  and  $\mu_1 = [0, 1]^T$ , then the return gap  $\Delta_{\text{Ret}} = c > 0$ . In this case, it is impossible to find policies to satisfy  $\epsilon$ -return parity for any  $\epsilon < c$ . Next we provide a sufficient condition to satisfy 0-return parity in specific conditions.

**Proposition 3.2.** Assume i) the reward functions and the initial distributions are the same across different groups, i.e.,  $r_0(s, a) = r_1(s, a)$  and  $\mu_0 = \mu_1$ ; ii) reward functions only depend on states, i.e.,  $r(s, a) = r(s)$ . Define  $\mathbf{d}_{ij} := [T_0(s_1 | a_i, s_j) - T_1(s_1 | a_i, s_j), \dots, T_0(s_m | a_i, s_j) - T_1(s_m | a_i, s_j)]^T$ . For  $\forall \mathbf{c} \in \mathbb{R}^m \setminus \mathbf{1}_m$ , if there exists  $i \in [n], j \in [m]$ , such that  $\langle \mathbf{c}, \mathbf{d}_{ij} \rangle \leq 0$ , then there exist  $\pi_0, \pi_1 \in \Pi$  such that 0-return parity is satisfied.

Proposition 3.2 states that exact return parity can be satisfied as long as  $\forall \mathbf{c} \in \mathbb{R}^m \setminus \mathbf{1}_m$ , such that the inner product between  $\mathbf{c}$  and one of the transition difference vectors  $\mathbf{d}_{ij}$  is nonpositive. Geometrically, it requires the directions of transition difference vectors  $\mathbf{d}_{ij}$  to be scattered in the hyperplane.

In addition, it is also natural to ask whether it is feasible to maximize the expected returns of the two MDPs simultaneously while achieving  $\epsilon$ -return parity in general. Formally, with the help of the linear programming (LP) approach for MDPs (De Farias, 2002; De Farias and Van Roy, 2003) and the duality of LP, it is equivalent to solve the following dual LP:

$$\begin{aligned} & \max \sum_s \sum_a \hat{\rho}_0(s, a) r_0(s, a) + \hat{\rho}_1(s, a) r_1(s, a) - \epsilon(b_0 + b_1) \\ & \text{s.t. } \sum_a \hat{\rho}_0(s_i, a) - \gamma \sum_s \sum_a T_0(s_i | s, a) \hat{\rho}_0(s, a) \\ & \quad = (\lambda + b_0 - b_1)(\mu_0)_i \quad \forall i \in [m] \\ & \quad \sum_a \hat{\rho}_1(s_i, a) - \gamma \sum_s \sum_a T_1(s_i | s, a) \hat{\rho}_1(s, a) \\ & \quad = (1 - \lambda + b_1 - b_0)(\mu_1)_i \quad \forall i \in [m], \end{aligned}$$

where  $\hat{\rho}_0(s, a), \hat{\rho}_1(s, a), b_0, b_1 \geq 0$ ,  $\forall s, a$  are dual variables. Note that  $\hat{\rho}_0(s, a)$  and  $\hat{\rho}_1(s, a)$  have the interpretation of discounted state-action counts of the policy when  $b_0 = b_1$ , and  $b_0, b_1$  are the corresponding dual variables of the  $\epsilon$ -return parity constraint, representing the ‘‘per unit cost’’ of the overall return to achieve  $\epsilon$ -return parity. The dual constraints are state transitions under the learned policies. We can now characterize

a sufficient condition for the optimal policies  $\pi_0$  and  $\pi_1$  to satisfy  $\epsilon$ -return parity with the dual formulation above.

**Proposition 3.3.** For  $\forall \hat{\rho}_0(s, a), \hat{\rho}_1(s, a), b_0, b_1 \geq 0$ , if there exists  $i \in [m]$ , such that

$$\begin{aligned} \sum_a \hat{\rho}_0(s_i, a) - \gamma \sum_s \sum_a T_0(s_i | s, a) \hat{\rho}_0(s, a) &> (b_0 - b_1)(\mu_0)_i \\ \sum_a \hat{\rho}_1(s_i, a) - \gamma \sum_s \sum_a T_1(s_i | s, a) \hat{\rho}_1(s, a) &> (b_1 - b_0)(\mu_1)_i \\ \sum_s \sum_a \hat{\rho}_0(s, a) r_0(s, a) + \hat{\rho}_1(s, a) r_1(s, a) &\leq (b_0 + b_1)\epsilon \end{aligned}$$

then the optimal policies  $\pi_0^*$  and  $\pi_1^*$  that maximize the expected returns of two MDPs satisfy  $\epsilon$ -return parity.

In light of the dual LP formulation, the first two inequalities in Proposition 3.3 indicate the probability masses of the initial distributions in at least one state are greater than zero, and the last inequality requires the maximum value of the objective function in the dual LP formulation is no greater than zero.

### 3.2 A Decomposition Theorem for Return Disparity

The linear programming methods to solve the return disparity problem of MDPs become impractical in continuous or high-dimensional discrete state and action spaces. However, in many real-world scenarios where return parity is desired, the number of the states or actions is often large (*e.g.*, recommend items to different demographic groups of users). In this section, we shall provide an upper bound to (1) quantitatively characterize return disparity in terms of the distance between the reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions and, (2) motivate our algorithm design to mitigate return disparity in continuous or high-dimensional discrete state and action spaces (Sec. 4).

**Theorem 3.1.** For  $g \in \{0, 1\}$ , given policies  $\pi_0, \pi_1 \in \Pi$  and assume there exists a witness function class  $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}\}$ , such that the reward functions  $r_g(s) = \mathbb{E}_{a \sim \pi_g(\cdot | s)}[r_g(a, s) | s] \in \mathcal{F}$  for  $\forall s \in \mathcal{S}$ ,  $a \in \mathcal{A}$  and  $g \in \{0, 1\}$ , then the following holds:

$$\Delta_{\text{Ret}} \leq \frac{1}{1 - \gamma} \left( \|r_0 - r_1\|_\infty + R \cdot \min \{ \mathbb{E}_{s \sim \mu^{\pi_0}} [d_{\pi_0, \pi_1}(s)], \mathbb{E}_{s \sim \mu^{\pi_1}} [d_{\pi_0, \pi_1}(s)] \} + d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s)) \right),$$

where  $d_{\pi_0, \pi_1}(s) := \|\pi_0(\cdot | s) - \pi_1(\cdot | s)\|_1$ .

**Remark** We see that return disparity is upper bounded by three terms: the distance between group-wise reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions of the two MDPs. Given any two MDPs, the distance between group-wise reward functions is constant. If we further assume the two MDPs share the same reward functions (*i.e.*,  $r_0 = r_1$ ) and the same policy (*i.e.*,  $\pi_0 = \pi_1$ ), then the upper bound in Theorem 3.1 is simplified as

$$\Delta_{\text{Ret}} \leq d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s)).$$

In this case, it implies a sufficient condition to minimize return disparity is to find policies  $\pi \in \Pi$  that minimize the distance between induced state visitation distributions in the two MDPs. In the subsequent sections, we assume the reward functions of different groups are (approximately) the same and propose algorithmic interventions to mitigate return disparity. For completeness, we also provide another decomposition theorem

(Theorem C.1) in Appendix C and motivate the design of another family of algorithms based on Theorem C.1 in Appendix D. We present the Theorem 3.1 in the main text since the algorithms (see Sec. 4) motivated by it are more efficient and stable in the application scenarios (*e.g.*, recommender system) we consider.

## 4 Mitigation of Return Disparity

Inspired by the theoretical results in Theorem 3.1, we introduce a state visitation distribution alignment procedure, which can be naturally incorporated into existing RL methods, to encourage policies to maximize expected returns while keeping state visitation distribution similar to each other. We use deep Q-networks (Mnih et al., 2015) as our baseline backbone algorithm, which has demonstrated its superior performance in recommender application (Zheng et al., 2018), in which we will conduct experiments next. In what follows, we first briefly introduce how to learn the deep Q-networks and then discuss state visitation distributional alignment via IPMs. We give the main pipeline of our algorithms in Algorithm 1.

### 4.1 Preliminaries: Learning Deep Q-networks

The main idea behind learning deep Q-networks (Q-learning) is to approximate the value functions in high dimensional state and action spaces. Specifically, we aim at learning deep Q-network  $Q(s, a; \theta) : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$  to approximate the reward when taking an action in a given state. Given the deep Q-network, we can construct the policy that maximizes the rewards:  $\pi(s) = \arg \max_a Q(s, a)$ . When the action space is discrete such as recommendation applications, the Q-network is often implemented as  $Q(s, a; \theta) : \mathcal{S} \rightarrow \mathbb{R}^{|\mathcal{A}|}$  for efficiency, where the value in output dimension  $i$  represents estimated reward when taking action  $a_i$  given the state  $s$ .

During model training, the parameter  $\theta$  of the Q-network is trained through a trial-and-error process. Take interactive recommendation process as an example: at each time step  $t$ , the recommender agent obtains a user’s state  $s_t$ , and takes an action  $a_t$  (*e.g.*, recommend an item) via the  $\epsilon$ -greedy policy (*i.e.*, with probability  $1 - \epsilon$  taking an action with the max Q-value, and with probability  $\epsilon$  choosing a random action). The agent then receives the reward  $r_t$  (*e.g.*, rating score on the recommended item) and the updated state  $s_{t+1}$  from the user’s feedback and stores the experience  $(s_t, a_t, r_t, s_{t+1})$  in replay buffer  $\mathcal{D}$ . After updating the replay buffer with batches of experiences from different users, the agent then optimizes the following loss function to improve the Q-network:

$$\mathcal{L}_Q(\theta) = \mathbb{E}_{(s_t, a_t, r_t, s_{t+1}) \sim \mathcal{D}} [(y_t - Q(s, a; \theta))^2], \quad (1)$$

with

$$y_t = r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta). \quad (2)$$

To avoid the overestimation problem (Thrun and Schwartz, 1993) in original DQN, we adopt the double DQN architecture (Van Hasselt et al., 2016): a target network  $Q'$  parameterized by  $\theta'$  is utilized along with the online network  $Q$ . The online network is updated at each model update step, and the target network is a duplicate of the online network and updated with delay (soft update):

$$\theta' = \tau \theta + (1 - \tau) \theta', \quad (3)$$

where  $\tau$  controls the update frequency. With the double DQN architecture,  $y_t$  in Eq. 2 is changed to

$$y_t = r_t + \gamma \max_{a_{t+1}} Q'(s_{t+1}, \arg \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta); \theta'). \quad (4)$$

---

**Algorithm 1** Algorithm to mitigate return disparity under the framework of DQN.

---

```
1: Initialize Q-functions  $Q_\theta$ , feature extractors  $f_{\psi_0}$  and  $f_{\psi_1}$ , environment buffers  $D_0$  and  $D_1$ 
2: for each iteration do
3:   for each environment step do
4:     for  $g \in \{0, 1\}$  do
5:       Sample an action  $a_g$  using  $\epsilon$ -greedy policy; Add the experience  $(s_g, a_g, s'_g, r_g)$  to  $D_g$ ;
6:     end for
7:   end for
8:   for each model update step do
9:     Sample a mini-batch of experiences from  $D_0$  and  $D_1$ ; Update online and target DQNs and feature
       extractors  $f_{\psi_g}$ ,  $g \in \{0, 1\}$  using Eq. 1 and Eq. 3;
10:  end for
11:  for each state visitation distributional alignment step do
12:    Update feature extractors  $f_{\psi_0}$  and  $f_{\psi_1}$  using Eq. 7 or Eq. 8;
13:  end for
14: end for
```

---

## 4.2 State Visitation distributional Alignment

Inspired by Theorem 3.1, we propose to align the state visitation distribution via learning group-invariant representation. Specifically, we introduce two feature extractors  $f_{\psi_g}$ ,  $g \in \{0, 1\}$  before inputting  $s_t$  to Q-function. We alternatively optimize  $f_{\psi_g}$  between minimizing the loss of Q-function (Eq. 1) and minimizing the loss of state visitation distributional alignment. At each iteration, the feature extractors and the Q-function, parameterized by  $\theta' = \{\psi_g \circ \theta\}$ , are first trained jointly to minimize  $\mathcal{L}_q(\theta')$ . After updating the models, the feature extractors are then updated to minimize the loss of state visitation distributional alignment via minimizing the estimated integral probability metrics between different groups. The whole algorithm is shown in Algorithm 1. Next, we introduce the detailed steps of state visitation distributional alignment via Wasserstein-1 distance, which is a kind of IPM choosing  $\mathcal{F} = \{f : \|f\|_L \leq 1\}$  and is more favorable than total variation distance from the optimization perspective (Arjovsky et al., 2017).

**Wasserstein-1 Distance** Given samples  $s_0$  from the state visitation distribution of group 0 and samples  $s_1$  from the state visitation distribution of group 1 respectively, two feature extractors  $f_0$  and  $f_1$ , with parameters  $\psi_0$  and  $\psi_1$  respectively, map the samples to feature representations:  $h_0 = f_0(s_0)$  and  $h_1 = f_1(s_1)$ . Upon receiving the feature representations from both groups, we use a critic function  $f_c$ , parameterized by  $\omega$ , to estimate Wasserstein-1 distance (Arjovsky et al., 2017) by maximizing the following objective function:

$$\mathcal{L}_{\text{Wass}}(\psi_0, \psi_1, \omega) = \frac{1}{N_0} \sum_{i=1}^{N_0} f_c(h_0^i) - \frac{1}{N_1} \sum_{j=1}^{N_1} f_c(h_1^j). \quad (5)$$

To enforce the Lipschitz constraint on the critic function  $f_c$ , we choose to minimize the gradient penalty loss (Gulrajani et al., 2017) meanwhile:

$$\mathcal{L}_{\text{GP}}(\omega) = \mathbb{E}_{\hat{h} \sim \hat{D}}[(\|\nabla f_c(\hat{h})\|_2 - 1)^2], \quad (6)$$

where  $\hat{D}$  represents the distribution of a uniformly distributed linear interpolations between the group visitation

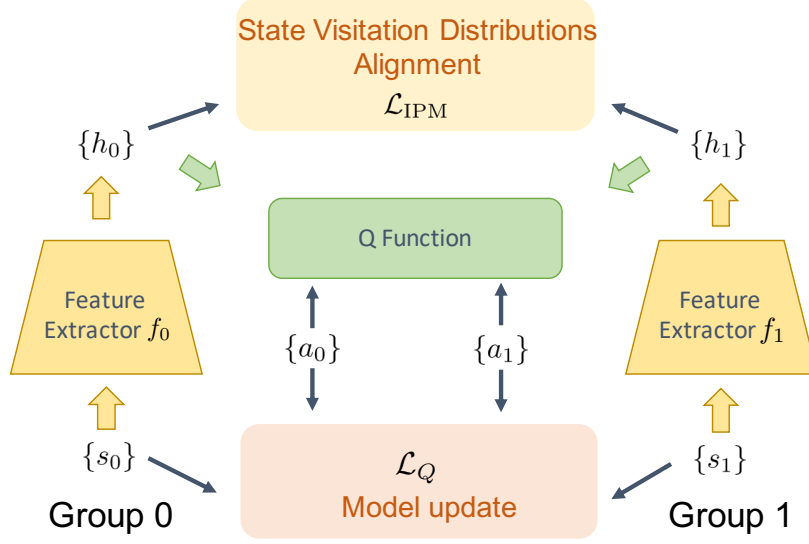


Figure 1: Illustration of our algorithms. At every iteration, we first update the models (*e.g.*, Q-function and feature extractors). The feature extractors are then updated to minimize the loss of state visitation distributions alignment via minimizing the estimated integral probability metrics between different groups.

distributions. Finally, the overall minimax objective becomes

$$\min_{\psi_0, \psi_1} \max_{\omega} \mathcal{L}_{\text{Wass}}(\psi_0, \psi_1, \omega) - \beta \mathcal{L}_{\text{GP}}(\omega), \quad (7)$$

where  $\beta$  is the balancing coefficient and by convention it is set to be 10 (Gulrajani et al., 2017). To ensure the training stability and do less harm to the group that obtains lower return, we use block coordinate descent algorithm (Wright, 2015) to update the feature extractors: at each iteration of state visitation distributions alignment, we first identify the group with higher return choose to update its feature extractor while fixing the parameters of the feature extractor in the other group. Figure 1 illustrates our algorithm framework.

**MMD Variant** Similarly to Wasserstein distance, we can also use maximum mean discrepancy to align state visitation distributions. Let  $k$  be the kernel of the corresponding RKHS  $\mathcal{H}$  on the feature space, then the squared MMD between the induced feature distributions of two groups  $\mathbf{D}_{h_0}$  and  $\mathbf{D}_{h_1}$  is

$$\text{MMD}^2(\mathbf{D}_{h_0}, \mathbf{D}_{h_1}) := \mathbb{E}_{h_0, h'_0} [k(h_0, h'_0)] + \mathbb{E}_{h_1, h'_1} [k(h_1, h'_1)] - 2\mathbb{E}_{h_0, h_1} [k(h_0, h_1)].$$

In practice, given samples from  $\{h_0^1, \dots, h_0^{N_0}\} \sim \mathbf{D}_{h_0}$  and  $\{h_1^1, \dots, h_1^{N_1}\} \sim \mathbf{D}_{h_1}$ , then unbiased estimation of the squared MMD is

$$\mathcal{L}_{\text{MMD}}(\psi_0, \psi_1) = \frac{1}{N_0(N_0 - 1)} \sum_{i \neq i'} k(h_0^i, h_0^{i'}) + \frac{1}{N_1(N_1 - 1)} \sum_{j \neq j'} k(h_1^j, h_1^{j'}) - \frac{2}{N_0 N_1} \sum_i \sum_j k(h_0^i, h_1^j) \quad (8)$$

To align state visitation distributions, we optimize the feature extractor using Eq. 8. In the implementation of the MMD variant, we use a linear combination of RBF kernels with bandwidths  $\{0.001, 0.005, 0.01, 0.05, 0.1, 1, 5, 10\}$  since it remains an open problem on choosing the optimal kernels.



## 5 Experiments

In the following, we conduct empirical evaluation on two real-world datasets, showing that our proposed algorithms help to mitigate return disparity while maintaining the performance of policies.

### 5.1 Experimental Setup

**Datasets** The two real-world datasets we use are benchmark recommender system datasets with user demographic information (*e.g.*, gender and age):

- MovieLens-1M<sup>2</sup> is a benchmark dataset consisting of 1 million ratings from more than 6000 users on more than 4000 movies on the MovieLens website. The movie ratings range from 1 to 5 and the users are provided with demographic information such as gender and age.
- Book-Crossing<sup>3</sup> is a book rating dataset collected from the Book-Crossing community. It consists of more than 1 million ratings from more than 278k users on about 271k books. The book ratings range from 0 to 10, and the users are provided with demographic information such as age.

Our goal of using these datasets is to learn a recommender system that maximizes the expected long-term user satisfaction while keeping the user satisfaction in different demographic groups similar.

**Environment Simulator** We focus on evaluating our proposed RL algorithms on the two benchmark datasets by setting up an environment simulator (Chen et al., 2019; Zhou et al., 2020) to mimic online environments. Specifically, we normalize the user ratings in each dataset into the range  $[-1, 1]$ . Given a user’s historical interaction with the recommender system at time step  $t$  (state  $s_t$ ), the recommender system recommends an item (action  $a_t$ ), and the user gives a rating to the recommended item (reward  $r_t$ ). Following (Chen et al., 2019), we give the details of the state representation scheme used in this work in Figure 2.

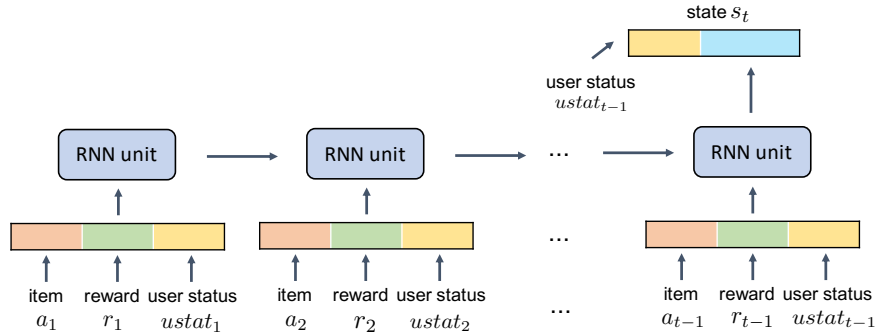


Figure 2: State representation in our RL environments.

As shown in Figure 2, the user state  $s_t$  is constructed by concatenating the user status at  $t - 1$  and the  $t - 1$ -th output of a recurrent neural network (RNN). The user status at  $t - 1$  contains statistics information such as the number of positive/negative rewards before time step  $t$ . The input of the RNN at each time step is composed of three signals: the recommended item, the reward gained by recommending the corresponding

<sup>2</sup><https://grouplens.org/datasets/movielens/>

<sup>3</sup><http://www2.informatik.uni-freiburg.de/~cziegler/BX/>

item, and the user status. Note that each item and reward are mapped to an embedding vector and a one-hot vector before inputting to the RNN. We perform matrix factorization (Koren et al., 2009) to train an item embedding for each item to recommend.

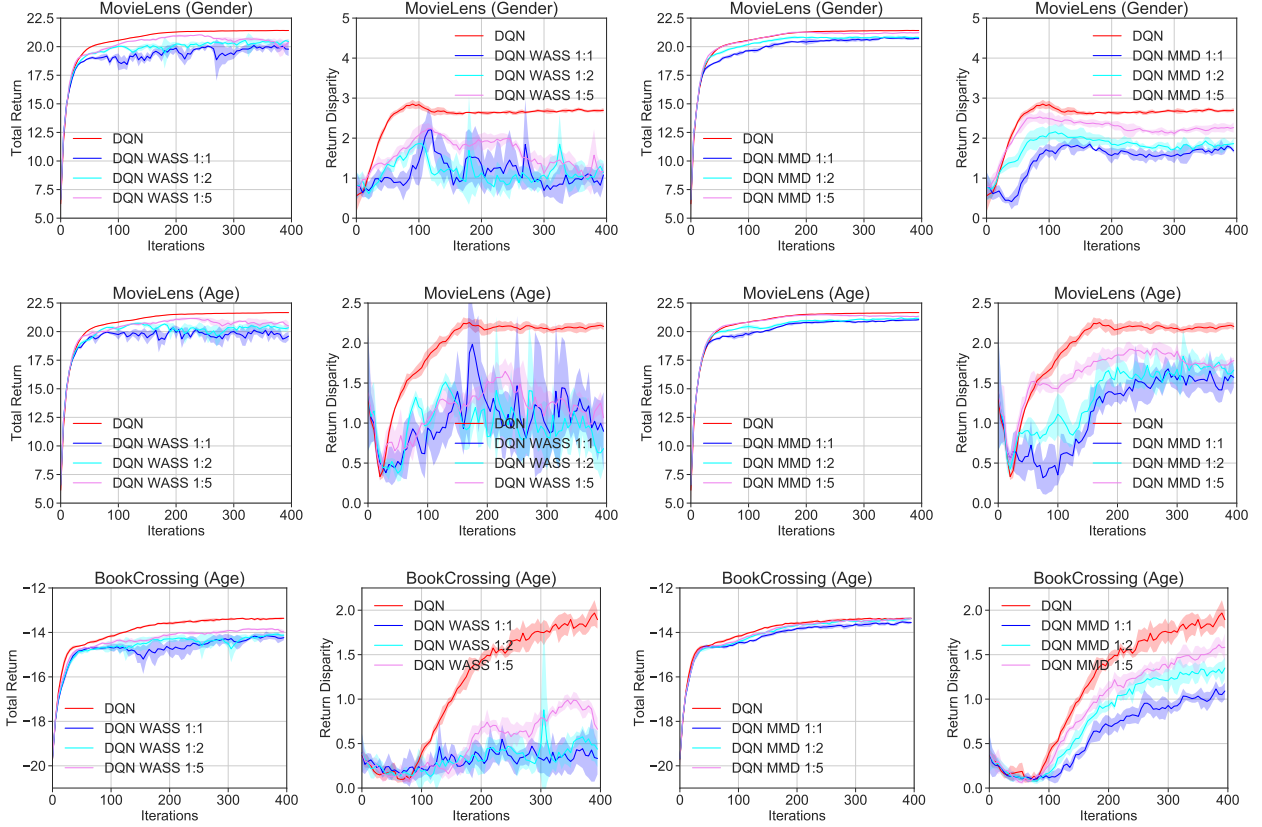


Figure 3: Learning curves of DQN, DQN-WASS and DQN-MMD in three different settings. The legend DQN WASS (DQN MMD) X:Y indicates the interval of model update versus the interval of state visitation distributional alignment with Wasserstein-1 distance (MMD distance) is X:Y (*i.e.*, smaller numbers means more frequent updates). With the increase of the frequency of state visitation distributional alignment, return disparities are decreasing at the cost of performances of policies in all environments.

For each dataset, we randomly split the users into two parts: 80% of the users are used for training, and the other 20% are used for testing. Due to the way we perform the train/test split in our datasets, our experiments are cold-start scenarios: the users in the test set have never been seen during training, and there is no interaction between the recommender system and the users at first. To deal with the problem, our recommender system recommends a popular item among the training users to a test user at time step  $t_0$  and recommends non-repeated items to the user interactively according to users' feedback. The episode length is set to be 32 for each user in the two datasets in our experiments.

**Methods and Implementation Details** We implement the DQN algorithm and our proposed algorithms that perform state visitation distributional alignment via Wasserstein-1 distance (DQN-WASS) and maximum mean discrepancy (DQN-MMD). To the best of our knowledge, our work is the first work that studies

the return disparity problem in MDPs. We also adapt the state-of-the-art reduction approach, constrained policy optimization (CPO) (Achiam et al., 2017) to our problem setting and find its training processes cannot converge, so we do not include the results here. We suspect the failure of CPO is because their setting is different from ours: they assume the constraint of the policy is a deterministic function determined by states and actions, while in our setting, the reward gap in each environment step is dynamically changing, making it hard to estimate.

We take gender and age in MovieLens dataset and age in Book-Crossing dataset as the binary demographic groups (*e.g.*, male/female, young/old). In each environment, we vary the model update frequency and the state visitation distributional alignment update frequency in DQN-WASS and DQN-MMD and report the corresponding overall return and the return disparity between groups. We average the results over five different random seeds and visualize the performance curves in each setting. The detailed data pre-processing pipelines and hyper-parameter configurations in our experiments are presented in Appendix B.

## 5.2 Results and Analysis

The performance curves of DQN, DQN-WASS and DQN-MMD are shown in Figure 3. We can see that with the increase of the frequency of state visitation distributional alignment, return disparities are decreasing at the cost of performances of policies in all environments, which demonstrates the trade-offs between return maximization and return parity. Our methods can flexibly tune the trade-off between return maximization and return parity by controlling the relative update frequency ratio between the model update step and state visitation distributional alignment update step. Compared to DQN-WASS, DQN-MMD leads to more stable training processes with slightly higher overall returns and return disparities on average.

Next, we provide more insights into how our algorithms work. Since the MMD distance can be estimated analytically using Eq. 8, we visualize the learning curves of estimated MMD distances between the (induced) state visitation distributions in different MMD update settings in MovieLens (Gender) in Figure 4. We can see that the more frequent the MMD update is, the smaller the MMD distance. We also perform principal component analysis (PCA) on sampled state representations of different groups for DQN and DQN-MMD in MovieLens (Gender). The visualization results are presented in Figure 5. We can see from Figure 5 that DQN-MMD helps to align the state visitation distributions of different groups, which is consistent with our theoretical findings in Theorem 3.1.

## 6 Related Work

**Fairness in MDPs** Jabbari et al. (2017) propose an individual fairness notion which requires an algorithm to select an action if the long-term (discounted) reward of choosing that action is higher than the others in MDPs; Wen et al. (2019) extend static fairness notions to Markov decision processes and propose model-based and model-free algorithms to maximize expected return while satisfying the fairness constraints. However, their proposed algorithms are based on linear programming or evolutionary algorithm, which cannot be scaled up to solve complex tasks compared to deep reinforcement learning approaches; Siddique et al. (2020) consider the fairness of multi-objective MDPs and propose to learn a policy with objective function satisfy the generalized Gini social welfare function (Weymark, 1981); Ge et al. (2021) consider the fairness of item exposure problem in recommendation systems and extend CPO algorithm (Achiam et al., 2017) to satisfy the fairness constraint; Thomas et al. (2019) propose an offline RL algorithm called quasi-Sheldonian

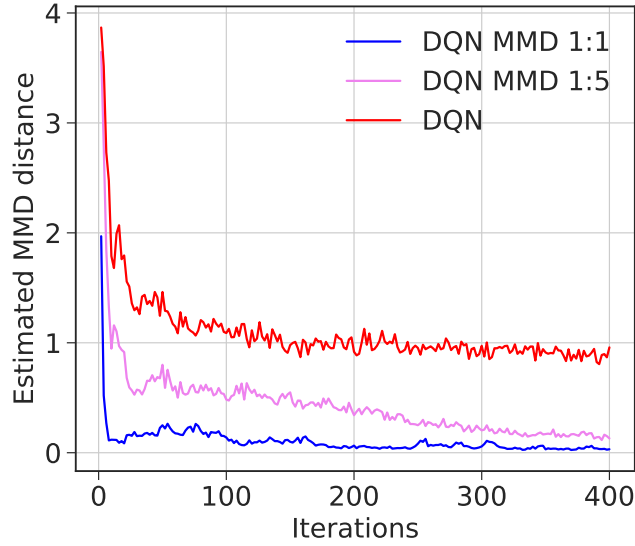


Figure 4: Training visualization of estimated MMD distance in MovieLens (Gender). With the increased frequency of MMD update, the MMD distance becomes smaller.

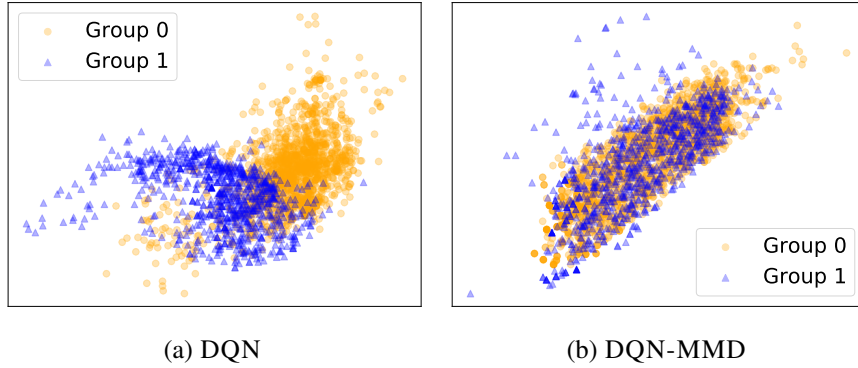


Figure 5: PCA visualization of sampled state visitation representations of different groups for DQN and DQN-MMD in MovieLens (Gender). Our method helps to align the state visitation distributions of different groups.

reinforcement learning algorithm to determine whether given sets of policy distributions satisfy a set of return constraints with guarantees; Zhang et al. (2020) analyze how static fairness constraints affect the dynamics of group qualification rates. In contrast to their work, we propose the notion of return parity to quantify the dynamics of performance disparity in general changing environments: we theoretically analyze the notion of return parity by providing sufficient conditions where return parity can be satisfied and propose a decomposition theorem of return disparity. Based on our decomposition theorem, we propose algorithms to mitigate return disparity and empirically show the effectiveness of our algorithms.

**Fairness under Other Temporal Models** A series of works focus on study fairness in the setting of online learning (Blum et al., 2018; Bechavod et al., 2019; Gupta and Kamble, 2019), multi-armed bandit (Joseph et al., 2016; Patil et al., 2020; Chen et al., 2020), and one-step feedback model (Liu et al., 2018; Hu and Chen, 2018; Kannan et al., 2019). In particular, Hashimoto et al. (2018) show that empirical risk minimization amplifies representation disparity over time with a low group retention rate for the underrepresented group. They further propose distributionally robust optimization to minimize the worst-case risk overall group distributions. Heidari and Krause (2018) propose a time-dependent individual fairness notion that requires similar individuals should receive similar outcomes during the same time epoch. Ensign et al. (2018) model predictive policing problem using Pólya urn model and show that all police officers will be allocated to one location if more officers are constantly assigned to the locations with higher predicted crime rates. Heidari et al. (2019) propose effort-based fairness, which measures unfairness as the disparity in the effort made by individuals from each group to get a target outcome. Zhang et al. (2019) quantify the condition of the exacerbation of relative group ratio under the fairness constraints such as demographic parity and equalized odds.

## 7 Conclusions

In this paper, we investigate the problem of return parity in MDPs theoretically and empirically. In particular, we prove a decomposition theorem for return disparity which decomposes the return disparity of any two MDPs into the distance between group-wise reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions. We then provide algorithmic interventions to mitigate return disparity via state visitation distributional alignment with IPMs. To corroborate our results, we conduct experiments on two real-world benchmark datasets. Experimental results suggest that our proposed algorithms help to mitigate return disparity while maintaining the performance of policies. We believe our work takes an important step towards better understanding the dynamics of performance disparity in changing environments.

## References

- Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International Conference on Machine Learning*, pages 22–31. PMLR, 2017.
- Alekh Agarwal, Miroslav Dudik, and Zhiwei Steven Wu. Fair regression: Quantitative definitions and reduction-based algorithms. In *International Conference on Machine Learning*, pages 120–129, 2019.
- Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.
- Yahav Bechavod, Katrina Ligett, Aaron Roth, Bo Waggoner, and Steven Z Wu. Equal opportunity in online classification with partial feedback. In *Advances in Neural Information Processing Systems*, pages 8974–8984, 2019.
- Avrim Blum, Suriya Gunasekar, Thodoris Lykouris, and Nati Srebro. On preserving non-discrimination when combining expert advice. In *Advances in Neural Information Processing Systems*, pages 8376–8387, 2018.
- Haokun Chen, Xinyi Dai, Han Cai, Weinan Zhang, Xuejian Wang, Ruiming Tang, Yuzhou Zhang, and Yong Yu. Large-scale interactive recommendation with tree-structured policy gradient. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3312–3320, 2019.
- Yifang Chen, Alex Cuellar, Haipeng Luo, Jignesh Modi, Heramb Nemlekar, and Stefanos Nikolaidis. Fair contextual multi-armed bandits: Theory and experiments. In *Conference on Uncertainty in Artificial Intelligence*, pages 181–190. PMLR, 2020.
- Jianfeng Chi, Yuan Tian, Geoffrey J Gordon, and Han Zhao. Understanding and mitigating accuracy disparity in regression. In *International conference on machine learning*, 2021.
- Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression with wasserstein barycenters. *arXiv preprint arXiv:2006.07286*, 2020.
- Daniela Pucci De Farias. *The linear programming approach to approximate dynamic programming: Theory and application*. PhD thesis, stanford university, 2002.
- Daniela Pucci De Farias and Benjamin Van Roy. The linear programming approach to approximate dynamic programming. *Operations research*, 51(6):850–865, 2003.
- Gabriel Dulac-Arnold, Richard Evans, Hado van Hasselt, Peter Sunehag, Timothy Lillicrap, Jonathan Hunt, Timothy Mann, Theophane Weber, Thomas Degris, and Ben Coppin. Deep reinforcement learning in large discrete action spaces. *arXiv preprint arXiv:1512.07679*, 2015.
- Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.
- Hadi Elzayn, Shahin Jabbari, Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, and Zachary Schutzman. Fair algorithms for learning in allocation problems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 170–179, 2019.

- Danielle Ensign, Sorelle A Friedler, Scott Neville, Carlos Scheidegger, and Suresh Venkatasubramanian. Run-away feedback loops in predictive policing. In *Conference on Fairness, Accountability and Transparency*, pages 160–171, 2018.
- Yingqiang Ge, Shuchang Liu, Ruoyuan Gao, Yikun Xian, Yunqi Li, Xiangyu Zhao, Changhua Pei, Fei Sun, Junfeng Ge, Wenwu Ou, et al. Towards long-term fairness in recommendation. *arXiv preprint arXiv:2101.03584*, 2021.
- Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, page 5769–5779, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.
- Swati Gupta and Vijay Kamble. Individual fairness in hindsight. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 805–806, 2019.
- Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 29:3315–3323, 2016.
- Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1929–1938, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL <http://proceedings.mlr.press/v80/hashimoto18a.html>.
- Hoda Heidari and Andreas Krause. Preventing disparate treatment in sequential decision making. In *IJCAI*, pages 2248–2254, 2018.
- Hoda Heidari, Vedant Nanda, and Krishna Gummadi. On the long-term impact of algorithmic decision policies: Effort unfairness and feature segregation through social learning. In *International Conference on Machine Learning*, pages 2692–2701, 2019.
- Lily Hu and Yiling Chen. A short-term intervention for long-term fairness in the labor market. In *Proceedings of the 2018 World Wide Web Conference*, pages 1389–1398, 2018.
- Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. Fairness in reinforcement learning. In *International Conference on Machine Learning*, pages 1617–1626, 2017.
- Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang, and Silvia Chiappa. Wasserstein fair classification. In *Uncertainty in Artificial Intelligence*, pages 862–872. PMLR, 2020.
- Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, pages 325–333, 2016.
- Sampath Kannan, Aaron Roth, and Juba Ziani. Downstream effects of affirmative action. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 240–248, 2019.
- Junpei Komiyama, Akiko Takeda, Junya Honda, and Hajime Shimao. Nonconvex optimization for regression with fairness constraints. In *International conference on machine learning*, pages 2737–2746, 2018.

- Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.
- Feng Liu, Huifeng Guo, Xutao Li, Ruiming Tang, Yunming Ye, and Xiuqiang He. End-to-end deep reinforcement learning based recommendation with supervised embedding. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 384–392, 2020a.
- Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. Delayed impact of fair machine learning. In *International Conference on Machine Learning*, pages 3150–3158, 2018.
- Lydia T Liu, Ashia Wilson, Nika Haghtalab, Adam Tauman Kalai, Christian Borgs, and Jennifer Chayes. The disparate equilibria of algorithmic decision making when individuals invest rationally. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 381–391, 2020b.
- Kristian Lum and William Isaac. To predict and serve? *Significance*, 13(5):14–19, 2016.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.
- Hussein Mouzannar, Mesrob I Ohannessian, and Nathan Srebro. From fair decision making to social equality. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 359–368, 2019.
- Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997.
- Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y Narahari. Achieving fairness in the stochastic multi-armed bandit problem. In *AAAI*, pages 5379–5386, 2020.
- Umer Siddique, Paul Weng, and Matthieu Zimmer. Learning fair policies in multi-objective (deep) reinforcement learning with average and discounted rewards. In *International Conference on Machine Learning*, pages 8905–8915. PMLR, 2020.
- Philip S Thomas, Bruno Castro da Silva, Andrew G Barto, Stephen Giguere, Yuriy Brun, and Emma Brunskill. Preventing undesirable behavior of intelligent machines. *Science*, 366(6468):999–1004, 2019.
- Sebastian Thrun and Anton Schwartz. Issues in using function approximation for reinforcement learning. In *Proceedings of the Fourth Connectionist Models Summer School*, pages 255–263. Hillsdale, NJ, 1993.
- Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.
- Min Wen, Osbert Bastani, and Ufuk Topcu. Fairness with dynamics. *arXiv preprint arXiv:1901.08568*, 2019.
- John A Weymark. Generalized gini inequality indices. *Mathematical Social Sciences*, 1(4):409–430, 1981.
- Stephen J Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.



- Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pages 962–970, 2017.
- Xueru Zhang, Mohammadmahdi Khaliligarekani, Cem Tekin, et al. Group retention when using machine learning in sequential decision making: the interplay between user dynamics and fairness. In *Advances in Neural Information Processing Systems*, pages 15269–15278, 2019.
- Xueru Zhang, Ruibo Tu, Yang Liu, Hedvig Kjellstrom, Kun Zhang, Cheng Zhang, et al. How do fair decisions fare in long-term qualification? *Advances in Neural Information Processing Systems*, 33, 2020.
- Han Zhao and Geoffrey J Gordon. Inherent tradeoffs in learning fair representations. In *Advances in neural information processing systems*, 2019.
- Guanjie Zheng, Fuzheng Zhang, Zihan Zheng, Yang Xiang, Nicholas Jing Yuan, Xing Xie, and Zhenhui Li. Drn: A deep reinforcement learning framework for news recommendation. In *Proceedings of the 2018 World Wide Web Conference*, pages 167–176, 2018.
- Sijin Zhou, Xinyi Dai, Haokun Chen, Weinan Zhang, Kan Ren, Ruiming Tang, Xiuqiang He, and Yong Yu. Interactive recommender system via knowledge graph-enhanced reinforcement learning. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 179–188, 2020.

## Appendix

### A Missing Proofs

#### A.1 Proof of Proposition 3.1

**Proposition 3.1.** For any constant  $c > 0$ , there exist two MDPs that share the same state and action spaces, such that  $\forall \pi_0, \pi_1 \in \Pi$ , the return disparity  $\Delta_{\text{Ret}} \geq c$ .

*Proof.* Consider two MDPs share two states  $s_1$  and  $s_2$ . Let  $r(s_1, a) = c(1 - \gamma) > 0$  and  $r(s_2, a) = 0, \forall a \in \mathcal{A}$ ,  $T(s_2 | s_1, a) = T(s_1 | s_2, a) = 0$  and  $T(s_1 | s_1, a) = T(s_2 | s_2, a) = 1, \forall a \in \mathcal{A}$ . Given  $\mu_0 = [1, 0]^T$  and  $\mu_1 = [0, 1]^T$ , then the expected return for group 0 is  $c$ , while the expected return for group 1 is 0. In this case, the return gap  $\Delta_{\text{Ret}} = c \geq 0$ . ■

#### A.2 Proof of Proposition 3.2

**Proposition 3.2.** Assume i) the reward functions and the initial distributions are the same across different groups, i.e.,  $r_0(s, a) = r_1(s, a)$  and  $\mu_0 = \mu_1$ ; ii) reward functions only depend on states, i.e.,  $r(s, a) = r(s)$ . Define  $\mathbf{d}_{ij} := [T_0(s_1 | a_i, s_j) - T_1(s_1 | a_i, s_j), \dots, T_0(s_m | a_i, s_j) - T_1(s_m | a_i, s_j)]^T$ . For  $\forall \mathbf{c} \in \mathbb{R}^m \setminus \mathbf{1}_m$ , if there exists  $i \in [n], j \in [m]$ , such that  $\langle \mathbf{c}, \mathbf{d}_{ij} \rangle \leq 0$ , then there exist  $\pi_0, \pi_1 \in \Pi$  such that 0-return parity is satisfied.

*Proof.* To start our proof, we first revisit Bellman equation in its matrix form:

$$V^\pi = R^\pi + \gamma P^\pi V^\pi,$$

where  $V^\pi = [v^\pi(s_1), \dots, v^\pi(s_m)]^T$  and  $R^\pi = [r^\pi(s_1), \dots, r^\pi(s_m)]^T$ . The analytic solution of value function is

$$V^\pi = (I - \gamma P^\pi)^{-1} R^\pi.$$

Given the initial distribution  $\mu_0$  and  $\mu_1$  for different groups, in order to satisfy return parity, we have:

$$\mu_0^T (I - \gamma P^{\pi_0})^{-1} R^{\pi_0} = \mu_1^T (I - \gamma P^{\pi_1})^{-1} R^{\pi_1}$$

Since we already assume that  $\mu_0 = \mu_1$  and  $R^{\pi_0} = R^{\pi_1}$ , it suffice to show  $P^{\pi_0} = P^{\pi_1}$  in order to satisfy return parity. In other word, the following linear conditions w.r.t  $\pi_0$  and  $\pi_1$  should satisfy:

$$\sum_a T_0(s_j | a, s_i) \pi_0(a | s_i) = \sum_a T_1(s_j | a, s_i) \pi_1(a | s_i) \quad \forall i, j \in [m]$$

Note that the group policies should also satisfy the following constraints:

$$\begin{aligned} \sum_{a_j} \pi_0(a_j | s_i) &= 1 \quad \text{and} \quad \sum_{a_j} \pi_1(a_j | s_i) = 1 \quad \forall i \in [m], j \in [n] \\ \pi_0(a_j | s_i) &\geq 0 \quad \text{and} \quad \pi_1(a_j | s_i) \geq 0 \quad \forall i \in [m], j \in [n] \end{aligned}$$

Next, we transform the above linear inequalities to the equivalent standard form of LP, which gives us

$$\begin{aligned}
\sum_a T_0(s_j | a, s_i) \pi_0(a | s_i) - \sum_a T_1(s_j | a, s_i) \pi_1(a | s_i) &\leq 0 \quad \forall i, j \in [m] \\
\sum_a T_1(s_j | a, s_i) \pi_1(a | s_i) - \sum_a T_0(s_j | a, s_i) \pi_0(a | s_i) &\leq 0 \quad \forall i, j \in [m] \\
\sum_a \pi_g(a | s_i) &\leq 1 \quad \forall g \in \{0, 1\} \\
-\sum_a \pi_g(a | s_i) &\leq -1 \quad \forall g \in \{0, 1\} \\
\pi_g(a_k | s_i) &\geq 0 \quad \forall g \in \{0, 1\}, k \in [n], i \in [m].
\end{aligned}$$

Next, we represent the above system in the form of  $\mathbf{Ax} \leq \mathbf{b}$  with  $\mathbf{x} \geq 0$ . Let  $\mathbf{x} = [\pi_0(a_1 | s_1), \pi_0(a_2 | s_1), \dots, \pi_0(a_n | s_m), \pi_1(a_1 | s_1), \pi_1(a_2 | s_1), \dots, \pi_1(a_n | s_m)]^T \in \mathbb{R}^{2 \cdot m \cdot n}$ ,  $\mathbf{b} = [\mathbf{0}_{1 \times m \cdot m}, 1, -1, 1, -1]^T \in \mathbb{R}^{2m+4}$ , construct

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \mathbf{A}_3 \end{pmatrix} \in \mathbb{R}^{(2 \cdot m + 4) \times (2 \cdot m \cdot n)}$$

where  $\mathbf{A}_1 \in \mathbb{R}^{m \times (2 \cdot m \cdot n)}$  and the  $(m(i-1) + j)$ -th line of  $\mathbf{A}_1$  is

$$[\mathbf{0}_{1 \times n \cdot (i-1)}, T_0(s_j | a_1, s_i), \dots, T_0(s_j | a_n, s_i), \mathbf{0}_{1 \times n \cdot (m-1)}, -T_1(s_j | a_1, s_i), \dots, -T_1(s_j | a_n, s_i), \mathbf{0}_{1 \times n \cdot (m-i)}],$$

$\mathbf{A}_2 \in \mathbb{R}^{m \times (2 \cdot m \cdot n)}$  and the  $(m(i-1) + j)$ -th line of  $\mathbf{A}_2$  is

$$[\mathbf{0}_{1 \times n \cdot (i-1)}, -T_0(s_j | a_1, s_i), \dots, -T_0(s_j | a_n, s_i), \mathbf{0}_{1 \times n \cdot (m-1)}, T_1(s_j | a_1, s_i), \dots, T_1(s_j | a_n, s_i), \mathbf{0}_{1 \times n \cdot (m-i)}],$$

and

$$\mathbf{A}_3 = \begin{pmatrix} \mathbf{A}'_3 & \mathbf{0}_{2 \times m \cdot n} \\ \mathbf{0}_{2 \times m \cdot n} & \mathbf{A}'_3 \end{pmatrix} \in \mathbb{R}^{4 \times (2 \cdot m \cdot n)} \quad \text{where } \mathbf{A}'_3 = \begin{pmatrix} 1 & \dots & 1 \\ -1 & \dots & -1 \end{pmatrix} \in \mathbb{R}^{2 \times (m \cdot n)}.$$

The variant of Farkas' Lemma states that either the system  $\mathbf{Ax} \leq \mathbf{b}$  has a solution with  $\mathbf{x} \geq 0$ , or the system  $\mathbf{A}^T \mathbf{y} \geq 0$  has a solution with  $\mathbf{b}^T \mathbf{y} < 0$  and  $\mathbf{y} \geq 0$ . In other words, if we want the above system has a non-negative solution  $\mathbf{x}$ , then  $\nexists \mathbf{y} = [y_1, y_2, \dots, y_{2m+4}]^T \in \mathbb{R}^{2m+4}$ , such that

$$\begin{aligned}
\sum_k^m (y_{k+(i-1)(j-1)} - y_{k+(i-1)(j-1)+mn}) T_0(s_k | a_i, s_j) + y_{2m+1} - y_{2m+2} &\geq 0 \quad \forall i \in [n], j \in [m] \\
\sum_k^m (y_{k+(i-1)(j-1)+mn} - y_{k+(i-1)(j-1)}) T_1(s_k | a_i, s_j) + y_{2m+3} - y_{2m+4} &\geq 0 \quad \forall i \in [n], j \in [m] \\
y_{2m+1} - y_{2m+2} + y_{2m+3} - y_{2m+4} &< 0 \\
y_k &\geq 0 \quad \forall k \in \{1, \dots, 2m+4\}.
\end{aligned}$$

To proceed, let  $a = y_{2m+2} - y_{2m+1}$  and  $b = y_{2m+4} - y_{2m+3}$ ,  $c_k = y_{k+(i-1)(j-1)} - y_{k+(i-1)(j-1)+mn}$  for

$k = [m]$ , then the conditions above are equivalent to  $\nexists a, b \in \mathbb{R}$  and  $c_k \in \mathbb{R}, \forall k = [m]$  such that

$$\begin{aligned} \sum_k^m c_k T_0(s_k \mid a_i, s_j) &\geq a \quad \forall i \in [n], j \in [m] \\ -\sum_k^m c_k T_1(s_k \mid a_i, s_j) &\geq b \quad \forall i \in [n], j \in [m] \\ a + b &> 0. \end{aligned}$$

It is easy to see that the above conditions above can be transformed into

$$\sum_k^m c_k (T_0(s_k \mid a_i, s_j) - T_1(s_k \mid a_i, s_j)) > 0 \quad \forall i \in [n], j \in [m].$$

Next, we define  $\mathbf{d}_{ij} := [T_0(s_1 \mid a_i, s_j) - T_1(s_1 \mid a_i, s_j), \dots, T_0(s_m \mid a_i, s_j) - T_1(s_m \mid a_i, s_j)]^T$ , and it is easy to verify that  $\mathbf{d}_{ij} \in [-1, 1]^m$  and  $\mathbf{1}_m^T \mathbf{d}_{ij} = 0$ . Then, the conditions above are equivalent to  $\nexists \mathbf{c} \in \mathbb{R}^m$  such that  $\forall i, j$ ,

$$\begin{aligned} \mathbf{c}^T \mathbf{d}_{ij} &> 0 \\ \mathbf{d}_{ij} &\in [-1, 1]^m \\ \mathbf{1}_m^T \mathbf{d}_{ij} &= 0. \end{aligned}$$

To proceed, let  $\mathbf{c} = \mathbf{c}^\perp + \mathbf{c}^\parallel$ , where  $\mathbf{c}^\perp \in \mathbb{R}^m \setminus \mathbf{1}_m^\perp$ , and  $\mathbf{c}^\parallel \parallel \mathbf{1}_m$ . It follows that  $(\mathbf{c}^\perp)^T \mathbf{d}_{ij} = 0, \forall i, j$ . The conditions above now become  $\nexists \mathbf{c}^\parallel \in \mathbb{R}^m \setminus \mathbf{1}_m \subseteq \mathbb{R}^{m-1}$ , such that  $\langle \mathbf{c}^\parallel, \mathbf{d}_{ij} \rangle > 0, \forall i, j$ . By the law of contraposition, it is equivalent to its contrapositive:  $\forall \mathbf{c}^\parallel \in \mathbb{R}^m \setminus \mathbf{1}_m, \exists i, j, \langle \mathbf{c}^\parallel, \mathbf{d}_{ij} \rangle \leq 0$ . Replacing  $\mathbf{c}^\parallel$  with  $\mathbf{c}$  finishes the proof.  $\blacksquare$

### A.3 Proof of Proposition 3.3

**Proposition 3.3.** For  $\forall \hat{\rho}_0(s, a), \hat{\rho}_1(s, a), b_0, b_1 \geq 0$ , if there exists  $i \in [m]$ , such that

$$\begin{aligned} \sum_a \hat{\rho}_0(s_i, a) - \gamma \sum_s \sum_a T_0(s_i \mid s, a) \hat{\rho}_0(s, a) &> (b_0 - b_1)(\mu_0)_i \\ \sum_a \hat{\rho}_1(s_i, a) - \gamma \sum_s \sum_a T_1(s_i \mid s, a) \hat{\rho}_1(s, a) &> (b_1 - b_0)(\mu_1)_i \\ \sum_s \sum_a \hat{\rho}_0(s, a) r_0(s, a) + \hat{\rho}_1(s, a) r_1(s, a) &\leq (b_0 + b_1)\epsilon \end{aligned}$$

then the optimal policies  $\pi_0^*$  and  $\pi_1^*$  that maximize the expected returns of two MDPs satisfy  $\epsilon$ -return parity.

*Proof.* Let  $\mu_0$  and  $\mu_1$  be initial distributions of group 0 and 1 respectively and denote  $V^{\pi_0} = [v^{\pi_0}(s_1), \dots, v^{\pi_0}(s_m)]^T$  and  $V^{\pi_1} = [v^{\pi_1}(s_1), \dots, v^{\pi_1}(s_m)]^T$ . By definition, in order to satisfy return parity, the following equation should be satisfied:

$$\mu_0^T V^{\pi_0} = \mu_0^T V^{\pi_1}.$$

Consider solving the optimal value function via linear programming while satisfying the constraint of return parity, then the whole optimization problem becomes

$$\begin{aligned}
& \min_{V^{\pi_0}, V^{\pi_1}} \quad \lambda \mu_0^T V^{\pi_0} + (1 - \lambda) \mu_1^T V^{\pi_1} \\
& \text{s.t.} \quad v^{\pi_0}(s) \geq r_0(s, a) + \gamma \sum_{s'} T_0(s' | s, a) v^{\pi_0}(s'), \quad \forall a \in \mathcal{A}, s \in \mathcal{S} \\
& \quad v^{\pi_1}(s) \geq r_1(s, a) + \gamma \sum_{s'} T_1(s' | s, a) v^{\pi_1}(s'), \quad \forall a \in \mathcal{A}, s \in \mathcal{S} \\
& \quad |\mu_0^T V^{\pi_0} - \mu_1^T V^{\pi_1}| \leq \epsilon
\end{aligned}$$

Next, we convert the constraints of the LP in the standard form:

$$\begin{aligned}
& (\gamma T_0(s | s, a) - 1) v^{\pi_0}(s) + \gamma \sum_{s' \neq s} T_0(s' | s, a) v^{\pi_0}(s') \leq -r_0(s, a) \quad \forall a \in \mathcal{A}, s \in \mathcal{S} \\
& (\gamma T_1(s | s, a) - 1) v^{\pi_1}(s) + \gamma \sum_{s' \neq s} T_1(s' | s, a) v^{\pi_1}(s') \leq -r_1(s, a) \quad \forall a \in \mathcal{A}, s \in \mathcal{S} \\
& \mu_0^T V^{\pi_0} - \mu_1^T V^{\pi_1} \leq \epsilon \\
& \mu_1^T V^{\pi_1} - \mu_0^T V^{\pi_0} \leq \epsilon.
\end{aligned}$$

Similar to the proof in Proposition 3.2, the variant of Farkas' Lemma states that either the system  $\mathbf{A}\mathbf{x} \leq \mathbf{b}$  has a solution with  $\mathbf{x} \geq 0$ , or the system  $\mathbf{A}^T \mathbf{y} \geq 0$  has a solution with  $\mathbf{b}^T \mathbf{y} < 0$  and  $\mathbf{y} \geq 0$ . In other words, if we want the above system has a non-negative solution, we can easily show that  $\nexists \hat{\rho}_0(s, a), \hat{\rho}_1(s, a), b_0, b_1 \geq 0, \forall i \in [m]$  such that

$$\begin{aligned}
& \sum_s \sum_a T_0(s_i | s, a) \hat{\rho}_0(s, a) - \sum_a \hat{\rho}_0(s_i, a) + (b_0 - b_1)(\mu_0)_i \geq 0 \\
& \sum_s \sum_a T_1(s_i | s, a) \hat{\rho}_1(s, a) - \sum_a \hat{\rho}_1(s_i, a) + (b_1 - b_0)(\mu_1)_i \geq 0 \\
& \sum_s \sum_a \hat{\rho}_0(s, a) r_0(s, a) + \hat{\rho}_1(s, a) r_1(s, a) - (b_0 + b_1)\epsilon > 0
\end{aligned}$$

By the law of contraposition, it is equivalent to its contrapositive:  $\forall \hat{\rho}_0(s, a), \hat{\rho}_1(s, a), b_0, b_1 \geq 0, \exists i \in [m]$  such that

$$\begin{aligned}
& \sum_s \sum_a T_0(s_i | s, a) \hat{\rho}_0(s, a) - \sum_a \hat{\rho}_0(s_i, a) + (b_0 - b_1)(\mu_0)_i < 0 \\
& \sum_s \sum_a T_1(s_i | s, a) \hat{\rho}_1(s, a) - \sum_a \hat{\rho}_1(s_i, a) + (b_1 - b_0)(\mu_1)_i < 0 \\
& \sum_s \sum_a \hat{\rho}_0(s, a) r_0(s, a) + \hat{\rho}_1(s, a) r_1(s, a) - (b_0 + b_1)\epsilon \leq 0
\end{aligned}$$

Reorganizing the above equations then completes the proof. ■

#### A.4 Proof of Theorem 3.1

**Theorem 3.1.** For  $g \in \{0, 1\}$ , given policies  $\pi_0, \pi_1 \in \Pi$  and assume there exists a witness function class  $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}\}$ , such that the reward functions  $r_g(s) = \mathbb{E}_{a \sim \pi_g(\cdot | s)}[r_g(a, s) | s] \in \mathcal{F}$  for  $\forall s \in \mathcal{S}$ ,

$a \in \mathcal{A}$  and  $g \in \{0, 1\}$ , then the following holds:

$$\Delta_{\text{Ret}} \leq \frac{1}{1-\gamma} \left( \|r_0 - r_1\|_\infty + R \cdot \min \left\{ \mathbb{E}_{s \sim \mu^{\pi_0}} [d_{\pi_0, \pi_1}(s)], \mathbb{E}_{s \sim \mu^{\pi_1}} [d_{\pi_0, \pi_1}(s)] \right\} + d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s)) \right),$$

where  $d_{\pi_0, \pi_1}(s) := \|\pi_0(\cdot | s) - \pi_1(\cdot | s)\|_1$ .

*Proof.* By definition, the return disparity is

$$\begin{aligned} \Delta_{\text{Ret}} &= |\mathbb{E}_{s \in \mu_0} [v^\pi(s)] - \mathbb{E}_{s \in \mu_1} [v^\pi(s)]| \\ &= \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \rho^{\pi_1}(s, a) \right| \\ &= \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| \\ &= \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_0}(s, a) + r_1(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| \\ &\leq \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_0}(s, a) \right| + \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right|, \end{aligned}$$

where the first term is upper bounded by

$$\begin{aligned} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_0}(s, a) \right| &= \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} (r_0(s, a) - r_1(s, a)) \rho^{\pi_0}(s, a) \right| \\ &\leq \max_{s, a} |r_0(s, a) - r_1(s, a)| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \rho^{\pi_0}(s, a) = \|r_0 - r_1\|_\infty, \end{aligned}$$

and the second term is upper bounded by

$$\begin{aligned} &\left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| \\ &= \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \mu^{\pi_0}(s) \pi_0(a | s) - r_1(s, a) \mu^{\pi_1}(s) \pi_1(a | s) \right| \\ &= \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \mu^{\pi_0}(s) \pi_0(a | s) - r_1(s, a) \mu^{\pi_0}(s) \pi_1(a | s) + r_1(s, a) \mu^{\pi_0}(s) \pi_1(a | s) - r_1(s, a) \mu^{\pi_1}(s) \pi_1(a | s) \right| \\ &\leq \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \mu^{\pi_0}(s) \pi_0(a | s) - r_1(s, a) \mu^{\pi_0}(s) \pi_1(a | s) \right| \\ &\quad + \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \mu^{\pi_0}(s) \pi_1(a | s) - r_1(s, a) \mu^{\pi_1}(s) \pi_1(a | s) \right|. \end{aligned} \tag{9}$$

The first term in the upper bound of 9 is

$$\begin{aligned}
\left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \mu^{\pi_0}(s) \pi_0(a | s) - r_1(s, a) \mu^{\pi_0}(s) \pi_1(a | s) \right| &\leq R \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \mu^{\pi_0}(s) \left| \pi_0(a | s) - \pi_1(a | s) \right| \\
&\leq R \mathbb{E}_{s \sim \mu^{\pi_0}} \left[ \sum_{a \in \mathcal{A}} \left| \pi_0(a | s) - \pi_1(a | s) \right| \right] \\
&= R \mathbb{E}_{s \sim \mu^{\pi_0}} \left[ \sum_{a \in \mathcal{A}} \left\| \pi_0(\cdot | s) - \pi_1(\cdot | s) \right\|_1 \right],
\end{aligned}$$

and the second term in the upper bound of 9 is

$$\begin{aligned}
&\left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \mu^{\pi_0}(s) \pi_1(a | s) - r_1(s, a) \mu^{\pi_1}(s) \pi_1(a | s) \right| \\
&= \left| \sum_{s \in \mathcal{S}} \mu^{\pi_0}(s) \sum_{a \in \mathcal{A}} r_1(s, a) \pi_1(a | s) - \sum_{s \in \mathcal{S}} \mu^{\pi_1}(s) \sum_{a \in \mathcal{A}} r_1(s, a) \pi_1(a | s) \right| \\
&= \left| \sum_{s \in \mathcal{S}} \mu^{\pi_0}(s) r_1(s) - \sum_{s \in \mathcal{S}} \mu^{\pi_1}(s) r_1(s) \right| \\
&\leq \sup_{r_1(s) \in \mathcal{F}} \left| \sum_{s \in \mathcal{S}} \mu^{\pi_0}(s) r_1(s) - \sum_{s \in \mathcal{S}} \mu^{\pi_1}(s) r_1(s) \right| \\
&= d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s)).
\end{aligned}$$

By symmetry of 9, we also have

$$\begin{aligned}
&\left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| \\
&\leq R \mathbb{E}_{s \sim \mu^{\pi_1}} \left[ \sum_{a \in \mathcal{A}} \left\| \pi_0(\cdot | s) - \pi_1(\cdot | s) \right\|_1 \right] + d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s))
\end{aligned}$$

Combining the above results then completes the proof. ■

## B Experimental Details

**MovieLen-1M** The original rating matrix of the dataset is a sparse matrix. In order to learn a good item embedding for each user, we first filter out the items which get less than 64 ratings (the dataset has ensured that each user has at least 20 ratings). We take gender (*e.g.*, male/female) and age (*e.g.*,  $\geq 45$  /  $< 45$  years old) as the binary demographic groups. Similar to (Zhou et al., 2020), we also perform the matrix completion (Koren et al., 2009) on the original rating matrix to avoid the sparsity of rating signals in the original data. To mimic the real-world scenarios where the number of users could be skewed across different demographic groups, we downsample one group’s data by a factor of 10. Similar to (Chen et al., 2019; Liu et al., 2020a), we fix the item embedding when updating our models and pre-train the RNNs.<sup>4</sup> The dimension of the item embedding vector, reward vector and user status (*i.e.*, the inputs of RNNs) are 50, 20 and 9, respectively. Thus, the state dimension is 88. We give the details of training hyperparameters in Table 1.

<sup>4</sup><https://github.com/chenhaokun/TPGR>

**Book-Crossing** Similar to MovieLen-1M, we first filter out the items which get less than 32 ratings and users who have less than 16 ratings. We take age (*e.g.*,  $\geq 35$  /  $< 35$  years old) as the binary demographic groups. The rest of data pre-processing pipelines are similar to those in the MovieLen dataset. The dimension of the item embedding vector, reward vector and user status (*i.e.*, the inputs of RNNs) are 60, 20 and 9, respectively. Thus, the state dimension is 98. We give the details of training hyperparameters in Table 1.

	MovieLens (Gender)	MovieLens (Age)	Book-Crossing (Age)
Q-network	MLP with hidden size [128]		
Soft-update frequency $\tau$	0.99		
Iteration	400		
Learning Rate	1e-3		
Q-network Weight Decay	1e-6		
$\epsilon$ -greedy Policy	Linear decay from max $\epsilon = 1.0$ to min $\epsilon = 0.1$ with decay steps 160		
Sample Batch Size	1000		
Update Batch Size	10000		
Update per Iteration	10		
Buffer Size	200000		
Feature Extractor	MLP with hidden size the same as the state dimension		

Table 1: Hyperparameter settings in our experiments.

## C Another Decomposition Theorem for Return Disparity

**Theorem C.1.** For  $g \in \{0, 1\}$ , given policies  $\pi_0, \pi_1 \in \Pi$  and assume there exists a witness function class  $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}\}$ , such that the reward functions  $r_g(\cdot, \cdot) \in \mathcal{F}$  for  $\forall s \in \mathcal{S}, a \in \mathcal{A}$  and  $g \in \{0, 1\}$ , then the following holds:

$$\Delta_{\text{Ret}} \leq \frac{1}{1-\gamma} \left( \|r_0 - r_1\|_{\infty} + d_{\mathcal{F}}(\rho^{\pi_0}(s, a), \rho^{\pi_1}(s, a)) \right),$$

**Remark** We see that the return gap is upper bounded by two terms: the distance between group-wise reward functions and the discrepancy between occupancy measures of the two MDPs. Given any two MDPs, the distance between group-wise reward functions is constant. If we further assume the two MDPs share the same reward function (*i.e.*,  $r_0(s, a) = r_1(s, a) = r(s, a)$ ,  $\forall s \in \mathcal{S}, a \in \mathcal{A}$ ), then the upper bound is simplified as

$$\Delta_{\text{Ret}} \leq d_{\mathcal{F}}(\rho^{\pi_0}(s, a), \rho^{\pi_1}(s, a)).$$

In this case, Theorem C.1 implies a sufficient condition to minimize return disparity is to find policies  $\pi_0, \pi_1 \in \Pi$  that minimize the distance between induced occupancy measures in the two MDPs. In what follows, we first give the detailed proof of Theorem C.1 and give the algorithm design inspired by Theorem C.1 in the subsequent section.



*Proof.* By definition, the return disparity is

$$\begin{aligned}
\Delta_{\text{Ret}} &= |\mathbb{E}_{s \in \mu_0}[v^\pi(s)] - \mathbb{E}_{s \in \mu_1}[v^\pi(s)]| \\
&= \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \rho^{\pi_1}(s, a) \right| \\
&= \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| \\
&= \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_0}(s, a) + r_1(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| \\
&\leq \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_0}(s, a) \right| + \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right|,
\end{aligned}$$

where the first term is upper bounded by

$$\begin{aligned}
\left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_0}(s, a) \right| &= \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} (r_0(s, a) - r_1(s, a)) \rho^{\pi_0}(s, a) \right| \\
&\leq \max_{s, a} |r_0(s, a) - r_1(s, a)| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \rho^{\pi_0}(s, a) = \|r_0 - r_1\|_\infty,
\end{aligned}$$

and the second term is upper bounded by

$$\begin{aligned}
\left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| &= \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) (\rho^{\pi_0}(s, a) - \rho^{\pi_1}(s, a)) \right| \\
&\leq \sup_{r_1 \in \mathcal{F}} \left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_1(s, a) (\rho^{\pi_0}(s, a) - \rho^{\pi_1}(s, a)) \right| \\
&= d_{\mathcal{F}}(\rho^{\pi_0}(s, a), \rho^{\pi_1}(s, a)).
\end{aligned}$$

By symmetry, we also have

$$\left| \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r_0(s, a) \rho^{\pi_0}(s, a) - r_1(s, a) \rho^{\pi_1}(s, a) \right| \leq d_{\mathcal{F}}(\rho^{\pi_0}(s, a), \rho^{\pi_1}(s, a)).$$

Combining the above results then completes the proof. ■

## D Algorithms to Mitigate Return Disparity via Occupancy Measures Alignment

Inspired by Theorem C.1, we introduce an occupancy measures alignment procedure when learning group policies for the two MDPs. We use the Wolpertinger policy (Dulac-Arnold et al., 2015) based on Deep Deterministic Policy Gradient (DDPG) (Lillicrap et al., 2015) as our baseline backbone algorithm, which is an actor-critic approach designed for learning policy in large discrete action spaces. We give an outline of our algorithm in Algorithm 2 and readers can refer to (Lillicrap et al., 2015) for more clarification of the Wolpertinger policy. The details of the occupancy measures alignment step are similar to those of state

---

**Algorithm 2** Algorithm to mitigate return disparity via occupancy measures alignment under the framework of (Dulac-Arnold et al., 2015).

---

```

1: Initialize policies  $\pi_{\phi_0}, \pi_{\phi_1}$ , Q-functions  $q_{\theta_1}$  and  $q_{\theta_2}$ , feature extractors  $f_{\psi_0}$  and  $f_{\psi_1}$ , environment buffers  $D_0$  and  $D_1$ 
2: for each iteration do
3:   for each environment step do
4:     for  $g \in \{0, 1\}$  do
5:       Sample an action  $a_g$  using policy  $\pi_{\phi_g}$ ; add the sample  $(s_g, a_g, s'_g, r_g)$  to  $D_g$ ;
6:     end for
7:   end for
8:   for each model update step do
9:     for  $g \in \{0, 1\}$  do
10:      Update policy  $\pi_{\phi_g}$ , Q-function  $q_{\theta_g}$ , and feature extractor  $f_{\psi_g}$  following (Dulac-Arnold et al., 2015);
11:    end for
12:  end for
13:  for each occupancy measures alignment c do
14:    Update feature extractors  $f_{\psi_0}$  and  $f_{\psi_1}$  via occupancy measures alignment via IPM (Wasserstein-1 distance/MMD);
15:  end for
16: end for

```

---

visitation distributional alignment step. The experimental results of the implementation of Algorithm 2 are shown in Figure 6.

The overall results in Figure 6 are similar to those in Figure 3. However, compared to the Algorithm 1, the training processes of Algorithm 2 are more unstable. Besides, it also incurs additional learning costs since we have to train one policy for each MDP instead of one policy for both MDPs.

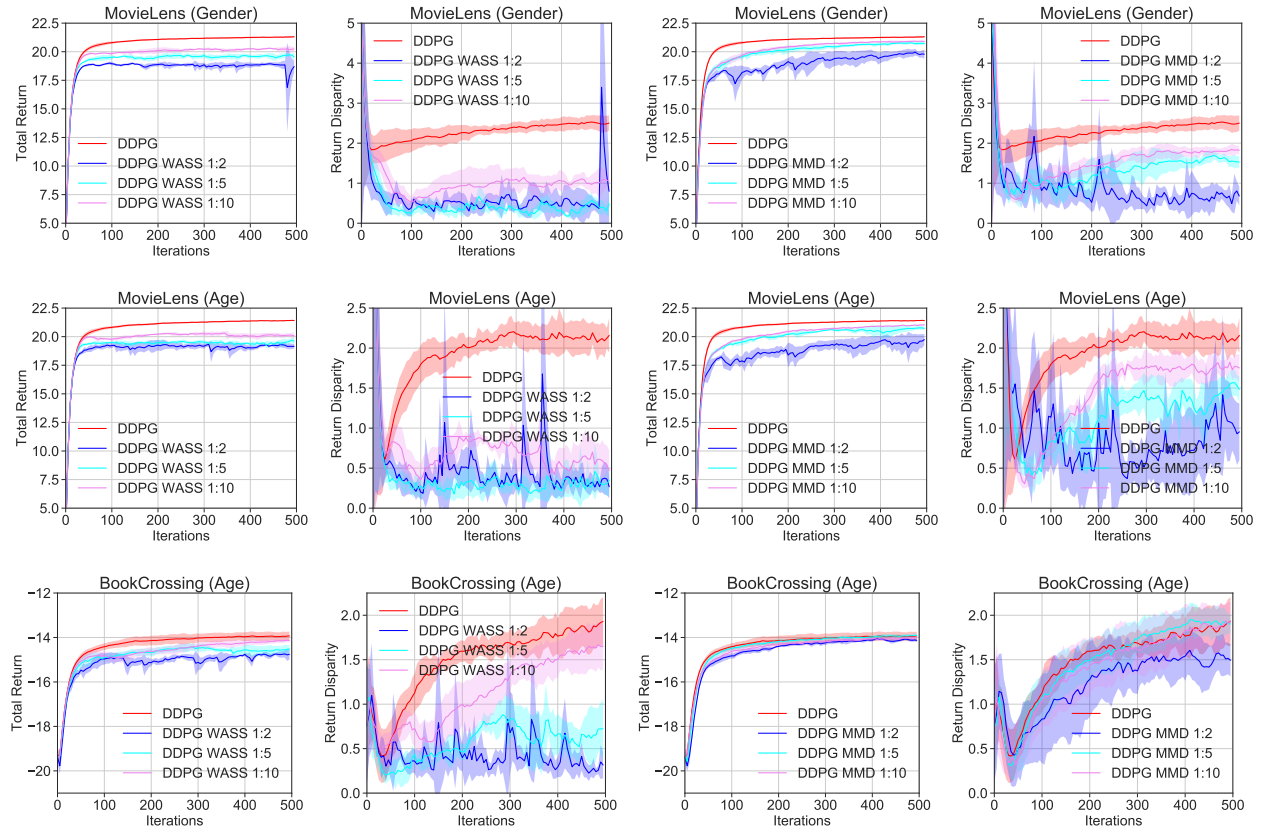


Figure 6: Learning curves of DDPG, DDPG-WASS and DDPG-MMD in three different settings.