Advances in Distributional Reinforcement Learning: Bridging Theory with Algorithmic Practice

by

Ke Sun

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Statistical Machine Learning

Department of Mathematical and Statistical Sciences
University of Alberta

Abstract

This thesis comprehensively investigates Distributional Reinforcement Learning (RL), a vibrant research field that interplays between statistics and RL. As an extension of classical RL, distributional RL, on the one hand, embraces plenty of statistical ideas by incorporating distributional learning, including density estimation and distribution divergence. At the same time, distributional RL involves frontier issues within the realm of RL, such as exploration, optimization, and uncertainty. In this thesis, we examine the benefits of being distributional in the context of RL by exploring the resulting theoretical advantages and properties, including regularization, optimization, and robustness against training noises. This investigation finally motivates the design of novel distributional RL algorithms.

In the first paper, we delve into the benefits of being categorical distributional in RL from the perspective of regularization. We attribute the potential superiority of distributional RL to a derived distribution-matching regularization by applying a return density function decomposition technique. This unexplored regularization in the distributional RL context is aimed at capturing additional return distribution knowledge regardless of only its expectation, contributing to an augmented reward signal in policy optimization. In the second paper, we further provide evidence of the benefits of distributional RL through the optimization lens. We demonstrate that the distribution loss of distributional RL has desirable smoothness characteristics and hence enjoys stable gradients. Furthermore, we show that distributional RL can perform

favorably if the return distribution approximation is appropriate, measured by the variance of gradient estimates in each environment. In the third paper, we study the training robustness of distributional RL by validating the contraction of distributional Bellman operators in the proposed State-Noisy Markov Decision Process (SN-MDP), a typical tabular case that incorporates both random and adversarial state observation noises. In the noisy setting with function approximation, we theoretically characterize the bounded gradient norm of distributional RL loss in terms of the state features, which interprets its better training robustness against state observation noises. In the last paper, we propose a novel distributional RL algorithm, called *Sinkhorn distributional RL (SinkhornDRL)*, which leverages Sinkhorn divergence—a regularized Wasserstein loss—to minimize the difference between current and target Bellman return distributions. Theoretically, we prove the contraction properties of SinkhornDRL, aligning with the interpolation nature of Sinkhorn divergence between Wasserstein distance and Maximum Mean Discrepancy (MMD).

In summary, these papers contribute to the theoretical understanding of the benefits of being fully distributional in RL compared with classical RL, which only focuses on the expectation of the return distribution. Along with our algorithm design, our work not only provides sufficient insights to guild practitioners for deploying distributional RL in real applications but also contributes to inspiring researchers from other relevant areas broadly in statistics, machine learning, operational research, and control.

Preface

The research presented in this thesis was conducted under the supervision of Dr. Linglong Kong, involving a series of collaborative research projects. The work described in Chapters 2 through 5 represents my original contributions to the field of Distributional Reinforcement Learning.

Chapter 2: Titled "*The Benefits of Being Categorical Distributional: Uncertainty-aware Regularized Exploration in Reinforcement Learning* (Ke Sun, Yingnan Zhao, Enze Shi, Yafei Wang, Xiaodong Yan, Bei Jiang, Linglong Kong), " this work was submitted to the Advances in Neural Information Processing Systems (NeurIPS) 2024. I developed the primary analytical framework, conducted experiments, and was responsible for writing the manuscript. Yingnan Zhao provided valuable experimental suggestions, while Enze Shi assisted in proofreading some of the theorems. The other co-authors contributed during the initial phase of the paper organization.

Chapter 3: Titled "*How Does Return Distribution in Distributional Reinforcement Learning Help Optimization?* (Ke Sun, Bei Jiang, Linglong Kong)," this paper was accepted at the International Conference on Machine Learning (ICML) 2024 workshop: Aligning Reinforcement Learning Experimentalists and Theorists. I was responsible for all theorem proofs, implementing the experiments, and writing the paper. Dr. Bei Jiang and Dr. Linglong Kong offered significant insights during the manuscript composition.

Chapter 4: Titled "*Exploring the Training Robustness of Distributional Reinforcement Learning against Noisy State Observations* (Ke Sun, Yingnan Zhao, Shangling Jui, Linglong Kong), " this paper was published in the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD), 2023. My role in this paper included proving theorem, implementing algorithms, and writing the paper. Yingnan Zhao collaborated closely on some implementation details, and Dr. Linglong Kong provided valuable feedback on paper organization and algorithm analysis.

Chapter 5: Titled "*Distributional Reinforcement Learning with Regular-*

ized Wasserstein Loss (Ke Sun, Yingnan Zhao, Wulong Liu, Bei Jiang, Linglong Kong)," this work was submitted to the Advances in Neural Information Processing Systems (NeurIPS) 2024. I led the algorithm design, theorems proofs, and manuscript writing. Yingnan Zhao assisted with the experiment execution, and Wulong Liu, Bei Jiang, and Linglong Kong provided constructive feedback that significantly enhanced the quality of the work.

This thesis not only reflects my main research accomplishments but also presents the collaborative spirit of the academic community, which considerably supports me in my doctoral study. I am profoundly grateful to all my collaborators for their insightful suggestions and feedback, which were instrumental in completing this work.

Acknowledgements

First and foremost, I extend my deepest gratitude to my supervisor, Dr. Linglong Kong, whose unwavering support and expert guidance have profoundly contributed to my doctoral study. He not only shaped the trajectory of my academic pursuits but also played an integral part in organizing my future research directions.

I am profoundly grateful to Dr. Chengchun Shi of the London School of Economics and Political Science (LSE), who not only invited me for a summer visit but also provided detailed guidance that was instrumental in my academic development. The opportunity to collaborate closely and conduct significant research under his mentorship has been invaluable.

I must express my appreciation to Dr. Richard Sutton, Dr. Martha White, and Dr. Csaba Szepesvari from the Department of Computing Science at the University of Alberta. Their courses laid the foundations of my understanding of reinforcement learning (RL) research, and the vibrant research environment they and other faculty fostered together has deeply influenced my lifelong research aspirations.

Special thanks to Dr. Bei Jiang from the University of Alberta for enhancing my research skills through her insightful courses and to Dr. Zoltán Szabó from LSE for sharing valuable research thoughts during my visit.

My appreciation also goes to my collaborators—Enze Shi, Jinhan Xie, Hongming Zhang, Yingnan Zhao, Chen Xi, Jun Jin, and Yangchen Pan. Their knowledge and innovative viewpoints have significantly broadened the scope of my doctoral research. I want to acknowledge my friends from both the University of Alberta and LSE, whose companionship provided moments of relief and laughter during the most stressful times.

I am also thankful to the University of Alberta for providing excellent academic resources and a supportive study environment to conduct my research. Special thanks to the staff and technical support team, whose assistance was indispensable in my development at the university.

Last but not least, I am immensely grateful to my family. Thanks to

my parents and my twin brother for their unwavering support throughout my studies; and especially to Jiayin Meng for her enduring love, who has consistently encouraged me to pursue my ambitions and supported me through all the ups and downs.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Background. In reinforcement learning (RL) [102], an agent seeks an optimal policy in a sequential decision-making process. Deep RL has recently achieved significant improvements in a variety of challenging artificial intelligence tasks, including game playing [74, 93, 69] and robotics navigation [72]. A flurry of state-of-the-art algorithms have been proposed, including Deep Q-Learning (DQN) [74] and variants such as Double-DQN [46], Dueling-DQN [107], Deep Deterministic Policy Gradient (DDPG) [59], Soft Actor-Critic [43] and Proximal Policy Optimization (PPO) [90], all of which have successfully solved end-to-end decision-making problems such as playing Atari games. The intrinsic characteristics of classical RL algorithms mentioned above are mainly based on the expectation of discounted cumulative rewards that an agent observes while interacting with the environment. In stark contrast to the expectation-based RL, a new branch of algorithms called *distributional RL* estimates the full distribution of total returns and has demonstrated the state-of-the-art performance in a wide range of environments [9, 22, 21, 115, 120, 77, 101]. Meanwhile, distributional RL also inherits other benefits in risk-sensitive control [21, 60, 18], offline learning [113, 68], policy exploration [70, 85], training robustness against state noises [99, 97], and optimization [98, 87, 55].

General Motivation. The idea of modeling the distribution beyond only the expectation of a random variable is rooted in the statistical inference in the statistical community. As the main target of interest in advanced sta-

tistical research, statistical inference emphasizes investigating the asymptotic distribution properties of the statistical estimates for the subsequent interval estimate and hypothesis testing. Similarly, distributional RL models the entire distribution of the return random variable, the target of interest, instead of only its expectation. The promising performance of distributional RL motivates us from the statistical community to study the underlying reasons and further design advanced theory-principled algorithms for effective deployment in broader applications.

Chapter 2. Despite the impressive empirical improvement of distributional RL, its theoretical advantages over classical RL are not yet fully understood. In the first paper, we dive deeper into this behavior difference, starting with categorical distributional RL (CDRL). The potential superiority of distributional RL may stem from a distribution-matching regularization in the objective function, decomposed by employing a return density function decomposition technique. This form of regularization aims to align with the uncertainties of target returns for the current return distribution estimates, fostering a novel exploration strategy. This uncertainty-aware regularized exploration differs from the standard entropy regularization in MaxEnt RL, which explicitly optimizes policies to promote exploration by encouraging diverse actions.

Chapter 3. Subsequently, in the second paper, we further explore the potential advantages of distributional RL through the optimization lens. The optimization benefits of being distributional arise from the leverage of additional return distribution information over classical RL, which we investigated in the Neural Fitted Z-Iteration (Neural FZI) framework. Initially, we establish that the loss function specific to distributional RL exhibits desirable smoothness properties, thereby facilitating stable gradients and contributing to enhanced optimization stability. Additionally, we unveil the acceleration effects of distributional RL, where we show that distributional RL can achieve superior performance when the return distribution approximation is accurate, as indicated by the variance of gradient estimates.

Chapter 4. In real-world scenarios, the state observations that agents encounter often include measurement inaccuracies or adversarial interferences, leading to suboptimal decision-making or even destabilizing training processes. In the third paper, we study the training robustness of distributional RL in the face of noisy state observations. We assess the robustness of distributional Bellman operators within the framework of State-Noisy Markov Decision Processes (SN-MDP) in a tabular context. In scenarios involving noisy states combined with function approximation, we attribute the robustness of distributional RL to its bounded gradient norm under the distributional loss, which enhances the overall training robustness of distributional RL.

Chapter 5. The choice of distribution divergence and the corresponding distribution representation considerably influences the effectiveness of distributional RL. In the last paper, we introduce Sinkhorn distributional RL, which utilizes Sinkhorn divergence—a form of regularized Wasserstein loss—to effectively minimize the discrepancy between the current and target Bellman return distributions. We provide a theoretical foundation for SinkhornDRL by demonstrating its contraction properties, aligning with the interpolation nature of Sinkhorn divergence between Wasserstein distance and Maximum Mean Discrepancy (MMD). Our comparative analysis sheds light on the behavioral nuances of SinkhornDRL relative to existing algorithms, offering a deeper understanding of its unique advantages and interactions within the broader framework of distributional RL methods.

Summary. Overall, these papers contribute to a deeper understanding of the potential benefits of being distributional in the context of RL, compared with classical RL, which only focuses on the expectation of return distribution in algorithm design. We explain this advantage from various perspectives, including regularization, optimization, and robustness, facilitating the broader development of distributional RL algorithms in real-world applications. Next, we propose a novel theory-principled distributional RL algorithm inspired by the optimal transport literature. This thesis significantly advances the development of distributional RL research.

Chapter 2

The Benefits of Being Categorical Distributional: Uncertainty-aware Regularized Exploration in Reinforcement Learning

2.1 Abstract

The theoretical advantages of distributional reinforcement learning (RL) over classical RL remain elusive despite its remarkable empirical performance. Starting from Categorical Distributional RL (CDRL), we attribute the potential superiority of distributional RL to a derived distribution-matching regularization by applying a return density function decomposition technique. This unexplored regularization in the distributional RL context is aimed at capturing additional return distribution knowledge regardless of only its expectation, contributing to an augmented reward signal in policy optimization. Compared with the standard entropy regularization in MaxEnt RL that explicitly optimizes the policy to encourage exploration, the derived regularization from CDRL implicitly updates policies guided by the new reward signal. Introduc-

ing this regularization helps to align with the uncertainty of target returns, leading to an uncertainty-aware exploration effect. Finally, extensive experiments substantiate the importance of this uncertainty-aware regularization in distributional RL on the empirical benefits over classical RL.

2.2 Introduction

Motivation: Interpreting the Benefits of Being (Categorical) Distributional in RL. Despite various distributional RL algorithms that have achieved remarkable empirical success, we still have a limited understanding of what the advantages of distributional RL stem from, particularly in the general function approximation setting. Early work [66] showed that in many realizations of tabular and linear approximation settings, distributional RL behaves similarly to classic RL and the benefits of distributional RL may mainly be attributed to non-linear approximation setting. While their findings offer profound insights, their analysis, based on a coupled updates method, overlooks several elements, such as the optimization effect for the different losses. The statistical benefits of quantile temporal difference (TD) used in quantile distributional RL algorithms, such as QR-DQN [22], were revealed in [86, 87], potentially leading to variance reduction properties. The theoretical properties of CDRL were first revealed in [84]; however, the empirical superiority of CDRL or being categorical distributional is not yet well understood. Recent work [113, 105] explained the benefits of distributional RL from the perspective of the small-loss and second-order PAC bounds. However, their results are primarily based on low-rank MDPs or offline RL, which may not be directly applicable to online RL with the general function approximation.

Contributions. In this paper, we investigate the underlying reasons behind the potential benefits of distributional RL over classical RL starting from CDRL, the first successful distributional RL family. We examine these benefits through the lens of regularization and exploration effects, offering a dramatically different perspective relative to existing works. Firstly, we decompose the objective function of CDRL into an expectation-based term and

a distribution-matching regularization via *return density decomposition technique*. The resulting regularization serves as an augmented reward in the actor-critic framework, encouraging the policies to explore states and actions whose current return distribution estimate lags far behind the target one determined by the environment. This leads to an uncertainty-aware exploration effect in contrast to the exploration for diverse actions in MaxEnt RL. Meanwhile, we propose a theoretically principled algorithm called *Distribution-Entropy-Regularized Actor Critic* accordingly, interpolating between expectation-based and distributional RL. Empirical results demonstrate the crucial role of the uncertainty-aware entropy regularization from CDRL in its empirical success over expectation-based RL on both Atari games and MuJoCo environments. We also demonstrate the distinct roles that the uncertainty-aware entropy in distributional RL and the explicit vanilla entropy in MaxEnt RL play by exploring their mutual impacts, providing more potential research directions in the future. Our contributions are summarized as follows:

- We propose a return density decomposition technique to decompose the objective function in CDRL, yielding an uncertainty-aware regularization. This derived regularization is thus used to interpret the benefits of being categorical distributional in RL over expectation-based RL.

- We incorporate the uncertainty-aware regularization into the actor-critic framework, thereby encouraging uncertainty-aware exploration when compared with MaxEnt RL. We also propose a theoretically grounded actor-critic algorithm, interpolating between classical and distributional RL.

- Empirically, we verify the effect of the decomposed uncertainty-aware regularization on the advantage of distributional RL and explore the mutual impacts of two types of regularization.

Outline. In Section 2.4, we provide the background knowledge of (categorical) distributional RL. We begin by revealing the uncertainty-aware regularization effect in value-based CDRL in Section 2.5, and further specifically study this implicit regularization into the policy gradient framework to directly

compare it with MaxEnt RL in Section 2.6. Extensive experiments demonstrate the uncertainty-aware regularization of distributional RL and its mutual impact with entropy regularization in Section 2.7.

2.3 Related Work

Distributional Learning via Categorical Representation. Categorical learning has been widely employed, with advantages in representation [78, 52] and optimization [51, 98]. Its empirical superiority has increasingly gained attention in various RL tasks [29], within the broader category of CDRL. The perspective of uncertainty-aware regularization-based exploration that our research introduces adds a significant theoretical understanding of the benefits of being categorical distributional in RL.

Exploration in RL in the Entropy Principle. As a general and effective mechanism, the entropy principle has been extensively studied to enhance the exploration in RL, which aims to explore more diverse actions. Classical algorithms are established upon the maximum entropy RL framework [111], including soft Q-learning [42], Soft Actor Critic (SAC) [43] and variants [44]. To leverage the knowledge in the learned return distribution to promote the exploration, existing works include [70] that utilizes the variance, and [56] in the ensemble way. By contrast, we decompose return distributions from CDRL and the derived regularization encourages a distinct uncertainty-aware exploration driven by the discrepancy between the agent's uncertain estimate and the environment.

2.4 Preliminaries

Markov Decision Process (MDP) and Classical RL. An environment is often modeled via an Markov Decision Process $(\mathcal{S}, \mathcal{A}, \mathcal{R}, P, \gamma)$, with a set of states $\mathcal{S}$ and actions $\mathcal{A}$, the transition kernel $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, the bounded reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \Delta([R_{\min}, R_{\max}])$, and a discounted factor $\gamma \in [0, 1]$. We denote the reward the agent receives at time t as $r(s_t, a_t) \sim \mathcal{R}(s_t, a_t)$.

Given a policy π, classical RL focuses on estimating the expectation of the return, i.e., the Q function: $Q^\pi(s,a) = \mathbb{E}_\pi\left[\sum_{t=0}^{+\infty}\gamma^t r_t | s_0 = s, a_0 = a\right]$. We also define Bellman evaluation operator $\mathcal{T}^\pi$ and Bellman optimality operator $\mathcal{T}^{\text{opt}}$: $\mathcal{T}^\pi Q(s,a) = \mathbb{E}[R(s,a)] + \gamma \mathbb{E}_{s'\sim P, a'\sim\pi}\left[Q\left(s',a'\right)\right]$ and $\mathcal{T}^{\text{opt}} Q(s,a) = \mathbb{E}[R(s,a)] + \gamma \max_{a'} \mathbb{E}_{s'\sim P}\left[Q\left(s',a'\right)\right]$.

Distributional RL and CDRL. Instead of only estimating the expectation for classical RL, distributional RL models the full distribution of the return $Z^\pi(s,a) = \sum_{t=0}^{\infty}\gamma^t r_t | s_0 = s, a_0 = a$. The return distribution $\eta^\pi : \mathcal{S}\times\mathcal{A} \to \Delta(\mathbb{R})$ is defined as $\eta^\pi(s,a) = \mathcal{D}(Z^\pi(s,a))$, where $\mathcal{D}$ extracts the distribution of the return random variable. We call the density function of $Z^\pi(s,a)$ as *action-state return density function*. $\eta^\pi(s,a)$ is updated via the distributional Bellman operator $\mathfrak{T}^\pi$, defined by

$$\mathfrak{T}^\pi Z(s,a) \overset{D}{=} R(s,a) + \gamma Z\left(s',a'\right), \tag{2.1}$$

where $\overset{D}{=}$ implies random variables of both sides are equal in distribution. CDRL is the first successful distributional RL algorithm family that approximates the return distribution by a discrete categorical distribution $\widehat{\eta}^\pi = \sum_{i=1}^N p_i \delta_{z_i}$, where $\{z_i\}_{i=1}^N$ is a set of fixed supports and $\{p_i\}_{i=1}^N$ are learnable probabilities. The leverage of a heuristic projection operator $\Pi_{\mathcal{C}}$ (see Appendix 2.9.1 for more details) and the Kullback–Leibler (KL) divergence guarantee the theoretical convergence of CDRL under Cramér distance or Wasserstein distance [84].

2.5 Uncertainty-aware Regularization in Value-based Distribution RL

2.5.1 Distributional RL: Neural FZI

Classical RL: Neural Fitted Q-Iteration (Neural FQI). Neural FQI [28, 83] offers a statistical explanation of DQN [74], capturing its key features, including experience replay and the target network Q_{θ^*}. In Neural FQI, we

update a parameterized Q_θ in each iteration k in a regression:

$$Q_\theta^{k+1} = \underset{Q_\theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \left[y_i^k - Q_\theta \left(s_i, a_i \right) \right]^2, \tag{2.2}$$

where the target $y_i^k = r(s_i, a_i) + \gamma \max_{a \in \mathcal{A}} Q_{\theta^*}^k \left(s_i', a \right)$ is fixed within every T_{target} steps to update target network Q_{θ^*} by letting $Q_{\theta^*}^{k+1} = Q_\theta^{k+1}$. The experience buffer induces independent samples $\{(s_i, a_i, r_i, s_i')\}_{i \in [n]}$. If $\{Q_\theta : \theta \in \Theta\}$ is sufficiently large such that it contains $\mathcal{T}^{\text{opt}} Q_{\theta^*}^k$, Eq. 2.2 has solution $Q_\theta^{k+1} = \mathcal{T}^{\text{opt}} Q_{\theta^*}^k$, which is exactly the updating rule under Bellman optimality operator [28]. From the viewpoint of statistics, the optimization problem in Eq. 2.2 in each iteration is a standard supervised and neural network parameterized regression regarding Q_θ.

Distributional RL: Neural Fitted Z-Iteration (Neural FZI). While our analysis is not intended to involve properties of neural networks, we interpret distributional RL as Neural FZI as it is by far closest to the practical algorithms. Analogous to Neural FQI, we simplify value-based distributional RL algorithms denoted by the parameterized Z_θ into Neural FZI, which is formulated as

$$Z_\theta^{k+1} = \underset{Z_\theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} d_p(Y_i^k, Z_\theta \left(s_i, a_i \right)), \tag{2.3}$$

where we denote the target random variable $Y_i^k = R(s_i, a_i) + \gamma Z_{\theta^*}^k \left(s_i', \pi_Z(s_i') \right)$ with the policy π_Z following the greedy rule $\pi_Z(s_i') = \operatorname{argmax}_{a'} \mathbb{E} \left[Z_{\theta^*}^k(s_i', a') \right]$. The target Y_i^k is fixed within every T_{target} steps to update target network Z_{θ^*}. d_p is a distribution divergence between two distributions and the lower cases of random variables s_i' and $\pi_Z(s_i')$ are given for convenience in notations.

2.5.2 Distributional RL: Entropy-regularized Neural FQI

Return Density Decomposition. To separate the impact of additional distribution information from the expectation of Z^π, we use a variant of *gross error model* from robust statistics [49], which was also similarly used to analyze Label Smoothing [76] and Knowledge Distillation [47]. Akin to the categorical

representation in CDRL [22], we utilize a *histogram function estimator* $\widehat{p}^{s,a}(x)$ with N bins to approximate an arbitrary continuous true density $p^{s,a}(x)$ of $Z^\pi(s,a)$, given a state s and action a. We leverage the continuous histogram estimator rather than the discrete categorical parameterization for richer analysis. Given a fixed set of supports $l_0 \leq l_1 \leq ... \leq l_N$ with the equal bin size as Δ, $\Delta_i = [l_{i-1}, l_i)$, $i = 1, ..., N-1$ with $\Delta_N = [l_{N-1}, l_N]$, the histogram density estimator is $\widehat{p}^{s,a}(x) = \sum_{i=1}^N p_i \mathbb{1}(x \in \Delta_i)/\Delta$ with p_i as the coefficient in the i-th bin. Denote Δ_E as the interval that $\mathbb{E}[Z^\pi(s,a)]$ falls into, i.e., $\mathbb{E}[Z^\pi(s,a)] \in \Delta_E$. Putting all together, we have an action-state return density decomposition over the histogram density estimator $\widehat{p}^{s,a}(x)$:

$$\widehat{p}^{s,a}(x) = (1-\epsilon)\mathbb{1}(x \in \Delta_E)/\Delta + \epsilon \widehat{\mu}^{s,a}(x) \tag{2.4}$$

where $\widehat{p}^{s,a}$ is decomposed into a single-bin histogram $\mathbb{1}(x \in \Delta_E)/\Delta$ with all mass on Δ_E and an *induced* histogram density function $\widehat{\mu}^{s,a}$ evaluated by $\widehat{\mu}^{s,a}(x) = \sum_{i=1}^N p_i^\mu \mathbb{1}(x \in \Delta_i)/\Delta$ with p_i^μ as the coefficient of the i-th bin. ϵ is a pre-specified hyper-parameter before the decomposition, controlling the proportion between $\mathbb{1}(x \in \Delta_E)/\Delta$ and $\widehat{\mu}^{s,a}(x)$. More specifically, the induced histogram $\widehat{\mu}^{s,a}$ in the second term is the difference between the considered histogram $\widehat{p}^{s,a}$ and a single-bin histogram, aiming at characterizing the impact of action-state return distribution **despite its expectation** $\mathbb{E}[Z^\pi(s,a)]$ on the performance of distributional RL. We first demonstrate that $\widehat{\mu}^{s,a}$ is a valid probability density function under certain ϵ in Proposition 1.

Proposition 1. *(Decomposition Validity) Denote* $\widehat{p}^{s,a}(x \in \Delta_E) = p_E/\Delta$ *with* p_E *as the coefficient on the bin* Δ_E. $\widehat{\mu}^{s,a}(x) = \sum_{i=1}^N p_i^\mu \mathbb{1}(x \in \Delta_i)/\Delta$ *is a valid density function if and only if* $\epsilon \geq 1 - p_E$.

The proof can be found in Appendix 2.9.2. Proposition 1 demonstrates that the return density decomposition is valid when the pre-specified hyper-parameter ϵ satisfies $\epsilon \geq 1 - p_E$. Under this condition, our analysis maintains the standard categorical distributional framework in distributional RL.

Equivalence between Histogram Density Estimator and Categorical Representation. The histogram function is a continuous estimator in con-

trast to the discrete nature of categorical parameterization. We show that they are equivalent in distributional RL in Appendix 2.9.3. As a supplementary analysis, with attribution to [108], we also discuss necessary theoretical underpinnings of the histogram density estimator in the context of distributional RL in Appendix 2.9.4.

Distributional RL: Entropy-regularized Neural FQI. We apply the decomposition on the target action-value histogram density function and choose KL divergence as d_p in Neural FZI. Let $\mathcal{H}(U, V)$ be the cross-entropy between two probability measures U and V, i.e., $\mathcal{H}(U, V) = -\int_{x\in\mathcal{X}} U(x) \log V(x)\, \mathrm{d}x$. The target histogram density function $\widehat{p}^{s,a}$ is decomposed as $\widehat{p}^{s,a}(x) = (1 - \epsilon)\mathbb{1}(x \in \Delta_E)/\Delta + \epsilon\widehat{\mu}^{s,a}(x)$. We derive the following entropy-regularized form for distributional RL in Proposition 2 with the proof provided in Appendix 2.9.6.

Proposition 2. *(Decomposed Neural FZI) Denote $q_\theta^{s,a}(x)$ as the histogram estimator of $Z_\theta^k(s, a)$ in Neural FZI. Based on Eq. 2.4 and the KL divergence as d_p, Neural FZI in Eq. 2.3 is simplified as*

$$Z_\theta^{k+1} = \operatorname*{argmin}_{q_\theta} \frac{1}{n} \sum_{i=1}^{n} [\underbrace{-\log q_\theta^{s_i,a_i}(\Delta_E^i)}_{(a)} + \alpha\mathcal{H}(\widehat{\mu}^{s_i',\pi_Z(s_i')}, q_\theta^{s_i,a_i})], \tag{2.5}$$

where Δ_E^i represents the interval that the expectation of the target random variable $R(s_i, a_i)+\gamma Z_{\theta^*}^k\,(s_i', \pi_Z(s_i'))$ falls into, i.e., $\mathbb{E}\left[R(s_i, a_i) + \gamma Z_{\theta^*}^k\,(s_i', \pi_Z(s_i'))\right] \in \Delta_E^i$. $\alpha = \varepsilon/(1-\varepsilon) > 0$ and $\widehat{\mu}^{s_i',\pi_Z(s_i')}$ is the induced histogram density function by decomposing the histogram density of $R(s_i, a_i) + \gamma Z_{\theta^*}^k\,(s_i', \pi_Z(s_i'))$.

In Proposition 3, we further demonstrate that minimizing the term (a) in Eq. 2.5 is equivalent to minimizing Neural FQI in terms of the minimizers. As such, the regularization term $\alpha\mathcal{H}(\widehat{\mu}^{s_i',\pi_Z(s_i')}, q_\theta^{s_i,a_i})$ interprets the potential benefits of CDRL over classical RL. For the uniformity of notation, we still use s, a in the following analysis instead of s_i, a_i.

Proposition 3. *(Equivalence between **the term (a)** in Decomposed Neural FZI and Neural FQI) In Eq. 2.5 of Neural FZI, assume the function class*

$\{Z_\theta : \theta \in \Theta\}$ is sufficiently large such that it contains the target $\{Y_i^k\}_{i=1}^n$ for all k, when $\Delta \to 0$, minimizing ***the term (a)*** *in Eq. 2.5 implies*

$$P(Z_\theta^{k+1}(s,a) = \mathcal{T}^{opt} Q_{\theta^*}^k(s,a)) = 1, \tag{2.6}$$

where $\mathcal{T}^{opt} Q_{\theta^*}^k(s,a)$ is the scalar-valued target in the k-th phase of Neural FQI.

See Appendix 2.9.7 for the detailed proof. Proposition 3 demonstrates that as $\Delta \to 0$, the random variable $Z_\theta^{k+1}(s,a)$ with the limiting distribution in Neural FZI (distributional RL) will *degrade* to a constant $\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)$, the minimizer (scalar-valued target) in Neural FQI (classical RL). That being said, *minimizing the term (a) in Neural FZI is equivalent to minimizing Neural FQI with the same limiting minimizer*. Please refer to Appendix 2.9.7 for more results about the convergence rate $o(\Delta)$ in distribution. With the underlying link between optimizing the term (a) of Neural FZI with Neural FQI established in Proposition 3, we can leverage the regularization term $\alpha\mathcal{H}(\widehat{\mu}^{s_i', \pi_Z(s_i')}, q_\theta^{s_i, a_i})$ to interpret the potential superiority of CDRL over classical RL. The assumption that $\{Z_\theta : \theta \in \Theta\}$ is sufficiently large such that it contains $\{Y_i^k\}_{i=1}^n$ implies good in-distribution generalization performance in each phase of Neural FZI, which is commonly used in the context of distributional RL to derive tractable theoretical results, such as [113]. Meanwhile, this connection with classic RL is also consistent with the mean-preserving property in classical RL [84]. Next, we are ready to elaborate on the impact of this regularization for Neural FZI (distributional RL).

2.5.3 Uncertainty-aware Regularized Exploration

Based on the equivalence between the term (a) of decomposed Neural FZI and FQI, the behavior difference of distributional RL compared with expectation-based RL can be attributed to the second regularization term $\mathcal{H}(\widehat{\mu}^{s_i', \pi_Z(s_i')}, q_\theta^{s_i, a_i})$. Minimizing Neural FZI pushes $q_\theta^{s,a}$ for the current return density estimator to catch up with the target return density function of $\widehat{\mu}^{s_i', \pi_Z(s_i')}$, which additionally incorporates the uncertainty of return distribution in the whole learning process instead of only encoding its expectation. Since it is a prevalent notion

that distributional RL can significantly reduce intrinsic uncertainty of the environment [70, 21], the derived distribution-matching regularization term helps the learning algorithms to capture more uncertainty of the environment by modeling the whole return distribution instead of only its expectation, leading to *an uncertainty-aware regularized exploration effect*.

Approximation of $\widehat{\mu}^{s',\pi_Z(s')}$. As in practical distributional RL algorithms, we typically use temporal-difference (TD) learning to attain the target probability density estimate $\widehat{\mu}^{s',\pi_Z(s')}$ based on Eq. 2.4, provided $\mathbb{E}\left[Z(s,a)\right]$ exists and $\epsilon \geq 1 - p_E$ in Proposition 1. The approximation error of $\widehat{\mu}^{s',\pi_Z(s')}$ is fundamentally determined by the TD learning nature. We also discuss the usage of KL divergence in distributional RL in Appendix 2.9.5.

2.6 Uncertainty-aware Regularized Exploration in Actor Critic Framework

In this section, we further investigate the uncertainty-aware regularization and its exploration effect in the actor-critic framework by comparing it with MaxEnt RL.

2.6.1 Connection with MaxEnt RL

Motivation for the Connection. The maximum entropy regularization is commonly used in RL, which has various conceptual and practical advantages. Firstly, the learned policy is encouraged to visit states with high entropy in the future, promoting the exploration of diverse actions [44, 43, 111]. It also considerably improves the learning speed [71] and therefore is widely employed in state-of-the-art algorithms, e.g., Soft Actor-Critic (SAC) [43]. Similar empirical benefits of both distributional RL and MaxEnt RL motivate us to probe their underlying connection.

Explicit Entropy Regularization in MaxEnt RL. MaxEnt RL [111] *explicitly* encourages the exploration by optimizing for policies to reach states

with higher entropy in the future:

$$J(\pi) = \sum_{t=0}^{T} \mathbb{E}_{(s_t,a_t)\sim\rho_\pi} \left[r\left(s_t, a_t\right) + \beta \mathcal{H}(\pi(\cdot|s_t)) \right], \tag{2.7}$$

where $\mathcal{H}\left(\pi_\theta\left(\cdot|s_t\right)\right) = -\sum_a \pi_\theta\left(a|s_t\right) \log \pi_\theta\left(a|s_t\right)$ and ρ_π is the generated distribution following π. The temperature parameter β determines the relative importance of the entropy term against the cumulative rewards and thus controls the action diversity of the optimal policy learned via Eq. 2.7.

Implicit Entropy Regularization in Distributional RL. For a direct comparison with MaxEnt RL, it is required to specifically analyze the impact of the regularization term in Eq. 2.5. Consequently, we incorporate the distribution-matching regularization of distributional RL into the Actor Critic (AC) framework akin to MaxEnt RL, enabling us to consider a new soft Q-value. The new Q function can be computed iteratively by applying a modified Bellman operator denoted as $\mathcal{T}_d^\pi$, called *Distribution-Entropy-Regularized Bellman Operator*. Given a fixed q_θ, $\mathcal{T}_d^\pi$ is defined as

$$\mathcal{T}_d^\pi Q\left(s_t, a_t\right) \triangleq r\left(s_t, a_t\right) + \gamma \mathbb{E}_{s_{t+1}\sim P(\cdot|s_t,a_t)} \left[V\left(s_{t+1}|s_t, a_t\right) \right], \tag{2.8}$$

where a new soft value function $V\left(s_{t+1}|s_t, a_t\right)$ conditioned on s_t, a_t is defined by

$$V\left(s_{t+1}|s_t, a_t\right) = \mathbb{E}_{a_{t+1}\sim\pi} \left[Q\left(s_{t+1}, a_{t+1}\right) \right] + f(\mathcal{H}\left(\mu^{s_t,a_t}, q_\theta^{s_t,a_t}\right)), \tag{2.9}$$

where f is a continuous increasing function over the cross-entropy $\mathcal{H}$. μ^{s_t,a_t} is the induced true target return histogram density function via the decomposition in Eq. 2.4 regardless of its expectation, which can be approximated via bootstrap estimate $\widehat{\mu}^{s_{t+1},\pi_Z(s_{t+1})}$ similar in Eq. 2.5. In this specific tabular setting regarding s_t, a_t, we particularly use $q_\theta^{s_t,a_t}$ to approximate the true density function of $Z(s_t, a_t)$. The f transformation over the cross-entropy $\mathcal{H}$ between μ^{s_t,a_t} and $q_\theta^{s_t,a_t}(x)$ serves as the uncertainty-aware entropy regularization that we implicitly derive from value-based distributional RL in Section 2.5.2.

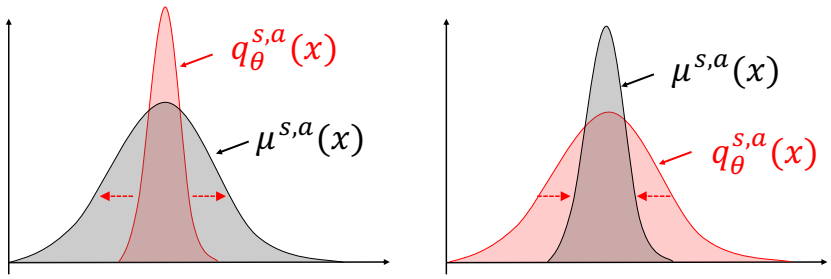


Figure 2.1: Uncertainty-aware distribution-matching regularization in CDRL to capture the intrinsic uncertainty of the environment. $q_\theta^{s,a}$ is forced to disperse (left) or concentrate (right) to align with the target return distribution.

By optimizing the value-based critic component in Actor-Critic, i.e., q_θ, this regularization reduces the mismatch between the target return distribution and current estimate, aligning with the regularization effect analyzed in Section 2.5.3. As illustrated in Figure 2.1, $q_\theta^{s,a}$ is optimized to catch up with the uncertainty of the target return distribution of $\mu^{s,a}$, expanding the knowledge of algorithms about the environment uncertainty for more informative decisions. Next, we elaborate on its additional impact on policy learning in the actor-critic in contrast to MaxEnt RL.

Reward Augmentation for Policy Learning. As opposed to the vanilla entropy regularization in MaxEnt RL that explicitly encourages the policy to explore, our derived distribution-matching regularization in distributional RL plays a role of **reward augmentation** for policy learning. The augmented reward incorporates additional return distribution knowledge in the learning process compared with expectation-based RL. As we will show later, *the augmented reward encourages policies to reach states s_t with actions $a_t \sim \pi(\cdot|s_t)$, whose current action-state return distribution $q_\theta^{s_t,a_t}$* ***lags far behind*** *the target one, measured by the magnitude of cross entropy.*

For a comprehensive analysis and a detailed comparison with MaxEnt RL, we now concentrate on the properties of our distribution-matching regularization in the Actor Critic (AC) framework. In Lemma 1, we first show that our Distribution-Entropy-Regularized Bellman operator $\mathcal{T}_d^\pi$ still inherits the convergence property in the policy evaluation phase with a cumulative augmented

reward function as the new objective function $J'(\pi)$.

Lemma 1. *(Distribution-Entropy-Regularized Policy Evaluation) Consider the distribution-entropy-regularized Bellman operator $\mathcal{T}_d^\pi$ in Eq. 2.8 and assume $\mathcal{H}(\mu^{s_t,a_t}, q_\theta^{s_t,a_t})$ is bounded for all $(s_t, a_t) \in \mathcal{S} \times \mathcal{A}$. We define $Q^{k+1} = \mathcal{T}_d^\pi Q^k$. Given q_θ, Q^{k+1} will converge to a corrected Q-value of π as $k \to \infty$ with the new objective function $J'(\pi)$ defined as*

$$J'(\pi) = \sum_{t=0}^{T} \mathbb{E}_{(s_t,a_t)\sim\rho_\pi} \left[r\left(s_t, a_t\right) + \gamma f(\mathcal{H}\left(\mu^{s_t,a_t}, q_\theta^{s_t,a_t}\right)) \right]. \tag{2.10}$$

We remain the updating rule $\pi_{\text{new}} = \arg\max_{\pi' \in \Pi} \mathbb{E}_{a_t \sim \pi'} \left[Q^{\pi_{\text{old}}}(s_t, a_t) \right]$ in the policy improvement phase. Next, we can immediately derive a new policy iteration algorithm, called *Distribution-Entropy-Regularized Policy Iteration (DERPI)* that alternates between the policy evaluation in Eq. 2.8 and the policy improvement. It will provably converge to a policy regularized by the distribution-matching term as shown in Theorem 1.

Theorem 1. *(Distribution-Entropy-Regularized Policy Iteration) Repeatedly applying distribution-entropy-regularized policy evaluation in Eq. 2.8 and the policy improvement, the policy converges to an optimal policy π^* such that $Q^{\pi^*}\left(s_t, a_t\right) \geq Q^\pi\left(s_t, a_t\right)$ for all $\pi \in \Pi$.*

Please refer to Appendix 2.9.8 for the proof of Lemma 1 and Theorem 1. Theorem 1 demonstrates that if we incorporate the distribution-matching regularization into the policy gradient framework in Eq. 2.10, we can design a variant of "soft policy iteration" [43] that can guarantee the convergence to an optimal policy given any fixed q_θ. Putting all the analyses above together, we comprehensively compare the regularization and exploration effect between MaxEnt RL and distributional RL (CDRL).

Uncertainty-aware Regularized Exploration in CDRL Compared with MaxEnt RL. For the objective function $J(\pi)$ in Eq. 2.7 of MaxEnt RL, the state-wise entropy $\mathcal{H}(\pi(\cdot|s_t))$ is maximized explicitly *w.r.t.* π for policies with a higher entropy in terms of diverse actions to encourage an explicit exploration.

For the objective function $J'(\pi)$ in Eq. 2.10 of distributional RL, the policy π is implicitly optimized through **the action selection** $a_t \sim \pi(\cdot|s_t)$ **mechanism** guided by an augmented reward signal from the distribution-matching regularization $f(\mathcal{H}\left(\mu^{s_t,a_t}, q_\theta^{s_t,a_t}\right))$. Concretely, the learned policy is encouraged to visit state s_t along with the policy-determined action via $a_t \sim \pi(\cdot|s_t)$, whose current action-state return distributions $q_\theta^{s_t,a_t}$ *lag far behind* the target return distributions. This discrepancy is measured by the magnitude of the cross entropy between two distributions. A large discrepancy indicates the uncertainty of current return distribution is considerably mis-estimated for considered states, results in an uncertainty-aware exploration against these states. As such, the policy learning will be additionally driven by the uncertainty difference between the current return distribution estimate and the target one. This leads to a distinct exploration strategy compared with MaxEnt RL that directly promotes diverse actions.

Interplay of Uncertainty-aware Regularization in Distributional Actor-Critic. Putting the critic and actor learning together in distributional RL, we reveal their interplay impact of the uncertainty-aware regularized exploration as opposed to expectation-based RL: 1) the actor (policy) learning seeks states and actions whose current return distribution estimate lags far behind the true one from the environment (approximated by the TD target distribution of $\mu^{s,a}$), 2) on the other hand, the critic learning reduces the return distribution mismatch on the explored states and actions between the current return distribution estimate and the true one determined by the environment, interpreting the benefits of CDRL over expectation-based RL.

2.6.2 DERAC Algorithm: Interpolating AC and Distributional AC

With the convergence guarantee of DERPI given a fixed q_θ, we also need to optimize q_θ within the actor-critic framework in the function approximation setting. Different from SAC that introduces another value function network, we only parameterize the return distribution $q_\theta(s_t, a_t)$ and the policy $\pi_\phi(a_t|s_t)$,

where we use $\mathbb{E}\left[q_\theta\right]$ to represent the Q function without parameterizing it again. Remarkably, the resulting *Distribution-Entropy-Regularized Actor-Critic (DERAC)* algorithm can interpolate expectation-based AC and distributional AC.

Optimize the critic by q_θ. The new value function $\hat{J}_q(\theta)$ is originally trained to minimize the squared residual error of Eq. 2.8. We show that $\hat{J}_q(\theta)$ can be simplified as:

$$\hat{J}_q(\theta) \propto (1-\lambda)\mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi \mathbb{E}\left[q_{\theta^*}(s,a)\right] - \mathbb{E}\left[q_\theta(s,a)\right]\right)^2\right] + \lambda \mathbb{E}_{s,a}\left[\mathcal{H}(\mu^{s,a}, q_\theta^{s,a})\right], \tag{2.11}$$

where we use a particular increasing function $f(\mathcal{H}) = (\tau\mathcal{H})^{\frac{1}{2}}/\gamma$ and $\lambda = \frac{\tau}{1+\tau} \in [0,1], \tau \geq 0$ is the hyperparameter that controls the uncertainty-aware regularization effect. The proof is given in Appendix 2.9.9. Interestingly, when we leverage the whole target density function $\widehat{p}^{s,a}$ to approximate the true return distribution of $\mu^{s,a}$, the objective function in Eq. 2.11 can be viewed as an exact interpolation of loss functions between expectation-based AC (the first term) and categorical distributional AC loss (the second term) [67]. In our implementation, for the target $\mathcal{T}^\pi \mathbb{E}\left[q_{\theta^*}(s,a)\right]$, we use the target return distribution neural network q_{θ^*} to stabilize the training, which is consistent with the Neural FZI framework analyzed in Section 2.5.1.

Optimize the policy π_ϕ. We optimize π_ϕ in the policy optimization based on the Q-function and therefore the new objective function $\hat{J}_\pi(\phi)$ can be expressed as $\hat{J}_\pi(\phi) = \mathbb{E}_{s,a\sim\pi_\phi}\left[\mathbb{E}\left[q_\theta(s,a)\right]\right]$. The complete DERAC algorithm is presented in Algorithm 1 of Appendix 2.9.11.

2.7 Experiments

In Section 2.7.1 of our experiments, we first verify the uncertainty-aware regularization effect of being categorical distribution in RL by applying the return density decomposition in Eq. 2.4 with different ϵ. In Section 2.7.2, we examine the interpolation performance of the proposed DERAC algorithm in continuous control environments, particularly interpreting the potential advantage of

DERAC that can mitigate the over-exploration of CDRL by pure categorical learning. Finally, we explore the mutual impacts between the vanilla entropy regularization in MaxEnt RL and the uncertainty-aware one from CDRL in Section 2.7.3, with a slight extension to quantile-based distributional RL, e.g., Implicit Quantile Networks (IQN) [21]. More implementation details are provided in Appendix 2.9.10.

2.7.1 Uncertainty-aware Regularization Effect by Return Density Decomposition

We demonstrate the decomposed uncertainty-aware entropy regularization analyzed in Eq. 2.5 though the return density function decomposition in Eq. 2.4 plays a crucial role in interpreting the benefits of CDRL over classical RL. Our experiments are conducted on both typical Atari games and Mujoco environments. Particularly, for the categorical distributional loss in C51 or the critic loss in the actor-critic algorithms, we replace the whole target categorical distribution $\widehat{p}^{s,a}(x)$ with the derived $\widehat{\mu}^{s,a}(x)$ decomposed under different ε based on Eq. 2.4. We then employ $\widehat{\mu}^{s,a}(x)$ instead of $\widehat{p}^{s,a}(x)$ to construct the KL divergence, leading to decomposed algorithms denoted by $\mathcal{H}(\mu, q_\theta)$. This decomposed algorithm enables us to assess the uncertainty-aware regularization

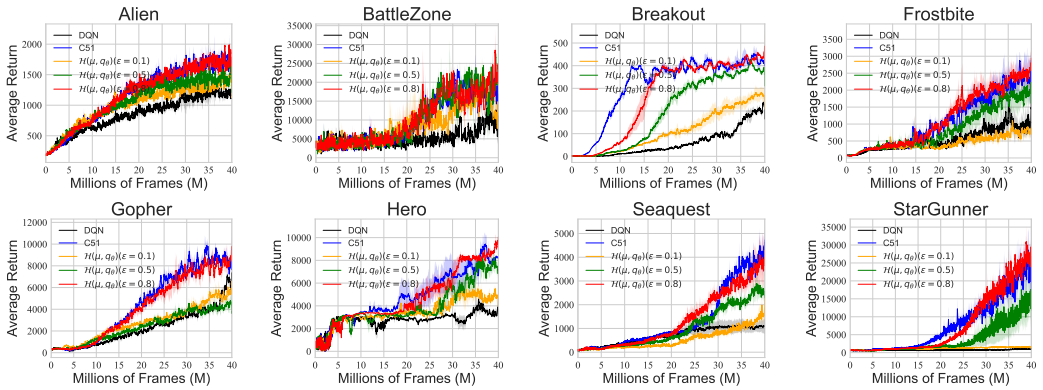


Figure 2.2: Learning curves of value-based CDRL, i.e., C51 algorithm, and decomposed algorithm $\mathcal{H}(\mu, q_\theta)$ after the return distribution decomposition with different ε on eight typical Atari games. Results are averaged over 3 seeds and the shade represents the standard deviation.

effect from distributional RL by comparing its performance with the classical RL and CDRL algorithms. To ensure a pre-specified ϵ that guarantees a valid decomposition analyzed in Proposition 1, we re-define a new notation ε, which shares the same utility with ϵ and is more convenient in the implementation. ε is defined as the mass proportion centered at the bin that contains the expectation *when transporting the mass to other bins*. A large proportion probability ε that transports less mass to other bins corresponds to a large ϵ in Eq. 2.4, under which the decomposed algorithm performs more similarly to a pure CDRL algorithm. See Appendix 2.9.10 for more explanation, including the transformation equation between ϵ and ε.

Figure 2.2 showcases that as ε gradually decreases from 0.8 to 0.1, learning curves of decomposed C51 denoted as $\mathcal{H}(\mu, q_\theta)(\varepsilon = 0.8/0.5/0.1)$ tend to degrade from vanilla C51 to DQN across most Atari games. The sensitivity of decomposed algorithm $\mathcal{H}(\mu, q_\theta)$ in terms of ε depends on the environment. Similar results in continuous control environments can be found in Appendix 2.9.12. Overall, our empirical result corroborates the decomposed uncertainty-aware entropy regularization is pivotal to the empirical benefits of being categorical distributional in CDRL over classical RL.

2.7.2 Interpolation Behavior of DERAC: Mitigating the Over-Exploration

Figure 2.3 suggests that DERAC (green) converges and tends to "interpolate" between the expectation-based AC and its distributional counterpart denoted by DAC (C51), which substantiates the theoretical convergence of the tabular DERPI algorithm in Theorem 1. We highlight that *the purpose of introducing DERAC is to interpret the benefits of CDRL from the perspective of uncertain-aware regularization, instead of only pursuing the empirical superiority.* In Group 1, it is important to note that DERAC achieves superior performance over both AC and DAC (C51) on bipedalwalkerhardcore, which demonstrates that the interpolation has extra advantages. We posit that the interpolation nature of DERAC mitigates the over-exploration induced by the categorical distributional learning in C51, as a pure CDRL algorithm may put

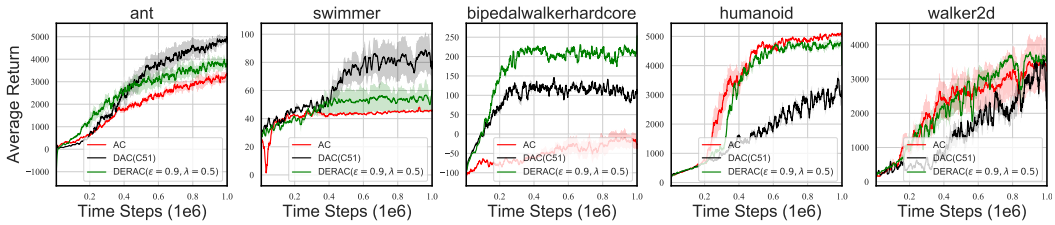


Figure 2.3: Learning curves of DERAC algorithm averaged over five seeds. The AC and DAC baselines are without the leverage of entropy regularization of MaxEnt. Group 1: Ant, Swimmer and Bipedalwalkerhardcore, where DAC (C51) outperforms AC. Group 2: Humanoid and Walker2d, where AC outperforms DAC (C51).

too much emphasis on the uncertainty-aware exploration, i.e., all weight on the regularization term in Entropy-regularized Neural FQI in Eq. 2.5. In Group 2 where distributional algorithm (DAC) is inferior to its expectation-based counterpart (AC), it turns out DERAC performs similarly to or slightly excels at AC. These results demonstrate that DERAC accomplishes a more robust performance between expectation-based AC and DAC (C51) algorithms and can even surpass DAC (C51) by potentially mitigating the over-exploration of variants of CDRL algorithms. We also provide a sensitivity analysis of DERAC regarding λ in Appendix 2.9.12.

2.7.3 Mutual Impacts of Vanilla Entropy Regularization and Uncertainty-aware Regularization

We demonstrate that the two types of exploration encouraged by Vanilla Entropy (*VE*) in MaxEnt RL and Uncertainty-aware Entropy (*UE*) in CDRL, despite having similar entropy regularization forms, play distinct roles in the learning when used simultaneously, either mutual improvement or potential interference. We conduct an ablation study for both DSAC (C51) and DSAC (IQN), where the latter is used to heuristically examine the mutual impacts in the quantile-based distributional RL algorithm. We leave similar results conducted on DSAC (C51) in Appendix 2.9.12. Specifically, we denote SAC with/without vanilla entropy as *AC+VE* and *AC*, and distributional SAC with/without vanilla entropy as *AC+UE+VE* and ***AC+UE*** or *DAC*. The implementation

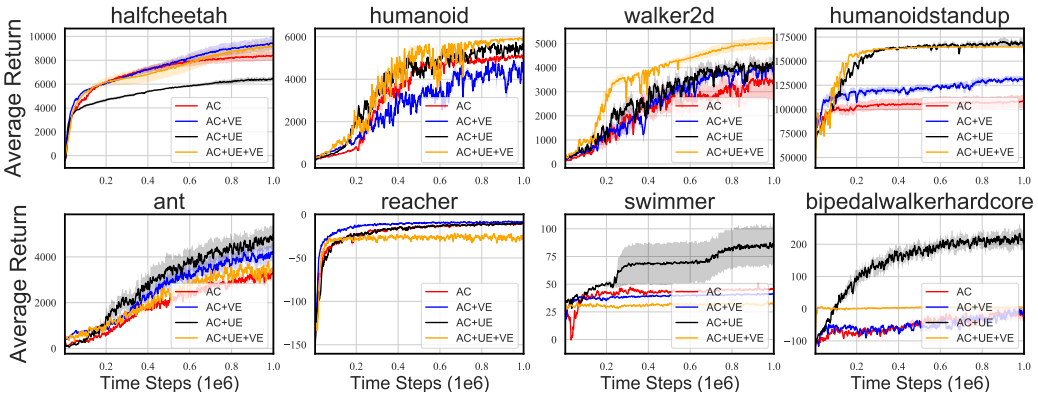


Figure 2.4: Learning curves of *AC*, *AC+VE* (SAC), *AC+UE* (DAC) and *AC+UE+VE* (DSAC) over five seeds across eight MuJoCo environments where DAC and DSAC are based on IQN. (First Row): Mutual improvement. (Second Row): Potential interference.

details can be found in Appendix 2.9.10.

In the first row in Figure 2.4, the simultaneous leverage of uncertainty-aware and vanilla entropy regularization renders a mutual improvement. Conversely, the two regularizations when employed together lead to performance degradation in the second row in Figure 2.4, such as Swimmer and Reacher, where *AC+UE+VE* is significantly inferior to ***AC+UE*** or *AC+VE*. We posit that the potential interference may result from distinct exploration directions in the policy learning for the two regularizations. SAC optimizes the policy to visit states with high entropy, while distributional RL updates the policy to explore states and the associated actions whose current return distribution estimate lags far behind the correct one determined by the environment uncertainty.

2.8 Discussions and Conclusion

In this paper, we interpret the benefits of CDRL over classical RL as an uncertainty-aware regularization derived through the return density decomposition. In contrast to encouraging diverse actions for the exploration in MaxEnt RL, the uncertainty-aware regularization in CDRL promotes to explore states where the environment uncertainty is largely underestimated. This

novel exploration from CDRL contributes to explaining the benefits of being (categorical) distributional in RL.

Limitations and Future Work. The uncertainty-aware regularization with the exploration effect is founded on CDRL. However, it remains elusive whether it is feasible to extend the uncertainty-aware exploration in CDRL to general distributional RL, given that the analytical techniques in other classes, such as QR-DQN, are highly different from CDRL. We leave this extension as future work.

2.9 Appendix

2.9.1 Convergence Guarantee of Categorical Distributional RL

Categorical Distributional RL [9] uses the heuristic projection operator $\Pi_{\mathcal{C}}$ that was defined as

$$\Pi_{\mathcal{C}}\left(\delta_y\right) = \begin{cases} \delta_{l_1} & y \leq l_1 \\ \frac{l_{i+1}-y}{l_{i+1}-z_i}\delta_{l_i} + \frac{y-l_i}{l_{i+1}-z_i}\delta_{l_{i+1}} & l_i < y \leq l_{i+1} \\ \delta_{l_K} & y > l_K \end{cases}, \tag{2.12}$$

and extended affinely to finite mixtures of Dirac measures, so that for a mixture of Diracs $\sum_{i=1}^{N} p_i \delta_{y_i}$, we have $\Pi_{\mathcal{C}}\left(\sum_{i=1}^{N} p_i \delta_{y_i}\right) = \sum_{i=1}^{N} p_i \Pi_{\mathcal{C}}\left(\delta_{y_i}\right)$. The Cramér distance was recently studied as an alternative to the Wasserstein distances in the context of generative models [10]. Recall the definition of Cramér distance.

Definition 1. *(Definition 3 [84]) The Cramér distance ℓ_2 between two distributions $\nu_1, \nu_2 \in \mathscr{P}(\mathbb{R})$, with cumulative distribution functions F_{ν_1}, F_{ν_2} respectively, is defined by:*

$$\ell_2\left(\nu_1, \nu_2\right) = \left(\int_{\mathbb{R}} \left(F_{\nu_1}(x) - F_{\nu_2}(x)\right)^2 \, \mathrm{d}x\right)^{1/2}.$$

Further, the supremum-Cramér metric $\bar{\ell}_2$ is defined between two distribution

functions $\eta, \mu \in \mathscr{P}(\mathbb{R})^{\mathcal{X}\times\mathcal{A}}$ *by*

$$\bar{\ell}_2(\eta, \mu) = \sup_{(x,a)\in\mathcal{X}\times\mathcal{A}} \ell_2\left(\eta^{(x,a)}, \mu^{(x,a)}\right).$$

Thus, the contraction of categorical distributional RL can be guaranteed under Cramér distance:

Proposition 4. *(Proposition 2 [84]) The operator* $\Pi_{\mathcal{C}}\mathcal{T}^{\pi}$ *is a* $\sqrt{\gamma}$*-contraction in* $\bar{\ell}_2$*.*

An insight behind this conclusion is that Cramér distance endows a particular subset with a notion of orthogonal projection, and the orthogonal projection onto the subset is exactly the heuristic projection $\Pi_{\mathcal{C}}$ (Proposition 1 in [84]). [84] also states that the operator $\Pi_{\mathcal{C}}\mathcal{T}^{\pi}$ is contractive under Wasserstein distance.

2.9.2 Proof of Proposition 1

Proposition 1. Denote $\widehat{p}^{s,a}(x \in \Delta_E) = p_E/\Delta$. Following the density function decomposition in Eq. 2.4, $\widehat{\mu}(x) = \sum_{i=1}^{N} p_i^{\mu}\mathbb{1}(x \in \Delta_i)/\Delta$ is a valid probability density function if and only if $\epsilon \geq 1 - p_E$.

Proof. Recap a valid probability density function requires non-negative and one-bounded probability in each bin and all probabilities should sum to 1.
Necessity. (1) When $x \in \Delta_E$, Eq. 2.4 can simplified as $p_E/\Delta = (1-\epsilon)/\Delta + \epsilon p_E^{\mu}/\Delta$, where $p_E^{\mu} = \widehat{\mu}(x \in \Delta_E)$. Thus, $p_E^{\mu} = \frac{p_E}{\epsilon} - \frac{1-\epsilon}{\epsilon} \geq 0$ if $\epsilon \geq 1 - p_E$. Obviously, $p_E^{\mu} = \frac{p_E}{\epsilon} - \frac{1-\epsilon}{\epsilon} \leq \frac{1}{\epsilon} - \frac{1-\epsilon}{\epsilon} = 1$ guaranteed by the validity of $\widehat{p}_E^{s,a}$. (2) When $x \notin \Delta_E$, we have $p_i/\Delta = \epsilon p_i^{\mu}/\Delta$, i.e.,When $x \notin \Delta_E$, We immediately have $p_i^{\mu} = \frac{p_i}{\epsilon} \leq \frac{1-p_E}{\epsilon} \leq 1$ when $\epsilon \geq 1 - p_E$. Also, $p_i^{\mu} = \frac{p_i}{\epsilon} \geq 0$.
Sufficiency. (1) When $x \in \Delta_E$, let $p_E^{\mu} = \frac{p_E}{\epsilon} - \frac{1-\epsilon}{\epsilon} \geq 0$, we have $\epsilon \geq 1 - p_E$. $p_E^{\mu} = \frac{p_E}{\epsilon} - \frac{1-\epsilon}{\epsilon} \leq 1$ in nature. (2) When $x \notin \Delta_E$, $p_i^{\mu} = \frac{p_i}{\epsilon} \geq 0$ in nature. Let $p_i^{\mu} = \frac{p_i}{\epsilon} \leq 1$, we have $p_i \leq \epsilon$. We need to take the intersection set of (1) and (2), and we find that $\epsilon \geq 1 - p_E \Rightarrow \epsilon \geq 1 - p_E \geq p_i$ that satisfies the condition in (2). Thus, the intersection set of (1) and (2) would be $\epsilon \geq 1 - p_E$.

As $\epsilon \geq 1 - p_E$ is both the necessary and sufficient condition, we have the conclusion that $\widehat{\mu}(x)$ is a valid probability density function $\iff \epsilon \geq 1 - p_E$.

□

2.9.3 Equivalence between Categorical and Histogram Parameterization

Proposition 5. *Suppose the target categorical distribution $c = \sum_{i=1}^{N} p_i \delta_{z_i}$ and the target histogram function $h(x) = \sum_{i=1}^{N} p_i \mathbb{1}(x \in \Delta_i)/\Delta$, updating the parameterized categorical distribution c_θ under KL divergence is equivalent to updating the parameterized histogram function h_θ.*

Proof. For the histogram density estimator h_θ and the true target density function $p(x)$, we can simplify the KL divergence as follows.

$$\begin{aligned} D_{\mathrm{KL}}(h, h_\theta) &= \sum_{i=1}^{N} \int_{l_{i-1}}^{l_i} \frac{p_i(x)}{\Delta} \log \frac{\frac{p_i(x)}{\Delta}}{\frac{h_\theta^i}{\Delta}} dx \\ &= \sum_{i=1}^{N} \int_{l_{i-1}}^{l_i} \frac{p_i(x)}{\Delta} \log \frac{p_i(x)}{\Delta} dx - \sum_{i=1}^{N} \int_{l_{i-1}}^{l_i} \frac{p_i(x)}{\Delta} \log \frac{h_\theta^i}{\Delta} dx \\ &\propto - \sum_{i=1}^{N} \int_{l_{i-1}}^{l_i} \frac{p_i(x)}{\Delta} \log \frac{h_\theta^i}{\Delta} dx \\ &= - \sum_{i=1}^{N} p_i(x) \log \frac{h_\theta^i}{\Delta} \propto - \sum_{i=1}^{N} p_i(x) \log h_\theta^i \end{aligned} \tag{2.13}$$

where h_θ^i is determined by i and θ and is independent of x. For categorical distribution estimator c_θ with the probability p_i in for each atom z_i, we also have its target categorical distribution $p(x)$ with each probability p_i, we have:

$$D_{\mathrm{KL}}(c, c_\theta) = \sum_{i=1}^{N} p_i \log \frac{p_i}{c_\theta^i} = \sum_{i=1}^{N} p_i \log p_i - \sum_{i=1}^{N} p_i \log c_\theta^i \propto - \sum_{i=1}^{N} p_i \log c_\theta^i \tag{2.14}$$

□

In CDRL, we only use a discrete categorical distribution with probabilities

centered on the fixed atoms $\{z_i\}_{i=1}^N$, while the histogram density estimator in our analysis is a continuous function defined on $[z_0, z_N]$ to allow richer analysis. We reveal that minimizing the KL divergence regarding the parameterized categorical distribution in Eq. 2.14 is equivalent to minimizing the cross-entropy loss regarding the parameterized histogram function in Eq. 2.13.

2.9.4 Theoretical Results of Histogram Density Estimator in Distributional RL

Histogram Function Parameterization Error: Uniform Convergence in Probability. The previous discrete categorical parameterization error bound in [84] (Proposition 3) is derived between the true return distribution and the limiting return distribution denoted as $\eta_{\mathcal{C}}$ iteratively updated via the Bellman operator $\Pi_{\mathcal{C}}\mathfrak{T}^{\pi}$ *in expectation*, without considering an asymptotic analysis when the number of sampled $\{s_i, a_i\}_{i=1}^n$ pairs goes to infinity. As a complementary result, we provide a uniform convergence rate for the histogram density estimator in the context of distributional RL. In this particular analysis within this subsection, we denote $\widehat{p}_{\mathcal{C}}^{s,a}$ as the density function estimator for the true limiting return distribution $\eta_{\mathcal{C}}$ via $\Pi_{\mathcal{C}}\mathfrak{T}^{\pi}$ with its true density $p_{\mathcal{C}}^{s,a}$. In Theorem 2, we show that the sample-based histogram estimator $\widehat{p}_{\mathcal{C}}^{s,a}$ can approximate any arbitrary continuous limiting density function $p_{\mathcal{C}}^{s,a}$ under a mild condition. This ensures the use of a histogram density estimator in the implementation of our subsequent algorithm adapted from CDRL.

Theorem 2. *(Uniform Convergence Rate in Probability) Suppose $p_{\mathcal{C}}^{s,a}(x)$ is Lipschitz continuous and the support of a random variable is partitioned by N bins with bin size Δ. Then*

$$\sup_x |\widehat{p}_{\mathcal{C}}^{s,a}(x) - p_{\mathcal{C}}^{s,a}(x)| = O(\Delta) + O_P\left(\sqrt{\frac{\log N}{n\Delta^2}}\right). \tag{2.15}$$

Proof. Our proof is mainly based on the non-parametric statistics analysis [108].

In particular, the difference of $\widehat{p}_{\mathcal{C}}^{s,a}(x) - p_{\mathcal{C}}^{s,a}(x)$ can be written as

$$\widehat{p}_{\mathcal{C}}^{s,a}(x) - p_{\mathcal{C}}^{s,a}(x) = \underbrace{\mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right) - p_{\mathcal{C}}^{s,a}(x)}_{\text{bias}} + \underbrace{\widehat{p}_{\mathcal{C}}^{s,a}(x) - \mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right)}_{\text{stochastic variation}}. \tag{2.16}$$

(1) The first bias term. Without loss of generality, we consider $x \in \Delta_k$, then

$$\begin{aligned}\mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right) &= \frac{P(X \in \Delta_k)}{\Delta} = \frac{\int_{l_0+(k-1)\Delta}^{l_0+k\Delta} p(y)dy}{\Delta} \\ &= \frac{F(l_0 + (k-1)\Delta) - F(l_0 + (k-1)\Delta)}{l_0 + k\Delta - (l_0 + (k-1)\Delta)} = p_{\mathcal{C}}^{s,a}(x'),\end{aligned} \tag{2.17}$$

where the last equality is based on the mean value theorem. According to the L-Lipschitz continuity property, we have

$$\left|\mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right) - p_{\mathcal{C}}^{s,a}(x)\right| = \left|p_{\mathcal{C}}^{s,a}(x') - p_{\mathcal{C}}^{s,a}(x)\right| \leq L|x' - x| \leq L\Delta \tag{2.18}$$

(2) The second stochastic variation term. If we let $x \in \Delta_k$, then $\widehat{p}_{\mathcal{C}}^{s,a} = p_k = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(X_i \in \Delta_k)$, we thus have

$$\begin{aligned}&P\left(\sup_x \left|\widehat{p}_{\mathcal{C}}^{s,a}(x) - \mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right)\right| > \epsilon\right) \\ &= P\left(\max_{j=1,\cdots,N}\left|\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}\left(X_i \in \Delta_j\right)/\Delta - P\left(X_i \in \Delta_j\right)/\Delta\right| > \epsilon\right) \\ &= P\left(\max_{j=1,\cdots,N}\left|\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}\left(X_i \in \Delta_j\right) - P\left(X_i \in \Delta_j\right)\right| > \Delta\epsilon\right) \\ &\leq \sum_{j=1}^{N} P\left(\left|\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}\left(X_i \in \Delta_j\right) - P\left(X_i \in \Delta_j\right)\right| > \Delta\epsilon\right) \\ &\leq N \cdot \exp\left(-2n\Delta^2\epsilon^2\right) \quad \text{(by Hoeffding's inequality)},\end{aligned} \tag{2.19}$$

where in the last inequality we know that the indicator function is bounded in [0, 1]. We then let the last term be a constant independent of N, n, Δ and simplify the order of ϵ. Then, we have:

$$\sup_x \left|\widehat{p}_{\mathcal{C}}^{s,a}(x) - \mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right)\right| = O_P\left(\sqrt{\frac{\log N}{n\Delta^2}}\right) \tag{2.20}$$

In summary, as the above inequality holds for each x, we thus have the uniform convergence rate of a histogram density estimator

$$\begin{aligned}\sup_x |\widehat{p}_{\mathcal{C}}^{s,a}(x) - p_{\mathcal{C}}^{s,a}(x)| &\leq \sup_x |\mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right) - p_{\mathcal{C}}^{s,a}(x)| + \sup_x |\widehat{p}_{\mathcal{C}}^{s,a}(x) - \mathbb{E}\left(\widehat{p}_{\mathcal{C}}^{s,a}(x)\right)| \\ &= O\left(\Delta\right) + O_P\left(\sqrt{\frac{\log N}{n\Delta^2}}\right).\end{aligned} \tag{2.21}$$

□

2.9.5 Discussion: KL Divergence in Distributional RL

Remark on KL Divergence. As stated in Section 2.4 of CDRL [9], when the categorical parameterization is applied after the projection operator $\Pi_{\mathcal{C}}$, the distributional Bellman operator $\mathfrak{T}^\pi$ has the contraction guarantee under Cramér distance or Wasserstein distance [84], albeit the direct use of a non-expansive KL divergence [75]. Similarly, our histogram density parameterization with the projection $\Pi_{\mathcal{C}}$ and KL divergence also enjoys a contraction property due to the equivalence between optimizing histogram function and categorical distribution analyzed in Appendix 2.9.3. We summarize some properties of KL divergence in distributional RL in Proposition 6.

Proposition 6. *Given two probability measures μ and ν, we define the supreme D_{KL} as a functional $\mathcal{P}(\mathcal{X})^{\mathcal{S}\times\mathcal{A}} \times \mathcal{P}(\mathcal{X})^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R}$, i.e.,*

$$D_{KL}^{\infty}(\mu,\nu) = \sup_{(s,a)\in\mathcal{S}\times\mathcal{A}} D_{KL}(\mu(s,a),\nu(s,a)).$$

We have: (1) $\mathfrak{T}^\pi$ is a non-expansive distributional Bellman operator under D_{KL}^{∞}, i.e., $D_{KL}^{\infty}(\mathfrak{T}^\pi Z_1, \mathfrak{T}^\pi Z_2) \leq D_{KL}^{\infty}(Z_1, Z_2)$, (2) $D_{KL}^{\infty}(Z_n, Z) \to 0$ implies the Wasserstein distance $W_p(Z_n, Z) \to 0$.

Proof. We first assume Z_θ is absolutely continuous and the supports of two distributions in KL divergence have a negligible intersection [6], under which the KL divergence is well-defined.

(1) The contraction analysis of distributional Bellman operator $\mathfrak{T}^\pi$ under a distribution divergence d_p depends on its *scale sensitive* **(S)** and *sum in-*

variant **(I)** properties [10, 9]. We say d_p is scale sensitive (of order τ) if there exists a $\tau > 0$, such that for all random variables X, Y and a real value $a > 0$, $d_p(aX, aY) \leq |a|^\tau d_p(X, Y)$. d_p has the sum invariant property if whenever a random variable A is independent from X, Y, we have $d_p(A + X, A + Y) \leq d_p(X, Y)$. We first prove that the D_{KL} is sum-invariant, which is based on the dual form of KL divergence via the variational representation [25, 3]:

$$D_{\mathrm{KL}}(X, Y) = \sup_{f \in \mathcal{L}^b} \{\mathbb{E}_X[f(x)] - \log\left(\mathbb{E}_Y\left[e^{f(y)}\right]\right)\}, \tag{2.22}$$

where $\mathcal{L}^b$ is the space of bounded measurable functions. Consequently,

$$\begin{aligned}
D_{\mathrm{KL}}(A + X, A + Y) &= \sup_{f \in \mathcal{L}^b} \{\mathbb{E}_{Z_1 = A+X}[f(z_1)] - \log\left(\mathbb{E}_{Z_2 = A+Y}\left[e^{f(z_2)}\right]\right)\} \\
&\overset{(a)}{=} \sup_{f \in \mathcal{L}^b} \{\mathbb{E}_A\left[\mathbb{E}_X\left[f(x + a)\right]\right] - \log\left(\mathbb{E}_A\left[\mathbb{E}_Y\left[e^{f(y+a)}\right]\right]\right)\} \\
&\overset{(b)}{\leq} \sup_{f \in \mathcal{L}^b} \{\mathbb{E}_A\mathbb{E}_X[f(x + a)] - \mathbb{E}_A \log\left(\mathbb{E}_Y\left[e^{f(y+a)}\right]\right)\} \\
&= \sup_{f \in \mathcal{L}^b} \{\mathbb{E}_A[\mathbb{E}_X[f(x + a)] - \log\left(\mathbb{E}_Y\left[e^{f(y+a)}\right]\right)]\} \\
&\overset{(c)}{\leq} \mathbb{E}_A \sup_{f \in \mathcal{L}^b} \{\mathbb{E}_X[f(x + a)] - \log\left(\mathbb{E}_Y\left[e^{f(y+a)}\right]\right)\} \\
&\overset{(d)}{=} \mathbb{E}_A \sup_{g \in \mathcal{L}^b} \{\mathbb{E}_X[g(x)] - \log\left(\mathbb{E}_Y\left[e^{g(y)}\right]\right)\} \\
&= D_{\mathrm{KL}}(X, Y),
\end{aligned} \tag{2.23}$$

where (a) results from the independence between A and X (Y). (b) and (c) rely on the Jensen inequality for the function $-\log$ and the operator sup. (d) is because the translation is still within the same bounded functional space. Next, we show that D_{KL} is not scale-sensitive, where we denote the probability density function of X and Y as p and q.

$$D_{\mathrm{KL}}(aX, aY) = \int_{-\infty}^{\infty} \frac{1}{a} p\left(\frac{x}{a}\right) \log \frac{\frac{1}{a} p\left(\frac{x}{a}\right)}{\frac{1}{a} q\left(\frac{x}{a}\right)} \mathrm{d}x = \int_{-\infty}^{\infty} p(y) \log \frac{p(y)}{q(y)} \mathrm{d}y = D_{\mathrm{KL}}(X, Y) \tag{2.24}$$

Putting the two properties together and given two return distributions $Z_1(s,a)$ and $Z_2(s,a)$, we have the non-expansive contraction property of the supremal form of D_{KL} as follows.

$$\begin{aligned} D_{\mathrm{KL}}^{\infty}(\mathfrak{T}^{\pi}Z_1, \mathfrak{T}^{\pi}Z_2) &= \sup_{s,a} D_{\mathrm{KL}}(\mathfrak{T}^{\pi}Z_1(s,a), \mathfrak{T}^{\pi}Z_2(s,a)) \\ &= \sup_{s,a} D_{\mathrm{KL}}(R(s,a)+\gamma Z_1(s',a'), R(s,a)+\gamma Z_2(s',a')) \\ &\overset{(a)}{\leq} D_{\mathrm{KL}}(\gamma Z_1(s',a'), \gamma Z_2(s',a')) \\ &\overset{(b)}{=} D_{\mathrm{KL}}(Z_1(s',a'), Z_2(s',a')) \\ &\leq \sup_{s,a} D_{\mathrm{KL}}(Z_1(s',a'), Z_2(s',a')) \\ &= D_{\mathrm{KL}}^{\infty}(Z_1, Z_2), \end{aligned} \tag{2.25}$$

where (a) relies on the sum invariant property of D_{KL} and (b) utilizes the non-scale sensitive property of D_{KL}. By applying the well-known Banach fixed point theorem, we have a unique return distribution when convergence of distributional dynamic programming under D_{KL}^{∞}.

(2) By the definition of D_{KL}^{∞}, we have $\sup_{s,a} D_{\mathrm{KL}}(Z_n(s,a), Z(s,a)) \to 0$ implies $D_{\mathrm{KL}}(Z_n, Z) \to 0$. $D_{\mathrm{KL}}(Z_n, Z) \to 0$ implies the total variation distance $\delta(Z_n, Z) \to 0$ according to a straightforward application of Pinsker's inequality

$$\delta\left(Z_n, Z\right) \leq \sqrt{\frac{1}{2} D_{\mathrm{KL}}\left(Z_n, Z\right)} \to 0, \quad \delta\left(Z, Z_n\right) \leq \sqrt{\frac{1}{2} D_{\mathrm{KL}}\left(Z, Z_n\right)} \to 0 \tag{2.26}$$

Based on Theorem 2 in WGAN [7], $\delta(Z_n, Z) \to 0$ implies $W_p(Z_n, Z) \to 0$. This is trivial by recalling the fact that δ and W give the strong and weak topologies on the dual of $(C(\mathcal{X}), \|\cdot\|_{\infty})$ when restricted to $\mathrm{Prob}(\mathcal{X})$.

□

2.9.6 Proof of Proposition 2

Proposition 2 (Decomposed Neural FZI) Denote $q_{\theta}^{s,a}(x)$ as the histogram density function of $Z_{\theta}^{k}(s,a)$ in Neural FZI. Based on Eq. 2.4 and KL divergence

as d_p, Neural FZI in Eq. 2.3 is simplified as

$$Z_\theta^{k+1} = \underset{q_\theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} [\underbrace{-\log q_\theta^{s_i,a_i}(\Delta_E^i)}_{(a)} + \alpha \mathcal{H}(\widehat{\mu}^{s_i',\pi_Z(s_i')}, q_\theta^{s_i,a_i})], \tag{2.27}$$

Proof. Firstly, given a fixed $p(x)$ we know that minimizing $D_{\mathrm{KL}}(p, q_\theta)$ is equivalent to minimizing $\mathcal{H}(p, q)$ by following

$$\begin{aligned} D_{\mathrm{KL}}(p, q_\theta) &= \sum_{i=1}^{N} \int_{l_{i-1}}^{l_i} p_i(x)/\Delta \log \frac{p^i(x)/\Delta}{q_\theta^i/\Delta} \, \mathrm{d}x \\ &= -\sum_{i=1}^{N} \int_{l_{i-1}}^{l_i} p_i(x)/\Delta \log q_\theta^i/\Delta \, \mathrm{d}x - (\sum_{i=1}^{N} \int_{l_{i-1}}^{l_i} p_i(x)/\Delta \log p^i(x)/\Delta \, \mathrm{d}x) \\ &= \mathcal{H}(p, q_\theta) - \mathcal{H}(p) \\ &\propto \mathcal{H}(p, q_\theta) \end{aligned} \tag{2.28}$$

where $p = \sum_{i=1}^{N} p_i(x)\mathbb{1}(x \in \Delta^i)/\Delta$ and $q_\theta = \sum_{i=1}^{N} q_i/\Delta$. Based on $\mathcal{H}(p, q_\theta)$, we use $p^{s_i',\pi_Z(s_i')}(x)$ to denote the target probability density function of the random variable $R(s_i, a_i)+\gamma Z_{\theta^*}^k(s_i', \pi_Z(s_i'))$. Then, we can derive the objective function within each Neural FZI as

$$\begin{aligned} &\frac{1}{n} \sum_{i=1}^{n} \mathcal{H}(p^{s_i',\pi_Z(s_i')}(x), q_\theta^{s_i,a_i}) \\ &= \frac{1}{n} \sum_{i=1}^{n} \left((1-\epsilon) \mathcal{H}(\mathbb{1}(x \in \Delta_E^i)/\Delta, q_\theta^{s_i,a_i}) + \epsilon \mathcal{H}(\widehat{\mu}^{s_i',\pi_Z(s_i')}, q_\theta^{s_i,a_i}) \right) \\ &= \frac{1}{n} \sum_{i=1}^{n} \left(-(1-\epsilon) \sum_{j=1}^{N} \int_{l_{j-1}}^{l_j} \mathbb{1}(x \in \Delta_E^i)/\Delta \log q_\theta^{s_i,a_i}(\Delta_j)/\Delta dx - \epsilon \sum_{j=1}^{N} \int_{l_{j-1}}^{l_j} p_j^\mu/\Delta \log q_\theta^{s_i,a_i}(\Delta_j)/\Delta \right) \\ &= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\Delta} \left((1-\epsilon)(-\log q_\theta^{s_i,a_i}(\Delta_E^i)/\Delta) - \epsilon \sum_{j=1}^{N} p_j^\mu \log q_\theta^{s_i,a_i}(\Delta_j)/\Delta \right) \\ &\propto \frac{1}{n} \sum_{i=1}^{n} \left((1-\epsilon)(-\log q_\theta^{s_i,a_i}(\Delta_E^i)) + \epsilon \mathcal{H}(\widehat{\mu}^{s_i',\pi_Z(s_i')}, q_\theta^{s_i,a_i}) \right) \\ &\propto \frac{1}{n} \sum_{i=1}^{n} \left(-\log q_\theta^{s_i,a_i}(\Delta_E^i) + \alpha \mathcal{H}(\widehat{\mu}^{s_i',\pi_Z(s_i')}, q_\theta^{s_i,a_i}) \right), \end{aligned} \tag{2.29}$$

where $\alpha = \frac{\epsilon}{1-\epsilon} > 0$. Recall that $\widehat{\mu}^{s_i',\pi_Z(s_i')} = \sum_{i=1}^{N} p_i^\mu(x)\mathbb{1}(x \in \Delta_i)/\Delta = \sum_{i=1}^{N} p_i^\mu/\Delta$ for conciseness and denote $q_\theta^{s_i,a_i} = \sum_{j=1}^{N} q_\theta^{s_i,a_i}(\Delta_j)/\Delta$. The cross-entropy $\mathcal{H}(\widehat{\mu}^{s_i',\pi_Z(s_i')}, q_\theta^{s_i,a_i})$ is based on the discrete distribution when $i =$

$1, ..., N$. Δ_E^i represent the interval that $\mathbb{E}\left[R(s_i, a_i) + \gamma Z_{\theta^*}^k\left(s_i', \pi_Z(s_i')\right)\right]$ falls into, i.e., $\mathbb{E}\left[R(s_i, a_i) + \gamma Z_{\theta^*}^k\left(s_i', \pi_Z(s_i')\right)\right] \in \Delta_E^i$. □

2.9.7 Proof of Proposition 3

Proposition 3 (Equivalence between **the term (a)** in Decomposed Neural FZI and Neural FQI) In Eq. 2.5 of Neural FZI, assume the function class $\{Z_\theta : \theta \in \Theta\}$ is sufficiently large such that it contains the target $\{Y_i^k\}_{i=1}^n$, when $\Delta \to 0$, for all k, minimizing **the term (a)** in Eq. 2.5 implies

$$P(Z_\theta^{k+1}(s,a) = \mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)) = 1, \text{ and } \int_{-\infty}^{+\infty} \left| F_{q_\theta}(x) - F_{\delta_{\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)}}(x) \right| dx = o(\Delta), \tag{2.30}$$

where $\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)$ is the scalar-valued target in the k-th phase of Neural FQI, and $\delta_{\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)}$ is the Dirac delta function defined on the scalar $\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)$.

Proof. Firstly, we define the distributional Bellman optimality operator $\mathfrak{T}^{\text{opt}}$ as follows:

$$\mathfrak{T}^{\text{opt}} Z(s,a) \overset{D}{=} R(s,a) + \gamma Z\left(S', a^*\right), \tag{2.31}$$

where $S' \sim P(\cdot \mid s,a)$ and $a^* = \underset{a'}{\text{argmax}} \mathbb{E}\left[Z\left(S', a'\right)\right]$. If $\{Z_\theta : \theta \in \Theta\}$ is sufficiently large enough such that it contains $\mathfrak{T}^{\text{opt}} Z_{\theta^*}$ $(\{Y_i^k\}_{i=1}^n)$, then optimizing Neural FZI in Eq. 2.3 leads to $Z_\theta^{k+1} = \mathfrak{T}^{\text{opt}} Z_{\theta^*}$.

Secondly, we apply the action-value density function decomposition on the target histogram function $\widehat{p}^{s,a}(x)$. Consider the parameterized histogram density function h_θ and denote h_θ^E/Δ as the bin height in the bin Δ_E, under the KL divergence between the first histogram function $\mathbb{1}(x \in \Delta_E)$ with $h_\theta(x)$, the objective function is simplified as

$$D_{\text{KL}}(\mathbb{1}(x \in \Delta_E)/\Delta, h_\theta(x)) = -\int_{x \in \Delta_E} \frac{1}{\Delta} \log \frac{\frac{h_\theta^E}{\Delta}}{\frac{1}{\Delta}} dx \quad = -\log h_\theta^E \tag{2.32}$$

Since $\{Z_\theta : \theta \in \Theta\}$ is sufficiently large enough that can represent the pdf of $\{Y_i^k\}_{i=1}^n$, it also implies that $\{Z_\theta : \theta \in \Theta\}$ can represent the term (a) part in its pdf via the return density decomposition. The KL minimizer would be $\widehat{h}_\theta = \mathbb{1}(x \in \Delta_E)/\Delta$ in expectation. Then, $\lim_{\Delta \to 0} \arg\min_{h_\theta} D_{\text{KL}}(\mathbb{1}(x \in$

$\Delta_E)/\Delta, h_\theta(x)) = \delta_{\mathbb{E}[Z^{\text{target}}(s,a)]}$, where $\delta_{\mathbb{E}[Z^{\text{target}}(s,a)]}$ is a Dirac Delta function centered at $\mathbb{E}\left[Z^{\text{target}}(s,a)\right]$ and can be viewed as a generalized probability density function. That being said, the limiting probability density function (pdf) converges to a Dirac delta function at $\mathbb{E}\left[Z^{\text{target}}(s,a)\right]$. The limit behavior from a histogram function $\widehat{p}$ to a continuous one for Z^{target} is guaranteed by Theorem 2, and this also applies from h_θ to Z_θ. In Neural FZI, we have $Z^{\text{target}} = \mathfrak{T}^{\text{opt}} Z_{\theta^*}$. Since here we use $Z_\theta^{k+1}(s,a)$ as the random variable who cdf is the limiting distribution, according to the definition of the Dirac function, as $\Delta \to 0$, we attain

$$P(Z_\theta^{k+1}(s,a) = \mathbb{E}\left[\mathfrak{T}^{\text{opt}} Z_{\theta^*}^k(s,a)\right]) = 1, \tag{2.33}$$

which is because if the pdf of a random variable is a Dirac delta function, it implies that the random variable takes this constant value with probability one. Due to the linearity of expectation in Lemma 4 of [9], we have

$$\mathbb{E}\left[\mathfrak{T}^{\text{opt}} Z_{\theta^*}^k(s,a)\right] = \mathfrak{T}^{\text{opt}} \mathbb{E}\left[Z_{\theta^*}^k(s,a)\right] = \mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a) \tag{2.34}$$

Finally, we obtain:

$$P(Z_\theta^{k+1}(s,a) = \mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)) = 1 \quad \text{as} \quad \Delta \to 0 \tag{2.35}$$

In order to characterize how the difference varies when $\Delta \to 0$, we further define $\Delta_E = [l_e, l_{e+1})$ and we have:

$$\begin{aligned}
&\int_{-\infty}^{+\infty} \left| F_{q_\theta}(x) - F_{\delta_{\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a)}}(x) \right| dx \\
&= \frac{1}{2\Delta} \left(\left(\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a) - l_e \right)^2 + \left(l_{e+1} - \mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a) \right)^2 \right) \\
&= \frac{1}{2\Delta} (a^2 + (\Delta - a)^2) \\
&\leq \Delta/2 \\
&= o(\Delta),
\end{aligned} \tag{2.36}$$

where $\mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a) = \mathbb{E}\left[\mathfrak{T}^{\text{opt}} Z_{\theta^*}^k(s,a)\right] \in \Delta_E$ and we denote $a = \mathcal{T}^{\text{opt}} Q_{\theta^*}^k(s,a) -$

l_e. The first equality holds as $q_\theta(x)$, the KL minimizer while minimizing the term (a), would follows a uniform distribution on Δ_E, i.e., $\widehat{q}_\theta = \mathbb{1}(x \in \Delta_E)/\Delta$. Thus, the integral of LHS would be the area of two centralized triangles according. The inequality is because the maximizer is obtained when $a = \Delta$ or 0. The result implies that the convergence rate in distribution difference is $o(\Delta)$.

□

2.9.8 Convergence Proof of DERPI in Theorem 1

Proof of Distribution-Entropy-Regularized Policy Evaluation in Lemma 1

Lemma 1(Distribution-Entropy-Regularized Policy Evaluation) Consider the distribution-entropy-regularized Bellman operator $\mathcal{T}_d^\pi$ in Eq. 2.8 and assume $\mathcal{H}(\mu^{s_t,a_t}, q_\theta^{s_t,a_t})$ is bounded for all $(s_t, a_t) \in \mathcal{S} \times \mathcal{A}$. Define $Q^{k+1} = \mathcal{T}_d^\pi Q^k$, then Q^{k+1} will converge to a *corrected* Q-value of π as $k \to \infty$ with the new objective function $J'(\pi)$ defined as

$$J'(\pi) = \sum_{t=0}^{T} \mathbb{E}_{(s_t,a_t)\sim\rho_\pi} \left[r\left(s_t, a_t\right) + \gamma f(\mathcal{H}\left(\mu^{s_t,a_t}, q_\theta^{s_t,a_t}\right)) \right]. \tag{2.37}$$

Proof. Firstly, we plug in $V(s_{t+1})$ into RHS of the iteration in Eq. 2.8, then we obtain

$$\begin{aligned}
&\mathcal{T}_d^\pi Q\left(s_t, a_t\right) \\
&= r\left(s_t, a_t\right) + \gamma \mathbb{E}_{s_{t+1}\sim P(\cdot|s_t,a_t)} \left[V\left(s_{t+1}\right) \right] \\
&= r\left(s_t, a_t\right) + \gamma f(\mathcal{H}\left(\mu^{s_t,a_t}, q_\theta^{s_t,a_t}\right)) + \gamma \mathbb{E}_{(s_{t+1},a_{t+1})\sim\rho^\pi} \left[Q\left(s_{t+1}, a_{t+1}\right) \right] \\
&\triangleq r_\pi\left(s_t, a_t\right) + \gamma \mathbb{E}_{(s_{t+1},a_{t+1})\sim\rho^\pi} \left[Q\left(s_{t+1}, a_{t+1}\right) \right],
\end{aligned} \tag{2.38}$$

where $r_\pi\left(s_t, a_t\right) \triangleq r\left(s_t, a_t\right) + \gamma f(\mathcal{H}\left(\mu^{s_t,a_t}, q_\theta^{s_t,a_t}\right))$ is the entropy augmented reward we redefine. Applying the standard convergence results for policy evaluation [102], we can attain that this Bellman updating under $\mathcal{T}_d^\pi$ is convergent under the assumption of $|\mathcal{A}| < \infty$ and bounded entropy augmented rewards r_π. □

Policy Improvement with Proof

Lemma 2. *(Distribution-Entropy-Regularized Policy Improvement) Let* $\pi \in \Pi$ *and a new policy* π_{new} *be updated via the policy improvement step in the policy optimization. Then* $Q^{\pi_{new}}\left(s_t, a_t\right) \geq Q^{\pi_{old}}\left(s_t, a_t\right)$ *for all* $\left(s_t, a_t\right) \in \mathcal{S} \times \mathcal{A}$ *with* $|\mathcal{A}| \leq \infty$.

Proof. The policy improvement in Lemma 2 implies that $\mathbb{E}_{a_t \sim \pi_{\text{new}}}\left[Q^{\pi_{\text{old}}}(s_t, a_t)\right] \geq \mathbb{E}_{a_t \sim \pi_{\text{old}}}\left[Q^{\pi_{\text{old}}}(s_t, a_t)\right]$, we consider the Bellman equation via the distribution-entropy-regularized Bellman operator $\mathcal{T}_{sd}^{\pi}$:

$$
\begin{aligned}
Q^{\pi_{\text{old}}}\left(s_t, a_t\right) & \triangleq r\left(s_t, a_t\right)+\gamma \mathbb{E}_{s_{t+1} \sim \rho}\left[V^{\pi_{\text{old}}}\left(s_{t+1}\right)\right] \\
& =r\left(s_t, a_t\right)+\gamma f(\mathcal{H}\left(\mu^{s_t, a_t}, q_\theta^{s_t, a_t}\right))+\gamma \mathbb{E}_{\left(s_{t+1}, a_{t+1}\right) \sim \rho^{\pi_{\text{old}}}}\left[Q^{\pi_{\text{old}}}\left(s_{t+1}, a_{t+1}\right)\right] \\
& \leq r\left(s_t, a_t\right)+\gamma f(\mathcal{H}\left(\mu^{s_t, a_t}, q_\theta^{s_t, a_t}\right))+\gamma \mathbb{E}_{\left(s_{t+1}, a_{t+1}\right) \sim \rho^{\pi_{\text{new}}}}\left[Q^{\pi_{\text{old}}}\left(s_{t+1}, a_{t+1}\right)\right] \\
& =r_{\pi^{\text{new}}}\left(s_t, a_t\right)+\gamma \mathbb{E}_{\left(s_{t+1}, a_{t+1}\right) \sim \rho^{\pi_{\text{new}}}}\left[Q^{\pi_{\text{old}}}\left(s_{t+1}, a_{t+1}\right)\right] \\
& \vdots \\
& \leq Q^{\pi_{\text{new}}}\left(s_{t+1}, a_{t+1}\right),
\end{aligned}
\tag{2.39}
$$

where we have repeated expanded $Q^{\pi_{\text{old}}}$ on the RHS by applying the distribution-entropy-regularized distributional Bellman operator. Convergence to $Q^{\pi_{\text{new}}}$ follows from Lemma 1. □

Proof of DERPI in Theorem 1 **Theorem 1** (Distribution-Entropy-Regularized Policy Iteration) Repeatedly applying distribution-entropy-regularized policy evaluation in Eq. 2.8 and the policy improvement, the policy converges to an optimal policy π^* such that $Q^{\pi^*}\left(s_t, a_t\right) \geq Q^{\pi}\left(s_t, a_t\right)$ for all $\pi \in \Pi$.

Proof. The proof is similar to soft policy iteration [43]. For completeness, we provide the proof here. By Lemma 2, as the number of iteration increases, the sequence Q^{π_i} at i-th iteration is monotonically increasing. Since we assume the uncertainty-aware entropy is bounded, the Q^{π} is thus bounded as the rewards are bounded. Hence, the sequence will converge to some π^*. Further, we prove that π^* is in fact optimal. At the convergence point, for all $\pi \in \Pi$, it must be

case that:

$$\mathbb{E}_{a_t\sim\pi^*}\left[Q^{\pi_{\text{old}}}\ (s_t,a_t)\right] \geq \mathbb{E}_{a_t\sim\pi}\left[Q^{\pi_{\text{old}}}\ (s_t,a_t)\right].$$

According to the proof in Lemma 2, we can attain $Q^{\pi^*}(s_t,a_t) > Q^{\pi}(s_t,a_t)$ for (s_t,a_t). That is to say, the "corrected" value function of any other policy in Π is lower than the converged policy, indicating that π^* is optimal. □

2.9.9 Proof of Interpolation Form of $\hat{J}_q(\theta)$

In SAC [43] (Section 4.2), it introduces another parameterized state value function to approximate the soft value in the function approximation setting. Instead, we are not intended to do so, but directly use a single Q network to be optimized, which allows the interpolation form of our algorithm. In particular, we directly evaluate the least squared loss between the current Q estimates and the target ones for the critic loss. With a particular form of $f_\pi(\mathcal{H})$, the removal of the interaction term, and the replacement of Q_θ with $\mathbb{E}\left[q_\theta\right]$, we can derive the interpolation form of $\hat{J}_q(\theta)$ according to the following formula:

$$\begin{aligned}
\hat{J}_q(\theta) &= \mathbb{E}_{s,a}\left[\left(\mathcal{T}_d^\pi Q_{\theta^*}(s,a) - Q_\theta(s,a)\right)^2\right] \\
&= \mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi Q_{\theta^*}(s,a) - Q_\theta(s,a) + \gamma(\tau^{1/2}\mathcal{H}^{1/2}(\mu^{s,a},q_\theta^{s,a})/\gamma)\right)^2\right] \\
&= \mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi\mathbb{E}\left[q_{\theta^*}(s,a)\right] - \mathbb{E}\left[q_\theta(s,a)\right]\right)^2\right] + \tau\mathbb{E}_{s,a}\left[\mathcal{H}(\mu^{s,a},q_\theta^{s,a})\right] \\
&+ \mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi\mathbb{E}\left[q_{\theta^*}(s,a)\right] - \mathbb{E}\left[q_\theta(s,a)\right]\right)\mathcal{H}(\mu^{s,a},q_\theta^{s,a})\right] \\
&\approx \mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi\mathbb{E}\left[q_{\theta^*}(s,a)\right] - \mathbb{E}\left[q_\theta(s,a)\right]\right)^2\right] + \tau\mathbb{E}_{s,a}\left[\mathcal{H}(\mu^{s,a},q_\theta^{s,a})\right] \\
&\propto (1-\lambda)\mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi\mathbb{E}\left[q_{\theta^*}(s,a)\right] - \mathbb{E}\left[q_\theta(s,a)\right]\right)^2\right] + \lambda\mathbb{E}_{s,a}\left[\mathcal{H}(\mu^{s,a},q_\theta^{s,a})\right],
\end{aligned} \tag{2.40}$$

where the second equation is based on the definition of Distribution-Entropy-Regularized Bellman Operator $\mathcal{T}_d^\pi$ in Eq. 2.8 and let $f(\mathcal{H}) = (\tau\mathcal{H})^{1/2}/\gamma$. The interaction term $+\mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi\mathbb{E}\left[q_{\theta^*}(s,a)\right] - \mathbb{E}\left[q_\theta(s,a)\right]\right)\mathcal{H}(\mu^{s,a},q_\theta^{s,a})\right]$ equal zero in the last equation is rooted in Lemma 1 in [92]. Although Lemma 1 considers the A/B testing with offline dataset, it demonstrates that the estimation equation between the Bellman error and and any function $\varphi\left(S_t,A_t\right)$ equals zero under mild conditions, such as the consistency assumption. Strictly speaking, we heuristically extend the conclusion in Lemma 1 of [92] to the simplification

of our critic loss, where we let $\varphi(S_t, A_t) = \mathcal{H}(\mu^{S_t,A_t}, q_\theta^{S_t,A_t})$. Consequently, we can approximately remove the interaction term as

$$\mathbb{E}_{s,a}\left[\left(\mathcal{T}^\pi \mathbb{E}\left[q_{\theta^*}(s,a)\right] - \mathbb{E}\left[q_\theta(s,a)\right]\right)\mathcal{H}(\mu^{s,a}, q_\theta^{s,a})\right] = 0. \tag{2.41}$$

We set $\lambda = \frac{\tau}{1+\tau} \in [0,1]$. Another simplification is that we directly use $\mathbb{E}[q_\theta]$ to replace Q_θ rather than to maintain both two networks q_θ and Q_θ with different parameters θ. This strategy simplifies our implementation and contributes to derive the final interpolation form in $\hat{J}_q(\theta)$.

2.9.10 Implementation Details

Replacing ϵ with the ratio ε for Visualization The substitution of ϵ with ε is for convenience in the implementation. As Proposition 1 elucidates, the return density decomposition requires that ϵ exceed certain thresholds to ensure the resultant decomposed $\widehat{\mu}^{s,a}$ qualifies as a valid density function. In practice, pinpointing this lower boundary for ϵ in each iteration to regulate its range could be prohibitively time-intensive. A more pragmatic approach involves redistributing the mass from the bin that contains the expectation to other bins in specified ratios, thereby introducing the corresponding ratio term ε. By varying ε from 0 to 1, it invariably meets the validity condition outlined in Proposition 1, thereby streamlining the process for conducting ablation studies concerning $\widehat{\mu}^{s,a}$ as demonstrated in Figure 2.2.

To delineate the relationship between the ratio ε and the coefficient ϵ in constructing $\widehat{\mu}^{s,a}$, after some calculations we establish their equivalence as follows:

$$\varepsilon = \frac{p_E - (1-\epsilon)}{p_E \epsilon}, \tag{2.42}$$

where p_E represents the weighting assigned to the bin Δ_E as specified in Proposition 1. The resulting $\varepsilon \in [0,1]$ has a monotonically increasing relationship with ϵ, which facilitates the visualization without undermining our conclusion. Please refer to the code in the implementation for more details.

Hyper-parameters and Network structure Our implementation is directly adapted from the source code in [67]. For Distributional SAC with C51, we use 51 atoms similar to the C51 [9]. For distributional SAC with quantile regression, instead of using fixed quantiles in QR-DQN, we leverage the quantile fraction generation based on IQN [21] that uniformly samples quantile fractions in order to approximate the full quantile function. In particular, we fix the number of quantile fractions as N and keep them in ascending

Table 2.1: Hyper-parameters Sheet.

Hyperparameter	Value
Shared	
Policy network learning rate	3e-4
(Quantile) Value network learning rate	3e-4
Optimization	Adam
Discount factor	0.99
Target smoothing	5e-3
Batch size	256
Replay buffer size	1e6
Minimum steps before training	1e4
DSAC with C51	
Number of Atoms (N)	51
DSAC with IQN	
Number of quantile fractions (N)	32
Quantile fraction embedding size	64
Huber regression threshold	1

Hyperparameter	Temperature Parameter β	Max episode lenght
Walker2d-v2	0.2	1000
Swimmer-v2	0.2	1000
Reacher-v2	0.2	1000
Ant-v2	0.2	1000
HalfCheetah-v2	0.2	1000
Humanoid-v2	0.05	1000
HumanoidStandup-v2	0.05	1000
BipedalWalkerHardcore-v2	0.002	2000

order. Besides, we adapt the sampling as $\tau_0 = 0, \tau_i = e_i / \sum_{i=0}^{N-1} e_i$, where $\epsilon_i \in U[0,1], i = 1, ..., N$. We adopt the same hyper-parameters, which are listed in Table 2.1 and network structure as in the original distributional SAC paper [67].

2.9.11 DERAC Algorithm

We provide a detailed algorithm description of DERAC algorithm in Algorithm 1.

Initialize two value networks q_θ, q_{θ^*}, and policy network π_ϕ.
for each iteration **do**
  for each environment step **do**
    $a_t \sim \pi_\phi(a_t|s_t)$.
    $s_{t+1} \sim p(s_{t+1}|s_t, a_t)$.
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r(s_t, a_t), s_{t+1})\}$
  end for
  for each gradient step **do**
    $\theta \leftarrow \theta - \lambda_q \nabla_\theta \hat{J}_q(\theta)$
    $\phi \leftarrow \phi + \lambda_\pi \nabla_\phi \hat{J}_\pi(\phi)$.
    $\theta^* \leftarrow \tau\theta + (1-\tau)\theta^*$
  end for
end for

Algorithm 1: Distribution-Entropy-Regularized Actor Critic (DERAC) Algorithm

2.9.12 Experiments Results

Uncertainty-aware Regularization Effect via Ablation Study in Actor Critic We study the uncertainty-aware regularization effect from being categorical distributional in the actor-critic framework, where we decompose the C51 critic loss in distributional SAC (DSAC) according to Eq. 2.4. We denote the decomposed DSAC (C51) with different ε as $\mathcal{H}(\mu, q_\theta)(\varepsilon = 0.8/0.5/0.1)$. As suggested in Figure 2.5, the performance of $\mathcal{H}(\mu, q_\theta)$ tends to vary from the vanilla DSAC (C51) to SAC with the decreasing of ε on three MuJoCo environments, except bipedalwalkerhardcore. In bipedalwalkerhardcore. this tendency may not be clear, as we hypothesis that the algorithm

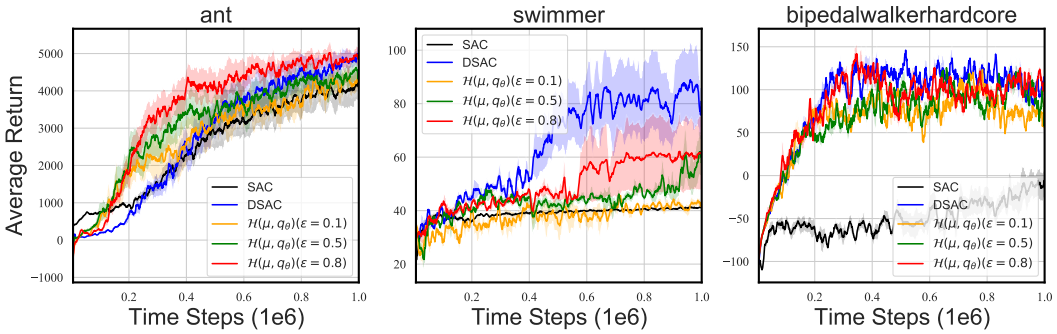


Figure 2.5: Learning curves of Distributional AC (C51) with the return distribution decomposition $\mathcal{H}(\mu, q_\theta)$ under different ε.

performance is not sensitive when ε changes within this restricted range, although this range is designed to guarantee a valid density decomposition. It is worth noting that our return density decomposition is valid only when $\epsilon \geq 1 - p_E$ as shown in Proposition 1, and therefore ϵ can not strictly go to 0, where $\mathcal{H}(\mu, q_\theta)$ would degenerate to SAC ideally. In addition, compared with the ablation study in Figure 2.2, the trend varying from DSAC to SAC by decreasing ε may not be as pronounced as that in value-based RL evaluated on Atari games. This is because the actor-critic architecture is generally perceived to be more prone to instability compared to value-based learning in RL. As outlined in [33], this instability stems from the policy updates, which may introduce additional bias or variance from the critic learning process.

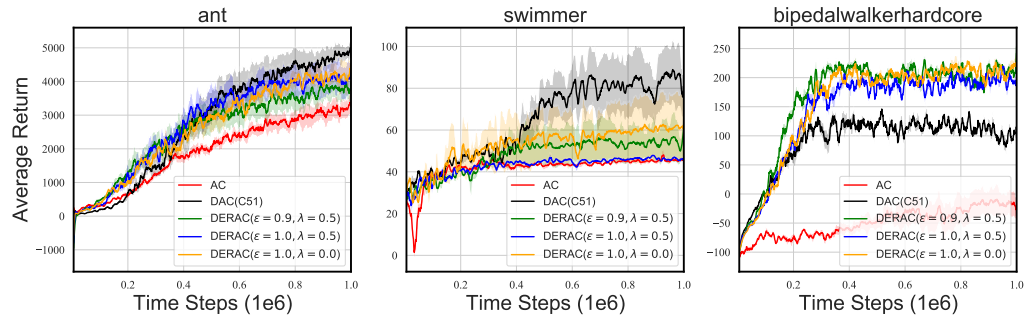


Figure 2.6: Learning curves of DERAC algorithms across different λ and ε on three MuJoCo environments over 5 seeds.

Sensitivity Analysis of DERAC Figure 2.6 shows that DERAC with different λ in Eq. 2.11 may behave differently in different environments. In general, DERAC with different ε and λ perform similarly to DERAC, with an interpolation nature between AC and DAC (C51). Notably, DERAC with different ε and λ still surpasses at both AC and DAC (C51) in bidedalwalker-hardcore, demonstrating the robust superiority of DERAC algorithm.

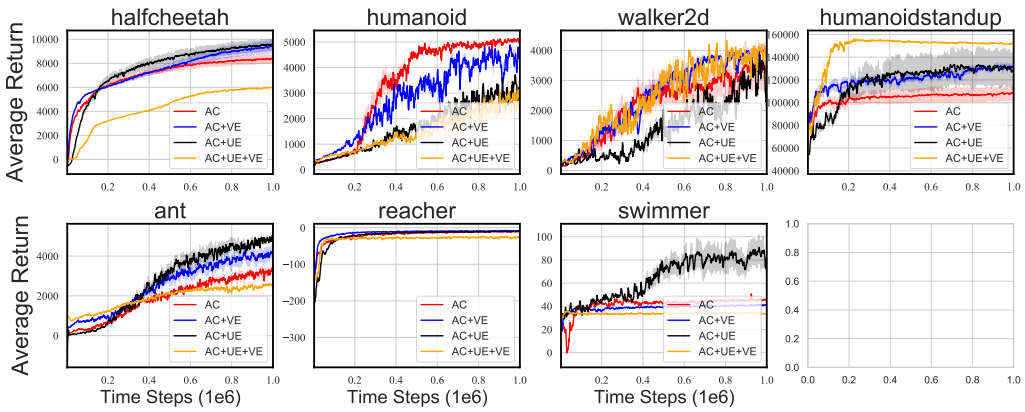


Figure 2.7: Learning curves of *AC*, *AC+VE* (SAC), *AC+UE* (DAC) and AC+UE+VE (DSAC) over 5 seeds across seven MuJoCo environments where distributional RL part is based on C51. Walker 2d and Humanoidstandup: Mutual Improvement. Others: Potential Interference.

Mutual Impacts on DSAC (C51) We presents results on seven MuJoCo environments and omits Bipedalwalkerhardcore due to some engineering issue when the C51 algorithm interacts with the simulator. Figures 2.7 showcases that the simultaneous leverage of uncertainty-aware and vanilla entropy regularization renders a mutual improvement on humanoidstandup and Walker2d. In contrast, the two regularization when employed together lead to a performance degradation in other environments, especially in swimmer and halfcheetah, where *AC+UE+VE* is significantly inferior to ***AC+UE*** or *AC+VE*.

Chapter 3

How Does Return Distribution in Distributional Reinforcement Learning Help Optimization?

3.1 Abstract

Distributional reinforcement learning (RL), which learns the whole return distribution compared with only its expectation in classical RL, has achieved great success in obtaining superior performance. However, we still have a poor understanding of how the return distribution in distributional RL works. In this study, we analyze the optimization benefits of distributional RL by leveraging its additional return distribution information over classical RL in the Neural Fitted Z-Iteration (Neural FZI) framework. To begin with, we demonstrate that the distribution loss of distributional RL has desirable smoothness characteristics and hence enjoys stable gradients, which is in line with its tendency to promote optimization stability. Furthermore, the acceleration effect of distributional RL is revealed by decomposing the return distribution. It shows that distributional RL can perform favorably if the return distribution approximation is appropriate, measured by the variance of gradient estimates in each environment. Rigorous experiments validate the stable optimization behaviors of distributional RL and its acceleration effects compared to classical RL. Our

research findings illuminate how the return distribution in distributional RL algorithms helps the optimization.

3.2 Introduction

Motivation. Despite the remarkable empirical success of distributional RL, the illumination of its theoretical advantages still needs to be studied. A distributional regularization effect [100] stemming from the additional return distribution knowledge has been characterized to explain the superiority of distributional RL over classical RL, but the benefit of the proposed regularization on the optimization of algorithms has not been further investigated. Such a gap inspires us to investigate the optimization impact of distributional RL by leveraging the full return distribution knowledge. However, existing literature [71, 96] that helps to analyze the optimization of RL learning may not apply to practical distributional RL algorithms as there still remains a gap between the theory and practice in RL.

In this paper, we study the optimization advantages of distributional RL over classical RL. Within the Neural FZI framework, our optimization analysis can not only sufficiently characterize offline distributional RL behaviors but also approximate the online setting. Within this framework, we study the uniform stability of distributional loss based on categorical parameterization. Owing to the smoothness properties of distributional loss, distributional RL algorithms tend to satisfy the uniform stability in the optimization process, thus enjoying stable gradient behaviors in the input space. In addition to the optimization stability, we also elaborate on the acceleration effect of distributional RL algorithms based on the return density decomposition technique proposed recently. Distributional RL can speed up the convergence and perform favorably if the return distribution is approximated appropriately, measured by the gradient estimates' variance. Empirical results corroborate that distributional RL possesses stable gradient behaviors and acceleration effects by suggesting smaller gradient norms concerning the states and model parameters. Our study opens up many exciting research pathways in this domain through the lens of optimization, paving the way for future investiga-

tions to reveal more advantages of distributional RL. Our contributions can be summarized as follows:

- We specifically study the optimization advantage of practical distributional RL algorithms with the general function approximators. Within the Neural FZI framework, we can analyze the optimization properties of distributional RL by establishing its connection with supervised learning.
- We reveal the uniform stability of distributional RL thanks to the smoothness properties of distributional loss. By contrast, classical RL may not guarantee such a stable optimization property due to the sensitivity of the least squared loss.
- The acceleration effects of distributional RL have also been demonstrated through the return density decomposition. We show that distributional RL can speed up convergence if the parameterization error of the return distribution is appropriate.

3.3 Related Work

Interpretation of distributional RL. Interpreting the behavior difference between distributional and classical RL was initially studied using the coupled updates method in [66]. They conclude that both distributional and classical RL behave the same in the tabular and linear approximation settings and attribute the superiority of distributional RL to its non-linear approximation. However, the coupled methodology mainly investigated preserving the expectation of return distribution to measure the behavior differences, which rules out other factors, including the optimization effect due to the distributional loss [51]. An implicit risk-sensitive entropy regularization was then revealed in distributional RL by [100], without further analyzing its optimization benefits. Our work complements and extends their results through the lens of optimization.

Convergence and Acceleration in RL. Existing optimization analysis in RL is mainly based on the policy gradient methods, such as the Actor

Critic framework [102]. [71] shows that the policy gradient with a softmax parameterization converges at a $\mathcal{O}(1/t)$ rate, which significantly expands the existing asymptotic convergence results. Entropy regularization [42, 43] has gained increasing attention and [4] provides a fine-grained understanding of the impact of entropy on policy optimization and emphasizes that any strategy, such as entropy regularization, can only affect learning in one of two ways: either it reduces the noise in the gradient estimates or it changes the optimization landscape. The seemingly applicable analysis framework on value-based RL is PAC-MDP [96], which effectively analyzes the convergence of typical RL algorithms in the tabular setting. However, it is unclear whether this analysis applies to practical distributional RL algorithms. By contrast, our optimization is within a more interpretable Neural FZI framework and focuses on accelerating the distributional RL algorithm.

Stable Optimization. Stable optimization is one of the crucial properties for RL algorithms, and common strategies include Batch Normalization [89], Spectral Normalization [73], gradient penalty [41]. In RL, stable optimization techniques [38, 58] also benefit the training and the final performance. By contrast, we show that (categorical) distributional RL naturally enjoys stable optimization compared with classical RL.

3.4 Optimization Analysis of Distributional RL

Under Neural FZI established in Section 3.4.1, we analyze two optimization aspects of distributional RL based on the categorical parameterization, including the stable optimization from the loss function in Section 3.4.2, and its acceleration effect determined by the gradient estimate variance in Section 3.4.3.

Notations. In CDRL, we denote the categorical distribution $\hat{\eta} = \sum_{i=1}^{k} f_i \delta_{l_i}$ to approximates the action-state return distribution η, where $l_1, l_2, ..., l_k$ is a set of fixed supports and $\{f_i\}_{i=1}^{k}$ are learnable probabilities, normally parameterized by a neural network.

3.4.1 Optimization Analysis for Distributional RL within Neural Fitted Z-Iteration

Approximate Supervised Learning within Neural FZI to Allow the Optimization Analysis. We conduct our analysis in this chapter still within the Neural FZI framework in Eq. 2.3 established in Chapter 2. Previous optimization analysis focuses on either policy gradient methods [71, 1] or the sample complexity in the tabular setting [96]. However, there remains some gap between the theory and the practical neural network parameterized RL algorithm, and the previous results may not be directly attainable for the optimization analysis of distributional RL. By contrast, Neural FZI simplifies the optimization problem in deep RL into an approximate iterative supervised learning on a local fixed offline dataset by leveraging experience buffer and target networks, allowing richer optimization analysis. It sufficiently characterizes the offline behaviors of practical distributional RL algorithms and can also approximate online algorithms. In particular, Neural FZI does not consider the exploration; the data distribution shift caused by exploration from an ϵ-greedy policy can be negligible in the online setting, *when the replay memory is sufficiently large or considering the short period.* Thus, the optimization in each phase of Neural FZI can be approximately viewed as supervised learning in contrast to PAC-MDP analysis [96] that involves the exploration impact.

Two Key Factors. The Neural FZI framework offers new insights to analyze the optimization benefits for practical distributional RL algorithms, within which there are mainly two crucial components.

- **Factor 1: the choice of d_p.** On the one hand, d_p determines the convergence rate of distributional Bellman update, i.e., the speed of outer iterations in Neural FZI. For instance, distributional Bellman operator under Crámer distance is $\sqrt{\gamma}$-contractive [10], γ-contractive under Wasserstein distance [9]. Moreover, d_p also largely affects the continuous optimization problem concerning parameters θ in Z_θ within each iteration of Neural FZI.

- **Factor 2: the parameterization of Z_θ.** Given the same d_p, a more informative parameterization can approximate the true return distribution more reasonably, promoting the optimization within each phase of Neural FZI. For example, with a more expressiveness power on quantile functions, IQN [21] outperforms QR-DQN [22] on a wider range of environments.

Remark. We mainly attribute the optimization benefit of distributional RL to the choice of distributional loss d_p in Neural FZI relative to the least squared loss in Neural FQI based on the same categorical parameterization on Z_θ, despite the different convergence rates under them.

Categorical Pameterization Equipped with KL Divergence. To allow for theoretical analysis, we resort to the histogram function [108, 51] as the density estimator of Z_θ, a continuous version of categorical parameterization with their equivalent proof provided in [100]. After incorporating the projection to redistribute probabilities of target return distribution by the neighboring smoothing proposed in CDRL, the target, and current histogram function estimators inherit the joint supports, based on which we apply KL divergence as d_p. In particular, we denote the histogram density estimator as $f^{s,a}$ with k uniform partitions on the support, denote $\mathbf{x}(s)$ as the state feature on each state s. We let the support of $Z(s,a)$ be uniformly partitioned into k bins. The output dimension of $f^{s,\cdot}$ can be $|\mathcal{A}| \times k$, where we use the index a to focus on the function $f^{s,a}$. Hence, the function $f^{s,a} : \mathcal{X} \to [0,1]^k$ provides a k-dimensional vector $f^{s,a}(\mathbf{x}(s))$ of the coefficients, indicating the probability that the target is in this bin given the state feature $\mathbf{x}(s)$ and action a. Next, we use *softmax* based on the linear approximation $\mathbf{x}(s)^\top \theta_i$ to express $f^{s,a}$, i.e., $f_i^{s,a,\theta}(\mathbf{x}(s)) = \exp\left(\mathbf{x}(s)^\top \theta_i\right) / \sum_{j=1}^k \exp\left(\mathbf{x}(s)^\top \theta_j\right)$. For simplicity, we use $f_i^\theta(\mathbf{x}(s))$ to replace $f_i^{s,a,\theta}(\mathbf{x}(s))$.

Categorical Distributional Loss. Note that the form of $f^{s,a}$ is similar to that in Softmax policy gradient optimization [71, 102], but we focus on the value-based RL rather than the policy gradient RL. Our prediction probability

$f_i^{s,a}$ is redefined as the probability in the i-th bin over the support of $Z(s,a)$, thus eventually serving as a density function. While the linear approximator is limited, this is the setting where, so far, the cleanest results can be firstly achieved, and understanding this setting is necessary for the first step towards bigger problems of understanding distributional RL algorithms. Under this categorical parameterization with KL divergence, the distributional objective function $\mathcal{L}_\theta(s,a)$ for the continuous optimization in each phase of Neural FZI (Eq. 2.3) can be expressed as:

$$\mathcal{L}_\theta(s,a) = -\sum_{i=1}^{k} \int_{z_i}^{z_i+w_i} p^{s,a}(y) \log \frac{f_i^\theta(\mathbf{x}(s))}{w_i} dy \propto -\sum_{i=1}^{k} p_i^{s,a} \log f_i^\theta(\mathbf{x}(s)), \tag{3.1}$$

where $\theta = \{\theta_1, ..., \theta_k\}$ and $p_i^{s,a}$ is the probability in the i-th bin of the true density function $p^{s,a}(x)$ for $Z(s,a)$ defined in Eq. 2.4. w_i is the width for the i-th bin $(z_i, z_{i+1}]$. The derivation of the categorical distributional loss under the categorical parameterization is given in Appendix 3.8.1.

3.4.2 Stable Optimization Analysis under Uniform Stability

Optimization Properties. Our stable optimization conclusions are based on the smoothness properties of categorical distributional loss in Eq. 3.1. A similar histogram loss was also analyzed in [51] along with a local Lipschitz constant analysis. By contrast, in Proposition 7, we extend their optimization results and further establish its connection with distributional RL.

Proposition 7. *(Properties of Categorical Distributional Loss) Assume the state features $\|\mathbf{x}(s)\|_2 \leq l$ for each state s, then $\mathcal{L}_\theta$ is kl-Lipschitz continuous, kl^2-smooth and convex w.r.t. the parameter θ.*

Please refer to Appendix 3.8.2 for the proof. The smoothness properties of categorical distributional loss d_p are the foundation for the stable optimization of distributional RL. In stark contrast, classical RL optimizes a least squared loss function [102] in Neural FQI. It is known that the least squared estimator has no bounded Lipschitz constant in general and is only $\lambda_{\max}$-smooth,

where $\lambda_{\max}$ is the largest singular value of the data matrix. Specifically, we have $\|\nabla_\theta \mathcal{L}_\theta\| \leq kl$ for the categorical distributional loss in distributional RL. By contrast, the gradient norm in classical RL is $|y_i - Q_\theta^k(s,a)|\|\mathbf{x}(s)\|$, where $Q_\theta^k(s,a) = \sum_{i=1}^k (z_i + z_{i+1}) f_i^\theta(\mathbf{x}(s))/2w_i$ under the same categorical parameterization for a fair comparison. Clearly, $Q_\theta^k(s,a)$ can be sufficiently large if the support $[z_0, z_k]$ is specified to be large, which is common in environments with a high level of expected returns [9]. As such, $|y_i - Q_\theta^k(s,a)|$ can vary significantly larger than k and classical RL with the potentially larger upper bound of gradient norms is prone to the instability optimization issue.

Uniform Stability of Distributional RL. As an application of stable analysis in [45], we next show that distributional RL loss can naturally induce a uniform stability property under the desirable smoothness properties in Proposition 7, while classical RL can not. We first recap the definition of uniform stability for an algorithm while running *Stochastic Gradient Descent* (SGD) in Definition 2.

Definition 2. *(Uniform Stability) [45] Consider a loss function $g_w(e)$ parameterized by w encountered on the example e, a randomized algorithm $\mathcal{M}$ is uniformly stable if for all data sets $\mathcal{D}, \mathcal{D}'$ such that $\mathcal{D}, \mathcal{D}'$ differ in at most one example, we have*

$$\sup_e \mathbb{E}_{\mathcal{M}} \left[g_{\mathcal{M}(\mathcal{D})}(e) - g_{\mathcal{M}(\mathcal{D}')}(e) \right] \leq \epsilon_{stab} \,. \tag{3.2}$$

Remark: Rationale of Uniform Stability Analysis. One may be concerned whether the uniform stability analysis is applicable to the RL setting with a gradually varying experience replay buffer. Thanks to the Neural FZI framework, it can be viewed as an approximate supervised learning on a nearly fixed offline dataset $\mathcal{D}$ with each iteration of Neural FZI, as the experiment replay allows nearly independent sampling on a fixed data distribution in a short period when the reply memory is large enough [28]. As such, the loss difference by varying the dataset for at most one sample can serve as a surrogate to measure the uniform stability for an algorithm in each phase of Neural FZI.

Theorem 3. *(Uniform Stability for Distributional RL) Suppose that we run SGD under $\mathcal{L}_\theta$ in Eq. 3.1 with step sizes $\lambda_t \leq 2/kl^2$ for T steps. Assume $\|\mathbf{x}(s)\| \leq l$ for each state s and action a, then we have $\mathcal{L}_\theta$ satisfies the uniform stability in Definition 2 with $\epsilon_{stab} \leq \frac{4kT}{n}$, i.e.,*

$$\mathbb{E}\left|\mathcal{L}_{\theta_T}(s,a) - \mathcal{L}_{\theta'_T}(s,a)\right| \leq \frac{4kT}{n}, \tag{3.3}$$

where θ_T and θ'_T are the minimizers after T steps under the dataset $\mathcal{D}$ and $\mathcal{D}'$, respectively.

Please refer to the proof of Theorem 3 in Appendix 3.8.3. Theorem 3 shows that while running SGD to solve the categorical distributional loss within each Neural FZI, the continuous optimization process in each iteration is ϵ_{stab}-uniformly stable with the stability errors shrinking at the rate of $O(n^{-1})$. The stable optimization has multiple advantages, including ϵ_{stab} - bounded generalization gap, a desirable local minimum in deep learning optimization literature [45], and improvement in performance in RL [11, 58]. By contrast, classical RL may not yield thestable optimization property without these smooth properties. For example, $\lambda_{\max}$-smooth may be of less help for the optimization given a bad conditional number of the design matrix where $\lambda_{\max}$ could be sufficiently large. Empirically, we validate the stable gradient behaviors, with smaller gradient norms in the input space, of CDRL compared with classical RL, and similar results are also observed in Quantile Regression distributional RL in Section 3.5.

Remark: Limitations. The potential optimization instability for classical RL can be used to partially explain its inferiority to distributional RL in most environments, although it may not explain why distributional RL could not perform favorably in certain games [14]. We leave the comprehensive explanation as future works.

Remark: Non-linear Categorical Parameterization. Although the stability above optimization conclusions are established on the linear categorical parameterization on Z^π, similar conclusions with a non-linear categorical pa-

rameterization can be naturally expected by non-convex optimization techniques proposed in [45]. We empirically validate our theoretical conclusions by directly applying practical neural network parameterized distributional RL algorithms.

3.4.3 Acceleration Effect of distributional RL

To characterize the acceleration effect of distributional RL, we additionally leverage the proposed *return density function decomposition* in Eq. 2.4 in Chapter 2, and then characterize the variance of the gradient estimates before providing the acceleration effect of distributional RL.

Measuring the Variance of Gradient Estimates. Within Neural FZI, our goal is to minimize $\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_\theta(s_i,a_i)$. We rewrite $\mathcal{L}_\theta(s,a)$ as $\mathcal{L}_\theta(g^{s,a},f_\theta^{s,a})$, where the target density function $g^{s,a}$ can be $p^{s,a}$, $\mu^{s,a}$ or $p_E^{s,a}$, and $f^{s,a,\theta}$ is rewritten as $f_\theta^{s,a}$ for conciseness. We denote $G^k(\theta)=\mathbb{E}\left[\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a})\right]$ and use $G(\theta)$ for $G^k(\theta)$ for simplicity. Based on Proposition 7 in Section 3.4.2, the appealing optimization properties concerning the parameter θ in f_θ still hold for $G(\theta)$. Although $p_E^{s,a}$ is a single-bin density without non-zero joint support as $f_\theta^{s,a}$, thanks to the leverage of target networks, the KL-based $\mathcal{L}_\theta$ would degrade to the cross-entropy loss, on which $\mathcal{L}_\theta$ is still well-defined. As the KL divergence has unbiased gradient estimates, we let the variance of its stochastic gradient over the expectation-related term $p_E^{s,a}$ be bounded, i.e.,

$$\mathbb{E}_{(s,a)\sim\rho^\pi}\left[\|\nabla\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a}))-\nabla G(\theta)\|^2\right]=\sigma^2. \tag{3.4}$$

Next, following the similar label smoothing analysis in [114], we further characterize the approximation degree of $f_\theta^{s,a}$ to the target return distribution $\mu^{s,a}$ by measuring its variance as $\kappa\sigma^2$:

$$\mathbb{E}_{(s,a)\sim\rho^\pi}\left[\|\nabla\mathcal{L}_\theta(\mu^{s,a},f_\theta^{s,a}))-\nabla G(\theta)\|^2\right]=\hat{\sigma}^2:=\kappa\sigma^2. \tag{3.5}$$

Notably, κ can be used to measure the approximation error between $f_\theta^{s,a}$ and $\mu^{s,a}$ and we do not assume $\hat{\sigma}^2$ to be bounded as κ can be arbitrarily large.

This expression $\kappa\sigma^2$ for $\hat{\sigma}^2$ allows us to utilize κ to characterize different acceleration effects for distributional RL given different κ. Concretely, a favorable approximation of $f_\theta^{s,a}$ to $\mu^{s,a}$, which coincides with the role of the Z_θ parameterization, will lead to a small κ, contributing to the acceleration effect of distributional RL as shown in Theorem 4.

Proposition 8. *Based on the return density decomposition in Eq. 2.4, and Eq. 3.5, we have:*

$$\mathbb{E}_{(s,a)\sim\rho^\pi}\left[\|\nabla\mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})) - \nabla G(\theta)\|^2\right] \leq (1-\epsilon)^2\sigma^2 + \epsilon^2\kappa\sigma^2. \tag{3.6}$$

Proposition 8 reveals the upper bound of gradient estimate variance for the whole target density function $p^{s,a}$, with proof in Appendix 3.8.4. Before comparing the sample complexity in optimizing both classical and distributional RL, we define the first-order τ-stationary point.

Definition 3. *(First-order τ-Stationary Point) When $\min_\theta G(\theta)$, the parameters θ_T after T steps is a first-order τ-stationary point if $\|\nabla_\theta G(\theta_T)\| \leq \tau$.*

Based on Definition 3, we formally characterize the acceleration effects for distributional RL in Theorem 4 that depends upon approximation errors between $\mu^{s,a}$ and $f_\theta^{s,a}$ measured by κ.

Theorem 4. *(Sample Complexity and Acceleration Effects of Distributional RL) While running SGD to minimize $\mathcal{L}_\theta$ in Eq. 3.1 within Neural FZI, we assume the step size $\lambda \leq \frac{1}{kl^2}\min\{1, \frac{\tau^2}{2\sigma^2}\}$, $\epsilon = 1/(1+\kappa)$, and the sample is uniformly drawn from T samples. Denote $G(\theta_0)$ as initialization.*
*(1) (**Classical RL**) The sample complexity $T = \frac{4G(\theta_0)}{\lambda\tau^2} = O(\frac{1}{\tau^4})$ when minimizing $\mathcal{L}_\theta(p_E^{s,a}, f_\theta^{s,a})$, such that $\mathcal{L}_\theta$ converges to a τ-stationary point.*
*(2) (**Distributional RL**) The sample complexity $T = O(\frac{1}{\tau^2})$ when minimizing $\mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})$, such that $\mathcal{L}_\theta$ converges to a $\max\{\tau, 2\sigma\kappa\}$-stationary point.*

The proof is provided in Appendix 3.8.5. Theorem 4 is inspired by the intuitive connection between the return distribution in distributional RL and the label distribution in label smoothing [114].

Interpretation of Theorem 4 . Theorem 4 demonstrates that optimizing the categorical distributional loss of distributional RL can speed up the convergence with the sample complexity from $O(\frac{1}{\tau^4})$ to $O(\frac{1}{\tau^2})$, if the distribution approximation error is favorable. In particular, when the agnostic κ determined by the environment satisfies $2\kappa\sigma \leq \tau$, the distributional RL algorithm has an effective return distribution parameterization for Z_θ with a smaller approximation error between $f_\theta^{s,a}$ and $\mu^{s,a}$ ($p^{s,a}$). In this case, the acceleration effect of distributional RL over classical RL can be guaranteed. However, it is not vice versa. When $2\kappa\sigma > \tau$, it is unclear whether the required sample complexity for distributional RL is higher than classical RL, as classical RL will require a lower sample complexity than $O(\frac{1}{\tau^4})$ to achieve a $2\kappa\sigma$-stationary point in this case. These theoretical results also coincide with past empirical observations [22, 14], where distributional RL algorithms outperform classical RL in most cases, but are inferior in certain environments. Based on our results in Theorem 4, we contend that these certain environments have much intrinsic uncertainty, the distribution parameterization error between Z_θ and the true return distribution under the distributional TD approximation is still too large ($\kappa > \frac{\tau}{2\sigma}$) to guarantee an acceleration effect as revealed in Theorem 4.

Smaller Gradient Norms in the Weight Space. The acceleration effect of distributional RL in Theorem 4 also implies that distributional RL tends to have smaller gradient norms concerning parameters than classical RL at the same training step, according to the definition of Lipschitz constant in terms of the first-order stationary point. The small gradient norms we analyze here are *in the weight space*, commonly used and directly linked with the convergence rate analysis. In contrast, the uniform stability analyzed in Section 3.4.2 is defined on the bounded loss difference that is strongly correlated to the gradient norms *in the input space*. Similar works include Spectral Normalization to stabilize the training of Generative Adversarial Networks [73] and RL [38], which normalizes the spectral norm of the weight matrix in each layer to lead to a one-valued Lipschitz constant concerning the input. We empirically demonstrate both of them in Section 3.5.

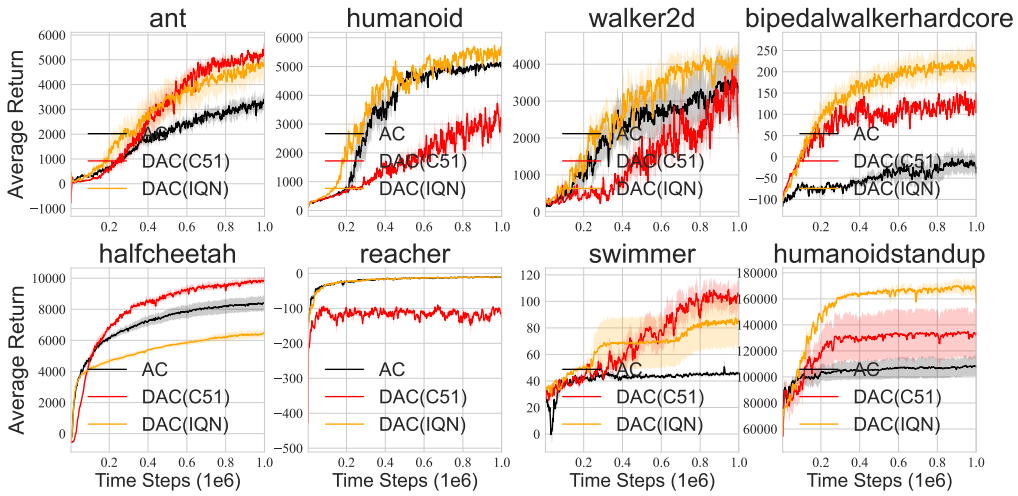


Figure 3.1: Performance. Learning curve of AC, DAC (C51), and DAC (IQN) over five seeds with smooth size five across eight MuJoCo games.

3.5 Experiments

Our experiments focus on the online distributional RL algorithms on continuous control Mujoco environments to demonstrate their stable gradient behaviors and acceleration effects.

Implementation. Our implementation is based Soft Actor Critic (SAC) [43] and distributional Soft Actor Critic [67]. We eliminate the optimization impact of entropy regularization in these algorithm implementations, and thus, we denote the resulting algorithms as Actor Critic (AC) and Distributional Actor Critic (DAC) for conciseness. For DAC, we first perform a categorical parameterized C51 critic loss from the classical least-squared critic loss dubbed DAC (C51), which coincides with our theoretical analysis in Sections 3.4.2 and 3.4.3. We further apply our experiments on Quantile Regression distributional RL, i.e., Implicit Quantile Network (IQN), denoted as DAC (IQN), to heuristically extend our conclusion in broader algorithm classes. More implementation details are provided in Appendix 3.8.6.

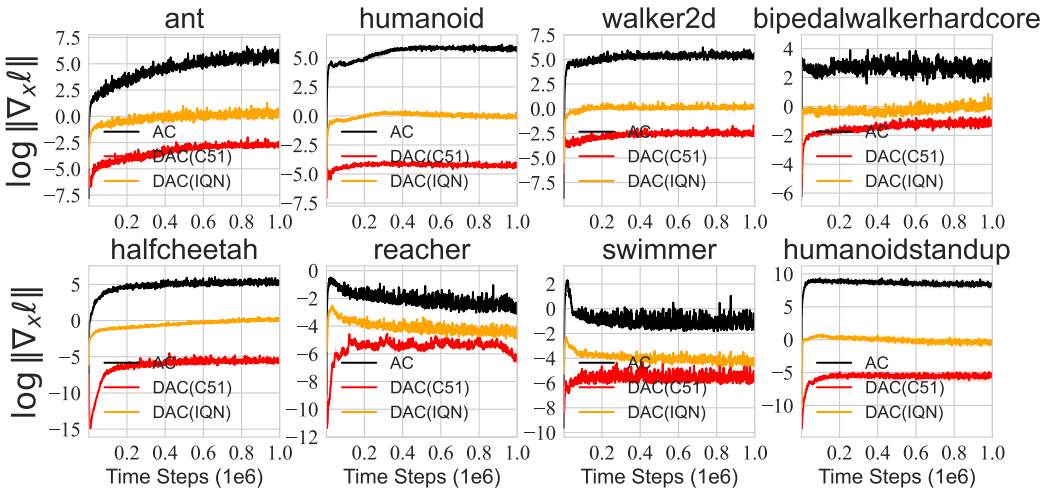


Figure 3.2: Uniform Stability. The critic gradient norms in the logarithmic scale regarding **the state** during the training of AC, DAC (C51), DAC (IQN) over 5 seeds on eight MuJoCo environments.

3.5.1 Performance and Uniform Stability

Figure 3.1 suggests both DAC (IQN) and DAC (C51) excel at the classical RL counterpart, i.e., AC (black lines), in most environments, which allows our further optimization analysis.

Proxy: Gradient Norms in the Input Space. We demonstrate the advantage of uniform optimization stability for distributional RL over classical RL. According to Theorem 3, the stable optimization of distribution loss within Neural FZI is described as a bounded loss difference for a neighboring dataset regarding each state s and action a. In other words, the error bound holds by taking the supreme over each state and action pair. To measure this algorithm stability, while far from perfect, we consider leveraging *the average gradient norms concerning the state feature* $\mathbf{x}(s)$ in the whole optimization process as the proxy. This is because the gradient magnitude in the input space could measure the sensitivity of the loss function regarding each state and action.

Results. Figure 3.2 suggests that both DAC (C51) and DAC (IQN) entail smaller gradient norm magnitudes than the classical AC (black lines) across all

environments, corroborating the uniform stability for distributional RL over classical RL analyzed in Theorem 3. As analyzed in Section 3.4.2, this result provides empirical evidence to interpret behaviors of distributional RL.

3.5.2 Acceleration Effect of Distributional RL

Proxy: Gradient Norms in the Weight Space . Theorem 4 implies that if the distribution parameterization is appropriate, distributional RL can speed up the convergence and thus can achieve better first-order stationary point, corresponding to smaller gradient norms given the time step in the learning process. To demonstrate it, we take the same step size for both DAC and AC, and evaluate the ℓ_2-norms of gradients *concerning network parameters* of their critics. A direct comparison between vanilla AC and DAC algorithm is given in Figure 3.3, despite the slight difference in the network architecture in the last layer. For an *apple-to-apple comparison*, we keep the same DAC architecture while implementing a variant AC by optimizing the expectation of represented return distribution. We also find a similar result in Appendix 3.8.7.

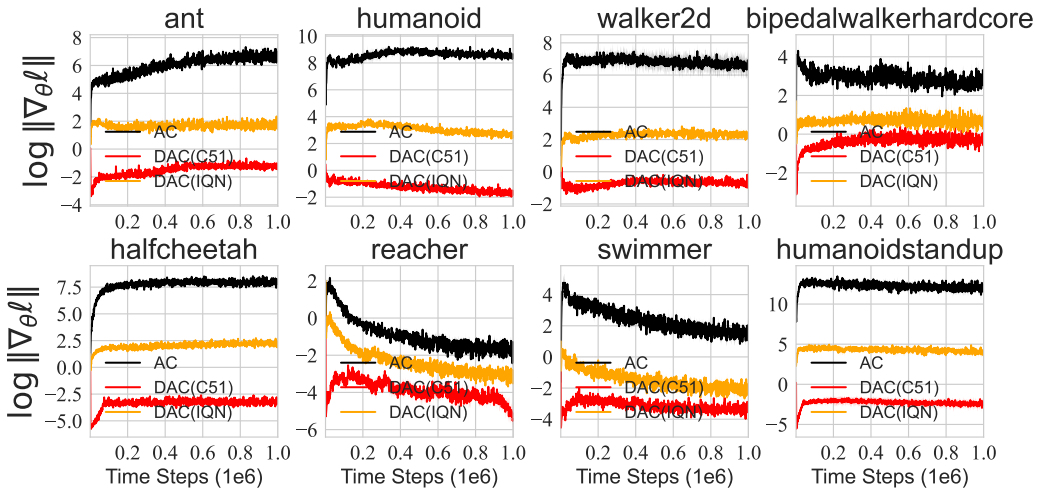


Figure 3.3: Acceleration Effect. The critic gradient norms in the logarithmic scale regarding **network parameters** in the training of AC, DAC (C51), DAC (IQN) over 5 seeds on MuJoCo environments.

Results. Figure 3.3 showcases that both DAC (C51) and DAC (IQN) have smaller gradient norms in terms of network parameters θ compared with AC in the whole optimization process. This result also validates that distributional RL loss tends to enjoy smoothness properties in Proposition 7. Moreover, it turns out that DAC (IQN) tends to have smaller gradient norms than DAC (C51). Given the fact that DAC (IQN) outperforms DAC (C51) in most environments in Figure 3.1, we hypothesize that DAC (IQN) may have a better acceleration effect than DAC (C51), contributing to explaining its superiority. Moreover, the more expressive parameterization of IQN over C51 is also helpful in interpreting both the acceleration and the improvement in the final performance. Lastly, according to Theorem 4, the access to the agnostic κ can serve as a sufficient condition to discriminate whether a specific distributional RL algorithm can accelerate the training in a given environment. However, a precise evaluation of κ is tricky, which we leave as valuable future work.

3.6 Conclusion

In our paper, we answer the question: *how does return distribution in distributional RL help the optimization* from perspectives of the uniform stability and acceleration effect in the optimization. Our conclusions are made within a new Neural FZI framework that connects the optimization results in supervised learning with practical deep RL algorithms.

3.7 Limitations and Future Work

Our optimization analysis of distributional RL is based on categorical parameterization, and therefore, some optimization properties, such as uniform stability, may not directly apply to other distributional RL families. The alternative analysis on distributional RL algorithms based on Wasserstein distance is also an integral and valuable complement to our conclusions, which we leave as future work.

3.8 Appendix

3.8.1 Derivation of Categorical Distributional Loss

We show the derivation details of the Categorical distribution loss starting from KL divergence between p and q_θ. p_i is the cumulative probability increment of target distribution $\{Y_i\}_{i\in[n]}$ within the i-th bin, and q_θ corresponds to a (normalized) histogram, and has density values $\frac{f_i^\theta(\mathbf{x}(s))}{w_i}$ per bin. Thus, we have:

$$\begin{aligned} D_{\mathrm{KL}}\left(p^{s,a}, q_\theta^{s,a}\right) &= \int_a^b p^{s,a}(y)\log p^{s,a}(y)dy - \int_a^b p^{s,a}(y)\log q_\theta^{s,a}(y)dy \\ &\propto -\int_a^b p^{s,a}(y)\log q_\theta^{s,a}(y)dy \\ &= -\sum_{i=1}^k \int_{z_i}^{z_i+w_i} p^{s,a}(y)\log\frac{f_i^\theta(\mathbf{x}(s))}{w_i}dy \\ &= -\sum_{i=1}^k \log\frac{f_i^\theta(\mathbf{x}(s))}{w_i}\underbrace{\left(F^{s,a}\left(z_i+w_i\right)-F^{s,a}\left(z_i\right)\right)}_{p_i^{s,a}} \\ &\propto -\sum_{i=1}^k p_i^{s,a}\log f_i^\theta(\mathbf{x}(s)) \end{aligned} \tag{3.7}$$

where the first $\propto$ results from the fixed target $p^{s,a}$ in the Neural FZI framework. The second equality is based on the categorical parameterization for the density function $q_\theta^{s,a}$. The last $\propto$ holds because the width parameter w_i can be ignored for this minimization problem.

3.8.2 Proof of Proposition 7

Proof. For the Categorical distributional loss below,

$$\mathcal{L}_\theta(s,a) = -\sum_{i=1}^k p_i^{s,a}\log f_i^\theta(\mathbf{x}(s)), \tag{3.8}$$

where $f_i^\theta(\mathbf{x}(s)) = \frac{\exp\left(\mathbf{x}(s)^\top\theta_i\right)}{\sum_{j=1}^k \exp\left(\mathbf{x}(s)^\top\theta_j\right)}$.

(1) Convexity. Note that $-\log \frac{\exp\left(\mathbf{x}(s)^\top \theta_i\right)}{\sum_{j=1}^k \exp\left(\mathbf{x}(s)^\top \theta_j\right)} = \log \sum_{j=1}^k \exp\left(\mathbf{x}(s)^\top \theta_j\right) - \mathbf{x}(s)^\top \theta_i$, the first term is Log-sum-exp, which is convex (see Convex optimization by Boyd and Vandenberghe), and the second term is affine function. Thus, $\mathcal{L}_\theta(s, a)$ is convex.

(2) $\mathcal{L}_\theta(s, a)$ is kl-Lipschitz continuous. We compute the gradient of the Histogram distributional loss regarding θ_i:

$$
\begin{aligned}
&\frac{\partial}{\partial \theta_i} \sum_{j=1}^k p_j^{s,a} \log f_j^\theta(\mathbf{x}(s)) \\
&= \sum_{j=1}^k p_j^{s,a} \frac{1}{f_j^\theta(\mathbf{x}(s))} \nabla_{\theta_i} f_j^\theta(\mathbf{x}(s)) \\
&= \sum_{j=1}^k p_j^{s,a} \frac{1}{f_j^\theta(\mathbf{x}(s))} f_i^\theta(\mathbf{x}(s))(\delta_{ij} - f_j^\theta(\mathbf{x}(s)))\mathbf{x}(s) \\
&= \left(p_i^{s,a}(1 - f_i^\theta(\mathbf{x}(s))) - \sum_{j \neq i}^k p_j^{s,a} f_i^\theta(\mathbf{x}(s)) \right) \mathbf{x}(s) \\
&= \left(p_i^{s,a} - p_i^{s,a} f_i^\theta(\mathbf{x}(s)) - (1 - p_i^{s,a}) f_i^\theta(\mathbf{x}(s)) \right) \mathbf{x}(s) \\
&= \left(p_i^{s,a} - f_i^\theta(\mathbf{x}(s)) \right) \mathbf{x}(s)
\end{aligned}
\tag{3.9}
$$

where $\delta_{ij} = 1$ if $i = j$, otherwise 0. Then, as we have $\|\mathbf{x}(s)\| \leq l$, we bound the norm of its gradient

$$
\begin{aligned}
\| \frac{\partial}{\partial \theta} \sum_{j=1}^k p_j \log f_j^\theta(\mathbf{x}(s)) \| &\leq \sum_{i=1}^k \| \frac{\partial}{\partial \theta_i} \sum_{j=1}^k p_j \log f_j^\theta(\mathbf{x}(s)) \| \\
&= \sum_{i=1}^k \| \left(p_i^{s,a} - f_i^\theta(\mathbf{x}(s)) \right) \mathbf{x}(s) \| \\
&\leq \sum_{i=1}^k |p_i^{s,a} - f_i^\theta(\mathbf{x}(s))| \|\mathbf{x}(s)\| \\
&\leq kl
\end{aligned}
\tag{3.10}
$$

The last equality satisfies because $|p_i - f_i^\theta(\mathbf{x}(s))|$ is less than 1 and even smaller. Therefore, we obtain that $\mathcal{L}_\theta$ is kl-Lipschitz.

(3) $\mathcal{L}_\theta$ is kl^2-Lipschitz smooth. A lemma is that $\log(1+\exp(x))$ is $\frac{1}{4}$-smooth as its second-order gradient is bounded by $\frac{1}{4}$, and if $g(w)$ is β-smooth w.r.t. w, then $g(\langle x, w\rangle)$ is $\beta\|x\|^2$-smooth. Based on this knowledge, we firstly focus on the 1-dimensional case of the function $\log f_j^\theta(z)$, where $f_j^\theta(z) = \frac{\exp z_j}{\sum_{i=1}^k \exp z_i}$. As we have derived, we know that $\frac{\partial}{\partial \theta_i} \log f_j^\theta(z_j) = \delta_{ij} - f_i^\theta(z_i)$. Then the second-order gradient is $\frac{\partial^2 \log f_j^\theta(z)}{\partial \theta_i \partial \theta_k} = -f_i^\theta(z)(\delta_{ik} - f_k^\theta(z)) = f_i^\theta(z)(f_k^\theta(z) - 1)$ if $i = k$, otherwise $f_i^\theta(z) f_k^\theta(z)$. Clearly, $|\frac{\partial^2 \log f_j^\theta(z)}{\partial \theta_i \partial \theta_k}| \leq 1$, which implies that $\log f_j^\theta(z)$ is 1-smooth. Thus, $\log f_j^\theta(\langle x, \theta_i\rangle)$ is $\|x\|^2$-smooth, or l^2-smooth. Further, $\sum_{j=1}^k p_j^{s,a} \log f_j^\theta(\mathbf{x}(s))$ is also l^2-smooth as we have

$$
\begin{aligned}
&\|\nabla_{\theta_i} \sum_{j=1}^k p_j^{s,a} \log f_j^\mu(\mathbf{x}(s)) - \nabla_{\theta_i} \sum_{j=1}^k p_j^{s,a} \log f_j^\nu(\mathbf{x}(s))\| \\
&\leq \sum_{j=1}^k p_j^{s,a} \|\nabla_{\theta_i} \log f_j^\mu(\mathbf{x}(s)) - \nabla_{\theta_i} \log f_j^\nu(\mathbf{x}(s))\| \\
&\leq \sum_{j=1}^k p_j^{s,a} \cdot l^2 \|\mu - \nu\| \\
&= l^2 \|\mu - \nu\|
\end{aligned}
\tag{3.11}
$$

for each parameter μ and ν. Therefore, we further have

$$
\begin{aligned}
&\|\nabla_\theta \sum_{j=1}^k p_j^{s,a} \log f_j^\mu(\mathbf{x}(s)) - \nabla_\theta \sum_{j=1}^k p_j^{s,a} \log f_j^\nu(\mathbf{x}(s))\| \\
&\leq \sum_{i=1}^k \|\nabla_{\theta_i} \sum_{j=1}^k p_j^{s,a} \log f_j^\mu(\mathbf{x}(s)) - \nabla_{\theta_i} \sum_{j=1}^k p_j^{s,a} \log f_j^\nu(\mathbf{x}(s))\| \\
&\leq \sum_{i=1}^k l^2 \|\mu - \nu\| \\
&= kl^2 \|\mu - \nu\|
\end{aligned}
\tag{3.12}
$$

Finally, we conclude that $\mathcal{L}_\theta(s, a)$ is kl^2-smooth.

□

3.8.3 Proof of Theorem 3

Proof. Consider the stochastic gradient descent rule as $G_{\lambda,\mathcal{L}}(\theta) = \theta - \lambda\nabla_\theta\mathcal{L}_\theta$. Firstly, we provide two definitions about $\mathcal{L}_\theta$ for the following proof.

Definition 4. *(σ-bounded) An update rule is σ-bounded if*

$$\sup_\theta \|\theta - \lambda\nabla_\theta\mathcal{L}_\theta\| \leq \sigma.$$

Definition 5. *(η-expansive) An update rule is η-expansive if*

$$\sup_{v,w} \frac{\|G_{\lambda,\mathcal{L}}(v) - G_{\lambda,\mathcal{L}}(w)\|}{\|u - w\|} \leq \eta.$$

Lemma 3. *(Grow Recursion, Lemma 2.5 [45]) Fix an arbitrary sequence of updates $G_1, ..., G_T$ and another sequence $G'_1, ..., G'_T$. Let $\theta_0 = \theta'_0$ be the starting point and define $\delta_t = \|\theta'_i - \theta_t\|$, where θ_t and θ'_t are defined recursively through*

$$\theta_{t+1} = G_{\lambda,\mathcal{L}}(\theta_t),\ \theta'_{t+1} = G'_{\lambda,\mathcal{L}}(\theta'_t)$$

Then we have the recurrence relation:

$$\delta_{t+1} \leq \begin{cases} \eta\delta_t & G_t = G'_t \text{ is } \eta\text{-expansive} \\ \min(\eta, 1)\delta_t + 2\sigma_t & G_t \text{ and } G'_t \text{ are } \sigma\text{-bounded }, G_t \text{ is } \eta \text{ expansive} \end{cases}$$

Lemma 4. *(Lipschitz Continuity) Assume $\mathcal{L}_\theta$ is L-Lipschitz, the gradient update $G_{\lambda,\mathcal{L}}$ is (λL)-bounded.*

Proof. $\|\theta - G_{\lambda,\mathcal{L}}(\theta)\| = \|\lambda\nabla_\theta\mathcal{L}_\theta\| \leq \lambda L$ □

Lemma 5. *(Lipschitz Smoothness and Convex) Assume $\mathcal{L}_\theta$ is β-smooth and convex, then for any $\lambda \leq \frac{2}{\beta}$, the gradient update $G_{\lambda,\mathcal{L}}$ is 1-expansive.*

Proof. Please refer to Lemma 3.7 in [45] for the proof. □

Based on all the results above, we start to prove Theorem 3. Our proof is largely based on [45], but it is applicable in distributional RL settings and considering desirable properties of histogram distributional loss. According to

Proposition 7, we attain that $\mathcal{L}_\theta$ is kl-Lipschitz as well as kl^2-smooth, and thus based on Lemma 4 and Lemma 5, we have $G_{\lambda,\mathcal{L}}$ is (λkl)-bounded, and 1-expansive if $\lambda \leq \frac{2}{kl^2}$. In the step t, SGD selects samples that are both in $\mathcal{D}$ and $\mathcal{D}'$, with probability $1 - \frac{1}{n}$. In this case, $G_t = G'_t$, and thus $\delta_{t+1} \leq \delta_t$ as G_t is 1-expansive based on Lemma 3. The other case is that samples selected are different with probability $\frac{1}{n}$, where $\delta_{t+1} \leq \delta_t + 2\lambda_t kl$ based on Lemma 3. Thus, if $\lambda_t \leq \frac{2}{kl^2}$, for each state s and action a, we have:

$$\begin{aligned}
\mathbb{E}\left|\mathcal{L}_{\theta_T}(s,a) - \mathcal{L}_{\theta'_T}(s,a)\right| &\leq kl\mathbb{E}\left[\delta_T\right], \text{ where } \delta_T = \|\theta_T - \theta'_T\| \\
&\leq kl\left((1-\frac{1}{n})\mathbb{E}\left[\delta_{T-1}\right] + \frac{1}{n}\mathbb{E}\left[\delta_{T-1}\right] + \frac{2\lambda_{T-1}kl}{n}\right) \\
&= kl\left(\mathbb{E}\left[\delta_{T-1}\right] + \frac{2\lambda_{T-1}kl}{n}\right) \\
&= kl\left(\mathbb{E}\left[\delta_0\right] + \sum_{t=0}^{T-1}\frac{2\lambda_t kl}{n}\right) \\
&\leq \frac{2k^2l^2}{n}\sum_{t=0}^{T-1}\frac{2}{kl^2} \\
&= \frac{4kT}{n}
\end{aligned} \tag{3.13}$$

Since this bound holds for all $\mathcal{D}$, $\mathcal{D}'$ and s, a, we attain the uniform stability in Definition 2 for our categorical distributional loss applied in distributional RL.

□

3.8.4 Proof of Proposition 8

$$\mathbb{E}_{(s,a)\sim\rho^\pi}\left[\|\nabla\mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})) - \nabla G(\theta)\|^2\right] \leq (1-\epsilon)^2\sigma^2 + \epsilon^2\kappa\sigma^2. \tag{3.14}$$

Proof. As we know that $p^{s,a}(x) = (1-\epsilon)p_E^{s,a} + \epsilon\mu^{s,a}(x)$ and we use KL divergence in $\mathcal{L}_\theta$, then we have:

$$\nabla\mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a}) = (1-\epsilon)\nabla\mathcal{L}_\theta(p_E^{s,a}, f_\theta^{s,a}) + \epsilon\nabla\mathcal{L}_\theta(\mu^{s,a}, f_\theta^{s,a})$$

Therefore,

$$\begin{aligned}
&\mathbb{E}_{(s,a)\sim\rho^\pi}\left[\|\nabla\mathcal{L}_\theta(p^{s,a},f_\theta^{s,a}))-\nabla G(\theta)\|^2\right]\\
&\leq \mathbb{E}_{(s,a)\sim\rho^\pi}\left[(1-\epsilon)^2\|\nabla\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a}))-\nabla G(\theta)\|^2+\epsilon^2\|\nabla\mathcal{L}_\theta(\mu^{s,a},f_\theta^{s,a}))-\nabla G(\theta)\|^2\right]\\
&=(1-\epsilon)^2\sigma^2+\epsilon^2\kappa\sigma^2,
\end{aligned} \tag{3.15}$$

where the first inequality uses the triangle inequality of norm, i.e., $\|(1-\epsilon)\mathbf{a}+\epsilon\mathbf{b}\|^2\leq(1-\epsilon)^2\|\mathbf{a}\|^2+\epsilon^2\|\mathbf{b}\|^2$, and the last equality uses the definition of the variance of $\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a})$ and $\mathcal{L}_\theta(\mu^{s,a},f_\theta^{s,a})$. □

3.8.5 Proof of Theorem 4

Proof. **Classical RL** (1) If we only consider the expectation of $Z^\pi(s,a)$, we use the information $p_E^{s,a}$ to construct the loss function. As $\mathcal{L}_\theta(p_E^{s,a},q_\theta^{s,a})$ is kl^2-smooth, we have

$$\begin{aligned}
G(\theta_{t+1})-G(\theta_t)&\leq\langle\nabla G(\theta_t),\theta_{t+1}-\theta_t\rangle+\frac{kl^2}{2}\|\theta_{t+1}-\theta_t\|^2\\
&=-\lambda\left\langle\nabla G(\theta_t),\nabla\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a})\right\rangle+\frac{kl^2\lambda^2}{2}\|\nabla\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a})\|^2
\end{aligned} \tag{3.16}$$

where the inequality is according to the definition of Lipschitz-smoothness, and the last equation is based on the updating rule of θ. Next, we take the expectation on both sides,

$$\begin{aligned}
&\mathbb{E}\left[G(\theta_{t+1})-G(\theta_t)\right]\\
&\leq-\lambda\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right]+\frac{kl^2\lambda^2}{2}\mathbb{E}\left[\|\nabla\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a})-\nabla G(\theta_t)+\nabla G(\theta_t)\|^2\right]\\
&\leq-\lambda\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right]+\frac{kl^2\lambda^2}{2}\mathbb{E}\left[\|\nabla\mathcal{L}_\theta(p_E^{s,a},f_\theta^{s,a})-\nabla G(\theta_t)\|^2\right]+\frac{kl^2\lambda^2}{2}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right]\\
&=\frac{\lambda(kl^2\lambda-2)}{2}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right]+\frac{kl^2\lambda^2}{2}\sigma^2\\
&\leq-\frac{\lambda}{2}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right]+\frac{kl^2\lambda^2}{2}\sigma^2
\end{aligned} \tag{3.17}$$

where the first two inequalities hold because $\nabla G(\theta) = \mathbb{E}\left[\nabla \mathcal{L}_\theta\right]$ and the last inequality comes from $\lambda \leq \frac{1}{kl^2}$. Through the summation, we obtain that

$$\mathbb{E}\left[G(\theta_T) - G(\theta_0)\right] \leq -\frac{\lambda}{2}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] + \frac{kl^2\lambda^2 T}{2}\sigma^2$$

We let $\mathbb{E}\left[G(\theta_T)\right] = 0$, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] \leq \frac{2G(\theta_0)}{\lambda T} + kl^2\lambda\sigma^2$$

By setting $\lambda \leq \frac{\tau^2}{2kl^2\sigma^2}$ (simultaneously $\lambda \leq \frac{1}{kl^2}$, i.e., $\lambda \leq \frac{1}{kl^2}\min\{1, \frac{\tau^2}{2\sigma^2}\}$) and $T = \frac{4G(\theta_0)}{\lambda\tau^2}$, we can have $\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] \leq \tau^2$, implying that the degenerated loss function based on the expectation $p_E^{s,a}$ can achieve τ-stationary point if the sample complexity $T = O(\frac{1}{\tau^4})$.

Distributional RL (2). We are still based on the kl^2-smoothness of $\mathcal{L}(p^{s,a}, f_\theta^{s,a})$.

$$\begin{aligned}
&G(\theta_{t+1}) - G(\theta_t)\\
&\leq \langle \nabla G(\theta_t), \theta_{t+1} - \theta_t\rangle + \frac{kl^2}{2}\|\theta_{t+1} - \theta_t\|^2\\
&= -\lambda\left\langle \nabla G(\theta_t), \nabla \mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})\right\rangle + \frac{kl^2\lambda^2}{2}\|\nabla \mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})\|^2\\
&= -\frac{\lambda}{2}\|\nabla G(\theta_t)\|^2 + \frac{\lambda}{2}\|\nabla G(\theta_t) - \nabla \mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})\|^2 + \frac{\lambda(kl^2\lambda - 1)}{2}\|\nabla \mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})\|^2\\
&\leq -\frac{\lambda}{2}\|\nabla G(\theta_t)\|^2 + \frac{\lambda}{2}\|\nabla G(\theta_t) - \nabla \mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})\|^2
\end{aligned} \tag{3.18}$$

where the second equation is based on $\langle \mathbf{a}, -\mathbf{b}\rangle = \frac{1}{2}\left(\|\mathbf{a} - \mathbf{b}\|^2 - \|\mathbf{a}\|^2 - \|\mathbf{b}\|^2\right)$, and the last inequality is according to $\lambda \leq \frac{1}{kl^2}$. After taking the expectation, we have

$$\begin{aligned}
\mathbb{E}\left[G(\theta_{t+1}) - G(\theta_t)\right] &\leq -\frac{\lambda}{2}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] + \frac{\lambda}{2}\mathbb{E}\left[\|\nabla G(\theta_t) - \nabla \mathcal{L}_\theta(p^{s,a}, f_\theta^{s,a})\|^2\right]\\
&\leq -\frac{\lambda}{2}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] + \frac{\lambda}{2}\left((1-\epsilon)^2\sigma^2 + \epsilon^2\kappa\sigma^2\right)
\end{aligned} \tag{3.19}$$

where the last inequality is based on Proposition 8. We take the summation,

and therefore,

$$\mathbb{E}\left[G(\theta_T) - G(\theta_0)\right] \leq -\frac{\lambda}{2}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] + \frac{T\lambda}{2}\left((1-\epsilon)^2\sigma^2 + \epsilon^2\kappa\sigma^2\right)$$

We let $\mathbb{E}\left[G(\theta_T)\right] = 0$ and $\epsilon = \frac{1}{1+\kappa}$, then,

$$\begin{aligned}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] &\leq \frac{2G(\theta_0)}{\lambda T} + (1-\epsilon)^2\sigma^2 + \epsilon^2\kappa\sigma^2 \\ &= \frac{2G(\theta_0)}{\lambda T} + \frac{2\kappa^2}{(1+\kappa)^2}\sigma^2 \\ &\leq \frac{2G(\theta_0)}{\lambda T} + 2\kappa^2\sigma^2\end{aligned} \tag{3.20}$$

If $\kappa \leq \frac{\tau}{2\sigma}$ and let $T = \frac{4G(\theta_0)}{\lambda\tau^2}$, this leads to $\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] \leq \tau^2$, i.e., τ-stationary point, with the sample complexity as $O(\frac{1}{\tau^2})$. If $\kappa > \frac{\tau}{2\sigma}$, we set $T = \frac{G(\theta_0)}{\lambda\kappa^2\sigma^2}$. This implies that $\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla G(\theta_t)\|^2\right] \leq 4\kappa^2\sigma^2$, which can only achieve $2\kappa\sigma$-stationary point. Putting two cases together, we conclude that distributional RL can achieve $\max\{\tau, 2\kappa\sigma\}$-stationary point (since τ can be pre-given, while $2\kappa\sigma$ is determined by the environment.) □

3.8.6 Implementation Details

Our implementation is directly adapted from the source code in [67]. For DAC (IQN), we consider the quantile regression for the distribution estimation on the critic loss. Instead of using fixed quantiles in QR-DQN [22], we leverage the quantile fraction generation based on IQN [21] that uniformly samples quantile fractions in order to approximate the full quantile function. In particular, we fix the number of quantile fractions as N and keep them ascending. Besides, we adapt the sampling as $\tau_0 = 0, \tau_i = \epsilon_i / \sum_{i=0}^{N-1}$, where $\epsilon_i \in U[0,1], i = 1, ..., N$.

Hyper-parameters and Network structure We adopt the same hyper-parameters listed in Table 3.1 and network structure as in the original distri-

butional SAC paper [67].

Best l_k for DAC (C51) As suggested in Table 3.1, after a line search for the hyperparameter tuning, we select l_k as 500, 10,000, 15,000, 160, 50, 5,000, 500, 500 for ant, halfcheetah, humanoidstand, swimmer, bipedalwalkerhardcore, humanoid, walker2d and reacher, respectively.

Table 3.1: Hyper-parameters Sheet.

Hyperparameter	Value
Shared	
Policy network learning rate	3e-4
(Quantile / Categorical) Value network learning rate	3e-4
Optimization	Adam
Discount factor	0.99
Target smoothing	5e-3
Batch size	256
Replay buffer size	1e6
Minimum steps before training	1e4
DAC (IQN)	
Number of quantile fractions (N)	32
Quantile fraction embedding size	64
Huber regression threshold	1
DAC (C51)	
Number of Atoms (k)	51

Hyperparameter	l_k for C51	Max episode lenght
Walker2d-v2	500	1000
Swimmer-v2	160	1000
Reacher-v2	500	1000
Ant-v2	500	1000
HalfCheetah-v2	10,000	1000
Humanoid-v2	5,000	1000
HumanoidStandup-v2	15,000	1000
BipedalWalkerHardcore-v2	50	2000

3.8.7 Experimental Results on Acceleration Effects of Distributional RL

Same Architecture. For a fair comparison, we keep the same DAC network architecture and evaluate the gradient norms of DAC (C51) and a variant of AC, which is optimized based on the expectation of the represented value distribution within the DAC implementation framework. Figure 3.4 suggests DAC (C51) still enjoys smaller gradient norms than AC in this fair comparison setting.

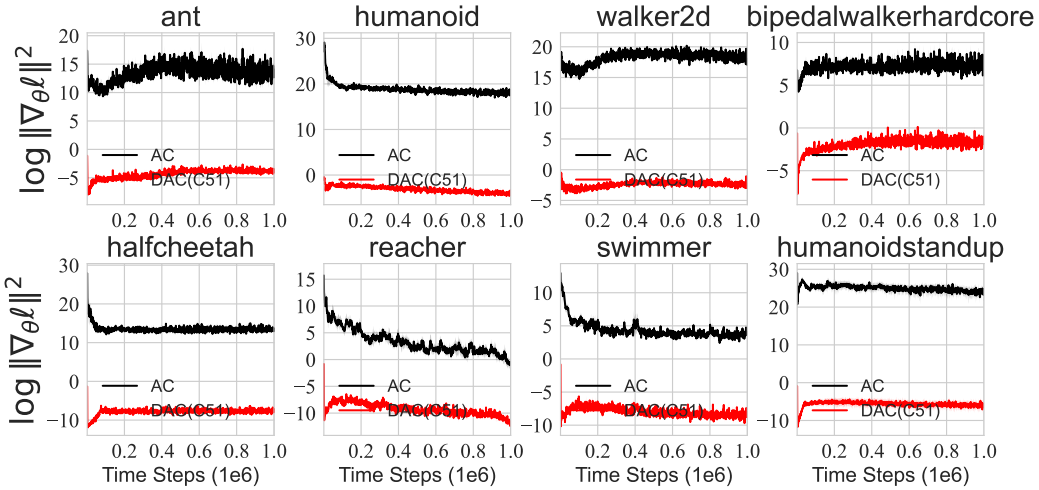


Figure 3.4: The critic gradient norms in the logarithmic scale during the training of AC and DAC (C51) over five seeds on three MuJoCo games. We keep the same DAC network architecture and evaluate based on the expectation of the represented value distribution.

Results under Return Density Decomposition We also provide gradient norms of both expectation and distribution based on the Return Density Function decomposition in Eq. 2.4. Similar results can still be observed in Figure 3.5.

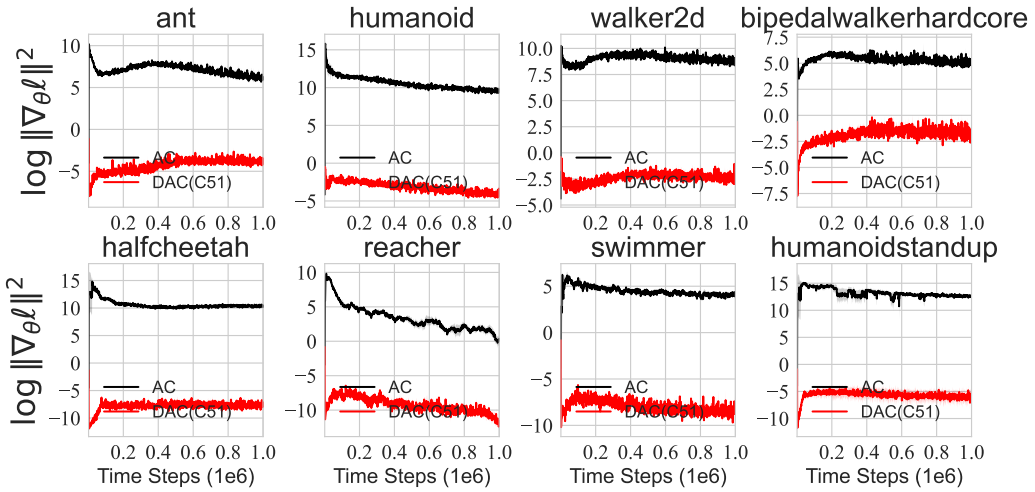


Figure 3.5: The critic gradient norms in the logarithmic scale during the training of AC and DAC (C51) over five seeds on three MuJoCo games. Results of AC is the expectation part calculated via the Return Density Decomposition.

Chapter 4

Exploring the Training Robustness of Distributional Reinforcement Learning against Noisy State Observations

4.1 Abstract

In real scenarios, state observations that an agent observes may contain measurement errors or adversarial noises, misleading the agent to take suboptimal actions or even collapse while training. In this paper, we study the training robustness of distributional Reinforcement Learning (RL), a class of state-of-the-art methods that estimate the whole distribution, as opposed to only the expectation, of the total return. Firstly, we validate the contraction of distributional Bellman operators in the State-Noisy Markov Decision Process (SN-MDP), a typical tabular case that incorporates both random and adversarial state observation noises. In the noisy setting with function approximation, we then analyze the vulnerability of least squared loss in expectation-based RL with either linear or nonlinear function approximation. By contrast, we theoretically characterize the bounded gradient norm of distributional RL loss based on the categorical parameterization equipped with the Kullback–Leibler (KL)

divergence. The resulting stable gradients while the optimization in distributional RL accounts for its better training robustness against state observation noises. Finally, extensive experiments on the suite of environments verified that distributional RL is less vulnerable against both random and adversarial noisy state observations compared with its expectation-based counterpart.

4.2 Introduction

Learning robust and high-performance policies for continuous state-action reinforcement learning (RL) domains is crucial to enable the successful adoption of deep RL in robotics, autonomy, and control problems. However, recent works have demonstrated that deep RL algorithms are vulnerable either to model uncertainties or external disturbances [48, 80, 50, 15, 116, 91, 94, 40]. Particularly, model uncertainties normally occur in a noisy reinforcement learning environment where the agent often encounters systematic or stochastic measurement errors on state observations, such as the inexact locations and velocity obtained from the equipped sensors of a robot. Moreover, external disturbances are normally adversarial in nature. For instance, the adversary can construct adversarial perturbations on state observations to degrade the performance of deep RL algorithms. These two factors lead to noisy state observations that influence the performance of algorithms, precluding the success of RL algorithms in real-world applications.

Existing works mainly focus on improving the robustness of algorithms in the *test environment* with noisy state observations. Smooth Regularized Reinforcement Learning [91] introduced a regularization to enforce smoothness in the learned policy, and thus improved its robustness against measurement errors in the test environment. Similarly, the State-Adversarial Markov Decision Process (SA-MDP) [116] was proposed and the resulting principled policy regularization enhances the adversarial robustness of various kinds of RL algorithms against adversarial noisy state observations. However, both of these works assumed that the agent can access *clean* state observations *during the training*, which is normally not feasible when the environment is inherently noisy, such as unavoidable measurement errors. Hence, the maintenance and

formal analysis of policies robust to noisy state observations *during the training* is a worthwhile area of research.

Recent distributional RL algorithms, e.g., C51 [9], Quantile-Regression DQN (QRDQN) [22], Implicit Quantile Networks (IQN) [21] and Moment-Matching DQN (MMD) [77], constantly set new records in Atari games, gaining huge attention in the research community. Existing literature mainly focuses on the performance of distributional RL algorithms, but *other benefits, including the robustness in the noisy environment, of distributional RL algorithms are less studied*. As distributional RL can leverage additional information about the value distribution that captures the uncertainty of the environment more accurately, it is natural to expect that distributional RL with this better representation capability can be less vulnerable to the noisy environment while training, which motivates our research. In this paper, we probe the robustness superiority of distributional RL against various kinds of state observation noises during the training process. Our contributions can be summarized as follows:

- **Tabular setting**. We firstly analyze a systematical noisy setting, i.e., State-Noisy Markov Decision Process (SN-MDP), incorporating both random and adversarial state observation noises. Theoretically, we derive the convergence of distributional Bellman operator in SN-MDP.

- **Function approximation setting**. We elaborate the additional convergence requirement of linear Temporal difference (TD) when exposed to noisy state observations. To clearly compare with distributional RL, we attribute its robustness advantage to the bounded gradients norms regarding state features based on the categorical parameterization of value distributions. This stable optimization behavior is in contrast to the potentially unbounded gradient norms of expectation-based RL.

- **Experiments**. We demonstrate that distributional RL algorithms potentially enjoy better robustness under various types of noisy state observations across a wide range of classical and continual control environments as well as Atari games. Our conclusion facilitates the deployment of distributional RL algorithms in more practical noisy settings.

4.2.1 Notations

We remain the notations in the last two chapters, except we slightly change the distributional Bellman operator. We first define the transition operator $P^\pi : \mathcal{Z} \to \mathcal{Z}$:

$$\mathcal{P}^\pi Z(s,a) :\overset{D}{=} Z\left(S', A'\right), S' \sim P(\cdot|s,a), A' \sim \pi\left(\cdot|S'\right), \tag{4.1}$$

where we use capital letters S' and A' to emphasize the random nature of both, and $:\overset{D}{=}$ indicates convergence in distribution. For simplicity, we denote $Z^\pi(s,a)$ by $Z(s,a)$. Thus, the distributional Bellman operator $\mathfrak{T}^\pi$ is defined as:

$$\mathfrak{T}^\pi Z(s,a) :\overset{D}{=} R(s,a,S') + \gamma \mathcal{P}^\pi Z(s,a). \tag{4.2}$$

4.3 Tabular Case: State-Noisy MDP

In this section, we extend State-Adversarial Markov Decision Process (SA-MDP) [116] to a more general State-Noisy Markov Decision Process (SN-MDP) by incorporating both random and adversarial state noises, and particularly provide a proof of the convergence and contraction of distributional Bellman operators in this setting.

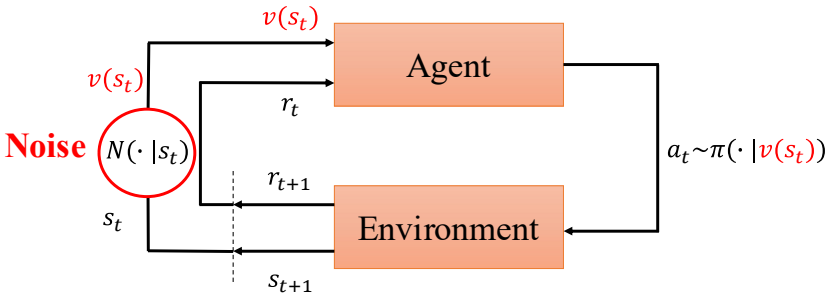


Figure 4.1: State-Noisy Markov Decision Process. $v(s_t)$ is perturbed by the noise mechanism N.

Definitions. SN-MDP is a 6-tuple $(\mathcal{S}, \mathcal{A}, R, P, \gamma, N)$, as exhibited in Figure 4.1, where the noise generating mechanism $N(\cdot|s)$ maps the state from

s to $v(s)$ using either random or adversarial noise with the Markovian and stationary probability $N(v(s)|s)$. It is worthwhile to note that the explicit definition of the noise mechanism N here is based on discrete state transitions, but the analysis can be naturally extended to the continuous case if we let the state space go to infinity. Moreover, let $\mathcal{B}(s)$ be the set that contains the allowed noise space for the noise generating mechanism N, i.e., $v(s) \in \mathcal{B}(s)$.

Following the setting in [116], we only manipulate state observations but do not change the underlying environment transition dynamics based on s or the agent's actions directly. As such, our SN-MDP is more suitable to model the random measurement error, e.g., sensor errors and equipment inaccuracies, and adversarial state observation perturbations in safety-critical scenarios. This setting is also aligned with many adversarial attacks on state observations [48, 61]. The following contractivity analysis regarding value function or distribution is directly based the state s rather than $v(s)$ as it is more natural and convenient to capture the uncertainty of MDP.

4.3.1 Analysis of SN-MDP for Expectation-based RL

We define the value function $\tilde{V}_{\pi \circ N}$ given π in SN-MDP. The Bellman Equations regarding the new value function $\tilde{V}_{\pi \circ N}$ are given by:

$$\tilde{V}_{\pi \circ N}(s) = \sum_{a} \sum_{v(s)} N(v(s)|s)\pi(a|v(s)) \sum_{s'} p(s'|s,a) \left[R(s,a,s') + \gamma \tilde{V}_{\pi \circ N}(s') \right], \tag{4.3}$$

where the random noise transits s into $v(s)$ with a certain probability and the adversarial noise is the special case of $N(v(s)|s)$ where $N(v^*(s)|s) = 1$ if $v^*(s)$ is the optimal adversarial noisy state given s, and $N(v(s)|s) = 0$ otherwise. We denote Bellman operators under random noise mechanism $N^r(\cdot|s)$ and adversarial noise mechanism $N^*(\cdot|s)$ as $\mathcal{T}_r^\pi$ and $\mathcal{T}_a^\pi$, respectively. This implies that $\mathcal{T}_r^\pi \tilde{V}_{\pi \circ N} = \tilde{V}_{\pi \circ N^r}$ and $\mathcal{T}_a^\pi \tilde{V}_{\pi \circ N} = \tilde{V}_{\pi \circ N^*}$. We extend Theorem 1 in [116] to both random and adversarial noise scenarios, and immediately obtain that both $\mathcal{T}_r^\pi$ and $\mathcal{T}_a^\pi$ are contraction operators in SN-MDP. We provide a rigorous description in Theorem 7 with the proof in Appendix 4.7.1.

The insightful and pivotal conclusion from Theorem 7 is $\mathcal{T}_a^\pi \tilde{V}_{\pi \circ N} = \min_N \tilde{V}_{\pi \circ N}$.

This implies that the adversary attempts to minimize the value function, forcing the agent to select the worse-case action among the allowed transition probability space $N(\cdot|s)$ for each state s. The crux of the proof is that Bellman updates in SN-MDP result in the convergence to the value function for another "merged" policy π' where $\pi'(a|s) = \sum_{v(s)} N(v(s)|s)\pi(a|v(s))$. Nevertheless, the converged value function corresponding to the merged policy might be far away from that for the original policy π, which is more likely to worsen the performance of RL algorithms.

4.3.2 Analysis of SN-MDP in distributional RL

In the SN-MDP setting for distributional RL, the new distributional Bellman equations use new transition operators in place of $\mathcal{P}^\pi$ in Eq. 4.1. The new transition operators $\mathcal{P}_r^\pi$ and $\mathcal{P}_a^\pi$, for the random and adversarial settings, are defined as:

$$\begin{aligned} \mathcal{P}_r^\pi Z_N(s,a) &:\overset{D}{=} Z_{N^r}(S',A'), A' \sim \pi(\cdot|V(S')), \text{and} \\ \mathcal{P}_a^\pi Z_N(s,a) &:\overset{D}{=} Z_{N^*}(S',A'), A' \sim \pi(\cdot|V^*(S')), \end{aligned} \tag{4.4}$$

where $V(S') \sim N^r(\cdot|S')$ is the state random variable after the transition, and $V^*(S')$ is attained from $N^*(\cdot|S')$ under the optimal adversary. Besides, $S' \sim P(\cdot|s,a)$. Therefore, the corresponding new distributional Bellman operators $\mathfrak{T}_r^\pi$ and $\mathfrak{T}_a^\pi$ are formulated as:

$$\begin{aligned} \mathfrak{T}_r^\pi Z_N(s,a) &:\overset{D}{=} R(s,a,S') + \gamma \mathcal{P}_r^\pi Z_N(s,a), \text{and} \\ \mathfrak{T}_a^\pi Z_N(s,a) &:\overset{D}{=} R(s,a,S') + \gamma \mathcal{P}_a^\pi Z_N(s,a). \end{aligned} \tag{4.5}$$

In this sense, four sources of randomness define the new compound distribution in the SN-MDP: (1) randomness of reward, (2) randomness in the new environment transition dynamics $\mathcal{P}_r^\pi$ or $\mathcal{P}_a^\pi$ that additionally includes (3) the stochasticity of the noisy transition N, and (4) the random next-state value distribution $Z(S',A')$. As our first theoretical contribution, we now show that the new derived distribution Bellman Operators defined in Eq. 4.5 in SN-MDP setting are convergent and contractive for policy evaluation in Theorem 5.

Theorem 5. *(Convergence and Contraction of Distributional Bellman operators in the SN-MDP) Given a policy π, we define the distributional Bellman operators $\mathfrak{T}_r^\pi$ and $\mathfrak{T}_a^\pi$ in Eq. 4.5, and consider the Wasserstein metric d_p, the following results hold.*
(1) $\mathfrak{T}_r^\pi$ is a contraction under the maximal form of d_p.
(2) $\mathfrak{T}_a^\pi$ is also a contraction under the maximal form of d_p, following the greedy adversarial rule, i.e., $N^*(\cdot|s') = \arg\min_{N(\cdot|s')} \mathbb{E}\left[Z(s', a')\right]$ where $a' \sim \pi(\cdot|V(s'))$ and $V(s') \sim N(\cdot|s')$.

We provide the proof in Appendix 4.7.2. Similar to the convergence conclusions in classical RL, Theorem 5 justified that distributional RL is also capable of converging in this SN-MDP setting. The contraction and convergence of distributional Bellman operators in the SN-MDP is one of our main contributions. This result allows us to deploy distributional RL algorithms comfortably in the tabular setting even with noisy state observations.

4.4 Function Approximation Case

In the tabular case, both expectation-based and distributional RL have convergence properties. However, in the function approximation case, we firstly show linear TD requires more conditions for the convergence, and point out the vulnerability of expectation-based RL against noisy states even under the bounded rewards assumption. In contrast, we analyze that distributional RL with the categorical representation for the value distributions, is more robust against noisy state observations due to its bounded gradient norms.

4.4.1 Convergence of Linear TD under Noisy States

In classical RL with function approximation, the value estimator $\hat{v} : \mathcal{S} \times \mathbb{R}^d \to \mathbb{R}$ parameterized by $\mathbf{w}$ is expressed as $\hat{v}(s, \mathbf{w})$. The objective function is *Mean Squared Value Error* [102] denoted as $\overline{\text{VE}}$:

$$\overline{\text{VE}}(\mathbf{w}) \doteq \sum_{s \in \mathcal{S}} \mu(s) \left[v_\pi(s) - \hat{v}(s, \mathbf{w})\right]^2, \tag{4.6}$$

where μ is the state distribution. In linear TD, the value estimate is formed simply as the inner product between state features $\mathbf{x}(s)$ and weights $\mathbf{w} \in \mathbb{R}^d$, given by $\hat{v}(s, \mathbf{w}) \stackrel{\text{def}}{=} \mathbf{w}^\top \mathbf{x}(s)$. At each step, the state feature can be rewritten as $\mathbf{x}_t \stackrel{\text{def}}{=} \mathbf{x}(S_t) \in \mathbb{R}^d$. Thus, the TD update at step t is:

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \alpha_t (R_{t+1} + \gamma \mathbf{w}_t^\top \mathbf{x}_{t+1} - \mathbf{w}_t^\top \mathbf{x}_t) \mathbf{x}_t \tag{4.7}$$

where α_t is the step size at time t. Once the system has reached the steady state for any $\mathbf{w}_t$, then the expected next weight vector can be written as $\mathbb{E}[\mathbf{w}_{t+1} | \mathbf{w}_t] = \mathbf{w}_t + \alpha_t (\mathbf{b} - \mathbf{A}\mathbf{w}_t)$, where $\mathbf{b} = \mathbb{E}(R_{t+1}\mathbf{x}_t) \in \mathbb{R}^d$ and $\mathbf{A} \doteq \mathbb{E}\left[\mathbf{x}_t d_t^\top\right] \in \mathbb{R}^{d \times d}$. The TD fixed point $\mathbf{w}_{\text{TD}}$ to the system satisfies $\mathbf{A}\mathbf{w}_{\text{TD}} = \mathbf{b}$. From [102], we know that the matrix $\mathbf{A}$ determines the convergence in the linear TD setting. In particular, $\mathbf{w}_t$ converges with probability one to the TD fixed point if $\mathbf{A}$ is positive definite. However, if we add state noises η on $\mathbf{x}_t$ in Eq. 4.7, the convergence condition will be different. As shown in Theorem 6 (a more formal version with the proof is given in Appendix 4.7.4), linear TD under noisy state observations requires additional positive definiteness condition.

Theorem 6. *(Covergence Conditions for Linear TD under Noisy State Observations) Define* $\mathbf{P}$ *as the* $|\mathcal{S}| \times |\mathcal{S}|$ *matrix forming from the state transition probability* $p(s'|s)$*,* $\mathbf{D}$ *as the* $|\mathcal{S}| \times |\mathcal{S}|$ *diagonal matrix with* $\mu(s)$ *on its diagonal, and* $\mathbf{X}$ *as the* $|\mathcal{S}| \times d$ *matrix with* $\mathbf{x}(s)$ *as its rows, and* $\mathbf{E}$ *is the* $|\mathcal{S}| \times d$ *perturbation matrix with each perturbation vector* $\mathbf{e}(s)$ *as its rows.* $\mathbf{w}_t$ *converges to TD fixed point* ***when both*** $\mathbf{A}$ ***and*** $(\mathbf{X} + \mathbf{E})^\top \mathbf{D} \mathbf{P} \mathbf{E}$ ***are positive definite.***

However, directly analyzing the convergence conditions of distributional linear TD and then comparing with them in Theorem 6 for classical linear TD is tricky in theory. As such, we additionally provide a sensitivity comparison of both expectation-based and distributional RL through the lens of their gradients regarding state features as follows.

4.4.2 Vulnerability of Expectation-based RL

We reveal that the vulnerability of expectation-based RL can be attributed to its unbounded gradient characteristics in both linear and nonlinear approxi-

mation settings.

Linear Approximation Setting. To solve the *weighted* least squared minimization in Eq. 4.6, we leverage Stochastic Gradient Descent (SGD) on the empirical version of $\overline{\text{VE}}$, which we denote as $g_{\overline{\text{VE}}}$. *We focus on the gradient norm of $g_{\overline{\text{VE}}}$ regarding the state features $\mathbf{x}(s)$ (or $\mathbf{x}_t$) as the gradient of loss w.r.t state observations is highly correlated with the sensitivity or robustness of algorithms against the noisy state observations.* For a fair comparison with distributional RL in next section, we additionally bound the norm of $\mathbf{w}$, i.e., $\|\mathbf{w}\| \leq l$, which can also be easily satisfied by imposing ℓ_1 or ℓ_2 regularization. Therefore, we derive the upper bound of gradient norm of $g_{\overline{\text{VE}}}$ as

$$\|\frac{\partial g_{\overline{\text{VE}}(\mathbf{w})}}{\partial \mathbf{x}_t}\| = |U_t - \mathbf{w}_t^\top \mathbf{x}_t| \|\mathbf{w}_t\| \leq |U_t - \mathbf{w}_t^\top \mathbf{x}_t| l, \tag{4.8}$$

where the target U_t can be either an unbiased estimate via Monte Carlo method with $U_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$, or a biased estimate via TD learning with $U_t = r_{t+1} + \gamma \mathbf{w}_t^\top \mathbf{x}_{t+1}$. However, this upper bound $|U_t - \mathbf{w}_t^\top \mathbf{x}_t| l$ heavily depends on the perturbation size or noise strength. Even under the bounded rewards assumption, i.e., $r \in [R_{\min}, R_{\max}]$, we can bound U_t as $U_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \in [\frac{R_{\min}}{1-\gamma}, \frac{R_{\max}}{1-\gamma}]$. However, this upper bound can be arbitrarily large if we have no restriction on the noise size, leading to a potentially huge vulnerability against state observation noises.

Nonlinear Approximation Setting. The potentially unbounded gradient norm issue of expectation-based RL in the linear case still remains in the nonlinear approximation setting. We express the value estimate $\hat{v}$ as $\hat{v}(s; \mathbf{w}, \theta) = \phi_{\mathbf{w}}(\mathbf{x}(s))^\top \theta$, where $\phi_{\mathbf{w}}(\mathbf{x}(s))$ is the representation vector of the state feature $\mathbf{x}(s)$ in the penultimate layer of neural network-based value function approximator. Correspondingly, θ would be the parameters in the last layer of this value neural network. We simplify $\phi_{\mathbf{w}}(\mathbf{x}(s))_t$ as $\phi_{\mathbf{w},t}$ in the step t update. As such, akin to the linear case, we derive the upper bound of gradient

norm of $g_{\overline{\text{VE}}}$ as

$$\|\frac{\partial g_{\overline{\text{VE}}(\mathbf{w},\theta)}}{\partial \mathbf{x}_t}\| = |U_t - \phi_{\mathbf{w},t}^\top \theta_t| \|\nabla_{\mathbf{x}_t} \phi_{\mathbf{w},t}^\top \theta_t\| \leq |U_t - \phi_{\mathbf{w},t}^\top \theta_t| lL, \tag{4.9}$$

where we assume the function $\phi_{\mathbf{w}}(\cdot)$ is L-Lipschitz continuous regarding its input state feature $\mathbf{x}(s)$, and $\|\theta\| \leq l$ as well for a fair comparison with distributional RL. It turns out that $|U_t - \phi_{\mathbf{w},t}^\top \theta_t|$ still depends on the perturbation size, and can be still arbitrarily large if there is no restriction on the noise size. In contrast, we further show that gradient norms in distributional RL can be upper bounded **regardless of the perturbation size or noise strength**.

4.4.3 Robustness Advantage of distributional RL

We analyze the distributional loss in distributional RL can potentially lead to bounded gradient norms regarding state features regardless of the perturbation size, yielding its training robustness against state noises. In distributional RL our goal is to minimize a distribution loss $\mathcal{L}\left(Z_{\mathbf{w}}, \mathfrak{T} Z_{\mathbf{w}}\right)$ between the current value distribution of $Z_{\mathbf{w}}$ and its target value distribution of $\mathfrak{T} Z_{\mathbf{w}}$.

Our robustness analysis is based on the categorical parameterization [51] on the value distribution with the KL divergence, a typical choice also used in the first distributional RL branch, i.e., C51 [9]. Specifically, we uniformly partition the support of $Z_{\mathbf{w}}(s)$ into k bins, and let the histogram function $f : \mathcal{X} \to [0,1]^k$ provide k-dimensional vector $f(\mathbf{x}(s))$ of the coefficients indicating the probability the target is in that bin given $\mathbf{x}(s)$. We use *softmax* to output the k probabilities of $f(\mathbf{x}(s))$. Therefore, the categorical distributional RL loss $\mathcal{L}(Z_{\mathbf{w}}(s), \mathfrak{T} Z_{\mathbf{w}}(s))$, denoted as $\mathcal{L}_{\mathbf{w}}$, equipped with KL divergence between $Z_{\mathbf{w}}$ and $\mathfrak{T} Z_{\mathbf{w}}$ can be simplified as

$$\mathcal{L}(Z_{\mathbf{w}}(s), \mathfrak{T} Z_{\mathbf{w}}(s)) \propto -\sum_{i=1}^{k} p_i \log f_i^{\mathbf{w}}(\mathbf{x}(s)), \tag{4.10}$$

where we use $\mathbf{w}$ to parameterize the function f in the distributional loss $\mathcal{L}_{\mathbf{w}}$, and the target probability p_i is the cumulative probability increment of target distribution $\mathfrak{T} Z_{\mathbf{w}}$ within the i-th bin. Detailed derivation about the simplifi-

cation of categorical distributional loss is in Appendix 3.8.3.

Linear Approximation Setting. We leverage $\mathbf{x}(s)^\top \mathbf{w}_i$ to express the i-th output of f, i.e., $f_i(\mathbf{x}(s)) = \exp\left(\mathbf{x}(s)^\top \mathbf{w}_i\right) / \sum_{j=1}^{k} \exp\left(\mathbf{x}(s)^\top \mathbf{w}_j\right)$, where all parameters are $\mathbf{w} = \{\mathbf{w}_1, ..., \mathbf{w}_k\}$. Based on this categorical distributional RL loss, we obtain Proposition 9 (proof in Appendix 4.7.3), revealing that value-based categorical distributional RL loss can result in bounded gradient norms regarding state features $\mathbf{x}(s)$.

Proposition 9. *(Gradient Property of distributional RL in Linear Approximation) Consider the categorical distributional RL loss $\mathcal{L}_{\mathbf{w}}$ in Eq. 4.10 with the linear approximation. Assume $\|\mathbf{w}_i\| \leq l$ for $\forall i = 1, .., k$, then $\left\| \frac{\partial \mathcal{L}_{\mathbf{w}}}{\partial \mathbf{x}(s)} \right\| \leq kl$.*

In contrast with the unbounded gradient norm in Eq. 4.8 of classical RL, we have a restricted upper bound in distributional RL loss with a linear approximator, i.e., kl, which is independent of the perturbation size or noise strength.

Nonlinear Approximation Setting. Similar to the nonlinear form in classical expectation-based RL as analyzed in Section 4.4.2, we express the i-th output probabilities of $f(\mathbf{x}(s))$ as

$$f_i^{\mathbf{w},\theta}(\mathbf{x}(s)) = \exp\left(\phi_{\mathbf{w}}(\mathbf{x}(s))^\top \theta_i\right) / \sum_{j=1}^{k} \exp\left(\phi_{\mathbf{w}}(\mathbf{x}(s))^\top \theta_j\right),$$

where the last layer parameter $\theta = \{\theta_1, ..., \theta_k\}$ and $\phi_{\mathbf{w}}(\mathbf{x}(s))$ is still the representation vector of $\mathbf{x}(s)$. In Proposition 10, we can still attain a bounded gradient norm of distributional RL loss in the nonlinear case.

Proposition 10. *(Gradient Property of distributional RL in Nonlinear Approximation) Consider the categorical distributional RL loss $\mathcal{L}_{\mathbf{w},\theta}$ in Eq. 4.10 with the nonlinear approximation. Assume $\|\theta_i\| \leq l$ for $\forall i = 1, .., k$ and $\phi_{\mathbf{w}}(\cdot)$ is L-Lipschitz continuous, then $\left\| \frac{\partial \mathcal{L}_{\mathbf{w},\theta}}{\partial \mathbf{x}(s)} \right\| \leq klL$.*

Please refer to Appendix 4.7.3 for the proof. For a fair comparison with nonlinear approximation in classical RL, we still assume the function $\phi_{\mathbf{w}}(\cdot)$ to

be L-Lipschitz continuous and $\|\theta_i\| \leq l$. *Interestingly, the bounded gradient norm of the distributional RL loss is independent of the noise size, which is in stark contrast to the potentially unrestricted gradients in classical RL in Eq. 4.9 that heavily depends on the noise size.* Based on Theorems 9 and 10, we conclude that the bounded gradient behaviors of distributional RL could reduce its sensitivity to state noises, and thus mitigate the interference of the state observation noises compared with expectation-based RL, potentially leading to better training robustness.

Extension of TD Convergence and Sensitivity Analysis. As supplementary, we also conduct the analysis on different TD convergence conditions under the unbalanced perturbations on either the current or next state observations. Please refer to Theorem 8 with the detailed explanation in Appendix 4.7.4. In addition, we also conduct a sensitivity analysis from the perspective of the *influence function* to characterize the impact of state noises on an estimator. We provide the details in Theorem 9 of Appendix 4.7.5.

4.5 Experiments

We make a comparison between expectation-based and distributional RL algorithms against various noisy state observations across **classical and continuous control environments as well as Atari games**, including Cartpole and MountainCar (classical control), Ant, Humanoidstandup and Halfcheetah (continuous control), Breakout and Qbert (Atari games). For the continuous control environment, we use Soft Actor Critic [43] and Distributional Soft Actor Critic [67] with C51 as the critic loss and thus we denote them as SAC and DAC (C51), respectively. For the classical control and Atari games, we utilize DQN [74] as the baseline, and C51 [9], QRDQN [22] as its distributional counterparts. The training robustness of C51 could be consistent with our theoretical analysis, while QRDQN, the more commonly-used one, is also applied to demonstrate that our robustness analysis can also be empirically applicable to broader distributional RL algorithms. The previous analysis is for policy evaluation, but there are natural—though in some cases heuristic—extensions

to the control setting.

Implementation and Experimental Setup. For the continuous control environment, we modified our algorithm based on released implementation of [67]. For classical control and Atari games, we followed the procedure in [37, 119]. All the experimental settings, including parameters, are identical to the distributional RL baselines implemented by [118, 22]. We perform 200 runs on both Cart Pole and Mountain Car and 3 runs on Breakout and Qbert. Reported results are averaged with shading indicating the standard error. The learning curve is smoothed over a window of size 10 before averaging across runs. Please refer to Appendix 4.7.6 for more details about the experimental setup.

Evaluation of Training Robustness. Due to final performance difference between expectation-based and distributional RL, for a fair comparison we calculate *the ratio between final average returns under random or adversarial state noises with different noise strengths and the original level without any state noises*. This ratio can be used to measure the robustness maintenance after the agent gets exposed to noisy state observations.

Random and Adversarial State Noises. We use Gaussian noise with different standard deviations to simulate random state noises, while for the adversarial state noise, we apply the most typical adversarial state perturbations proposed in [48, 80]. For the choice of perturbation size, we followed [116], where the set of noises $B(s)$ is defined as an ℓ_∞ norm ball around s with a radius ϵ, given by $\ell_\infty B(s) := \{\hat{s} : \|s - \hat{s}\|_\infty \leq \epsilon\}$. We apply Projected Gradient Descent (PGD) version in [80], with 3 fixed iterations while adjusting ϵ to control the perturbation strength. Due to the page limit, we defer similar results under more advanced MAD attack [116] in Appendix 4.7.7.

4.5.1 Results on Continuous Control Environments

We compare SAC with DAC (C51) on Ant and Humanoidstandup. Due to the space limit, we mainly present the algorithm performance in the **adver-**

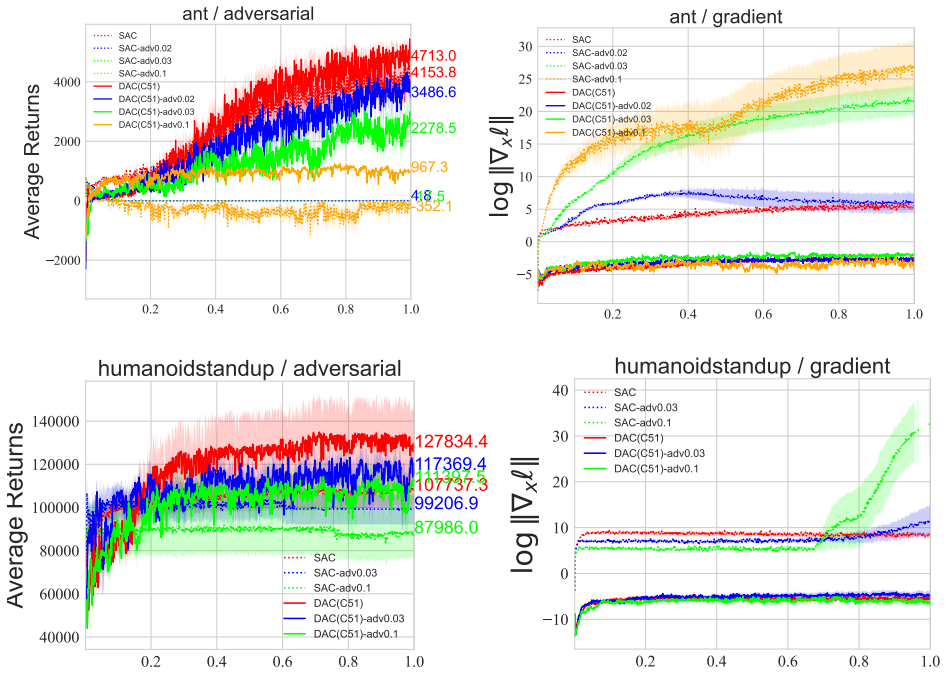


Figure 4.2: Average returns of SAC and DAC (C51) against **adversarial** state observation noises in the training on Ant and Humanoidstandup under 5 runs. Gradient norms in the logarithm scale of AC and DAC (C51) in the adversarial setting. **advX** in the legend indicates random state observations with the perturbation size ϵ **X**.

sarial setting. Figure 4.2 suggests that distributional RL algorithms, i.e., DAC (C51), are less sensitive to their expectation-based counterparts, i.e., SAC, according to learning curves of average returns on Ant and Humanoidstandup. More importantly, Figure 4.2 demonstrates that DAC (C51) enjoys smaller gradient norms compared with SAC, and SAC with a larger perturbation size is prone to unstable training with much larger gradient magnitudes. In particular, On Humanoidstandup, SAC converges undesirably with adv0.01 (green line), but its gradient norm diverges (even infinity in the very last phase). By contrast, DSAC (C51) has a lower level gradient norms, which is less likely to suffer from divergence. This result corroborates with theoretical analysis in Section 4.4.3 that exploding gradients are prone to divergence when exposed to state noises.

A quantitative result is also shown in Table 4.1, where distributional RL

Robustness(%)	**Adversarial**	ϵ**=0.02**	ϵ**=0.03**	ϵ**=0.1**
Ant	SAC	≈ 0	≈ 0	≈ 0
	DAC (C51)	**74.0**	**48.3**	**20.5**
Robustness(%)	**Adversarial**	ϵ**=0.03**	ϵ**=0.1**	
Humanoidstandup	SAC	**92.1**	81.7	
	DAC (C51)	91.8	**87.1**	

Table 4.1: Robustness ratio of algorithms under **adversarial** state observations with different ϵ on Ant and Humanoidstandup.

algorithms tend to maintain a higher robustness ratio as opposed to their expectation-based RL versions. We also note that the training robustness of distributional RL algorithms may not be significant if the perturbation size is slightly small, e.g., on Humanoidstandup. However, if we carefully vary perturbation sizes in a proper range, we can easily observe the robustness advantage of distributional RL against adversarial noises, e.g., on Ant. We also investigate the training robustness of more distributional RL algorithms over more games. Thus, we evaluate the sensitivity of D4PG [8] against adversarial noises on Halfcheetah, which can be viewed as the distributional version of DDPG. As suggested in Figure 4.6 in Appendix 4.7.8, the distributional RL algorithm D4PG is much less vulnerable than it expectation-based RL counterpart DDPG against adversarial noises.

4.5.2 Results on Classical Control and Atari Games

Results under Random State Noises. We investigate the training robustness of DQN, C51 and QRDQN on classical control environments and typical Atari games, against the random noisy state observations. Gaussian state noises are continuously injected in the while training process of RL algorithms, while the agent encounters noisy current state observations while conducting the TD learning. Due to the space limit, here we mainly present learning curves of algorithms on CartPole and Breakout. As shown in Figure 4.3, both C51 and QRDQN achieve similar performance to DQN after the training *without any random state noises*. However, when we start to inject random state noises with different noise sizes during the training process, their learning curves show different sensitivity and robustness. Both C51 and

Robustness(%)	**Random**	**std=0.05**	**std=0.1**
CartPole	DQN	44.2	28.6
	QRDQN	54.5	43.4
	C51	**67.0**	**47.3**
Robustness(%)	**Random**	**std=0.01**	**std=0.05**
Breakout	DQN	59.1	≈ 0
	QRDQN	81.1	73.1
	C51	**146.5**	**88.7**

Table 4.2: Robustness ratio of three algorithms under **random** state observations with different standard deviations (std) on CartPole and Breakout.

QRDQN are more robust against the random state noises than DQN, with the less interference for the training under the same random noises. Remarkably, in Breakout the performance of both C51 and QRDQN (solid lines) only slightly decreases, while DQN (dashed lines) degrades dramatically and even diverges when the standard deviation is 0.05. This significant difference provides a strong empirical evidence to verify the robustness advantage of distributional RL algorithms.

A detailed comparison is summarized in Table 4.2. It turns out that the

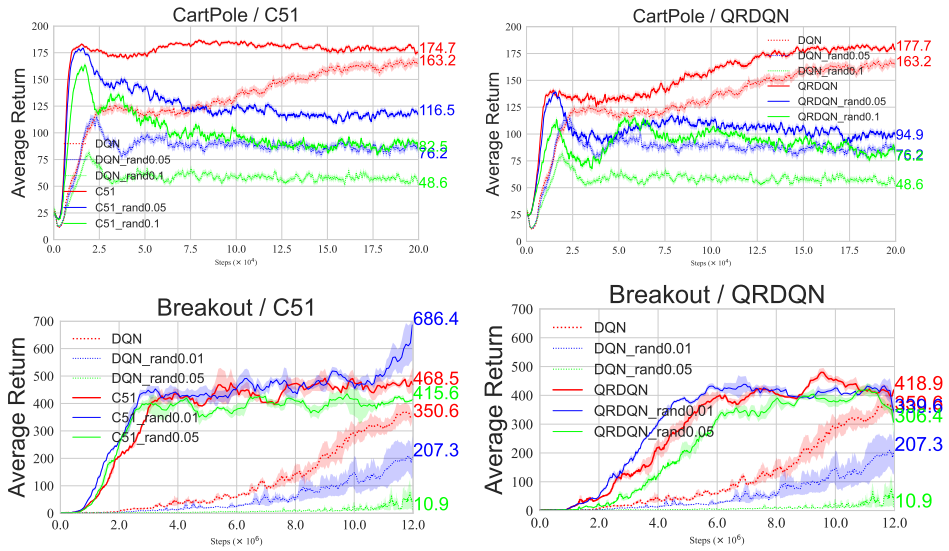


Figure 4.3: Average returns of DQN, C51 and QRDQN against **random** state observation noises on CartPole and Breakout. **randX** in the legend indicates random state observations with the standard deviation **X**.

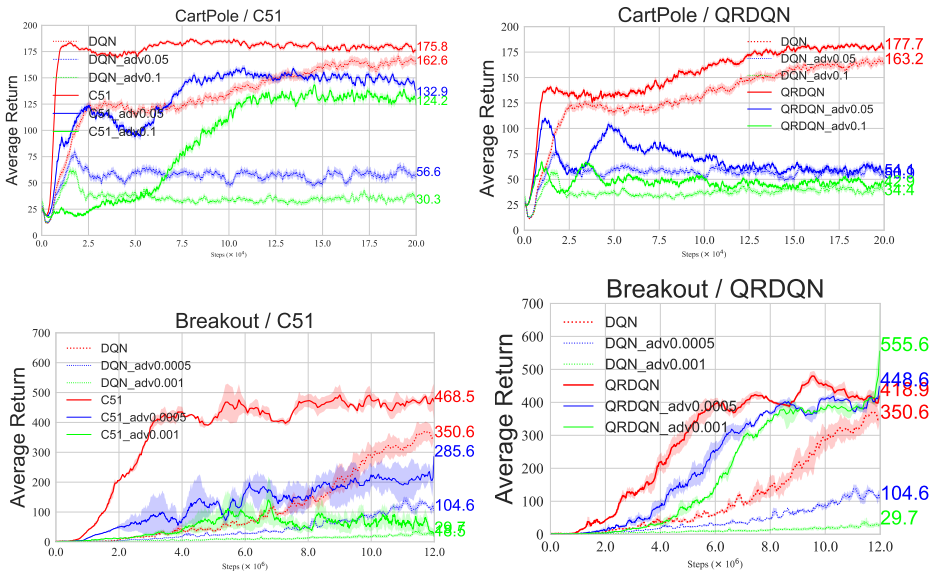


Figure 4.4: Average returns of DQN, C51 and QRDQN against **adversarial** state observation noises across four games. **advX** in the legend indicates random state observations with the perturbation size ϵ **X**.

training robustness of both QRDQN and C51 surpass DQN significantly. Note that the robustness ratio for C51 under std=0.01 noises is 146.5%, which is above 100%. This can be explained as a proper randomness added in the training might be beneficial to exploration, yielding better generalization of algorithms.

Results under Adversarial State Noises. Next, we probe the training robustness of DQN, QRDQN and C51 in the setting where the agent encounters the *adversarial* state observations in the current state in the function approximation case. Figure 4.4 presents the learning curves of algorithms on CartPole and Breakout against noisy states under different adversarial perturbation sizes ϵ.

It turns out that results under the adversarial state observations are similar to those in the random noises case. Specifically, all algorithms tend to degrade when getting exposed to adversarial state observations, and even are more likely to diverge. However, a key observation is that *distributional RL algorithms, especially QRDQN, are capable of obtaining desirable performance*

Robustness(%)	**Adversarial**	**ϵ=0.05**	**ϵ=0.1**
CartPole	DQN	34.8	18.6
	QRDQN	26.0	24.8
	C51	**75.6**	**70.6**
Robustness(%)	**Adversarial**	**ϵ=0.0005**	**ϵ=0.001**
Breakout	DQN	29.8	≈ 0
	QRDQN	**107.1**	**132.6**
	C51	61.0	6.3

Table 4.3: Robustness ratio of three algorithms under **adversarial** state observations with different perturbation sizes ϵ on CartPole and Breakout.

even when DQN diverges. For instance, in Breakout DQN (dotted green line) in Figure 4.4 under the adversarial perturbation with $\epsilon = 0.001$ leads to divergence, while QRDQN (solid green lines) still maintains a desirable performance. The quantitative robustness ratio comparison is also provided in Table 4.3. It suggests that the adversarial robustness of C51 is superior to DQN and QRDQN in CartPole, while QRDQN is remarkably less sensitive to adversarial noises than both DQN and C51 in Breakout.

Results on MountainCar and Qbert. Due to the space limit, we mainly summarize the robustness ratio of algorithms on MountainCar and Qbert in Table 4.4. It turns out that the training robustness of QRDQN is significantly advantageous over DQN on both MountainCar and Qbert environments across two types of state noises, which also corroborates the robustness advantage of distributional RL algorithms over their expectation-based RL counterpart.

Robustness(%)	**Algorithms**	**std=0.0125**	**ϵ=0.1**
MountainCar	DQN	32.4	32.5
	QRDQN	**79.0**	**44.7**
Robustness(%)	**Algorithms**	**std=0.05**	**ϵ=0.005**
Qbert	DQN	10.8	6.3
	QRDQN	**34.5**	**32.9**

Table 4.4: Robustness ratio of DQN and QRDQN under random and adversarial state noises on MountainCar and Qbert.

4.6 Discussion and Conclusion

The robustness advantage analysis is based on the categorical distributional RL with categorical parameterization and the choice of KL divergence between current and target value distributions. However, it would be more convincing if we can still have such an analytical conclusion under Wasserstein distance. Moreover, we attribute the robustness advantage of distributional RL algorithms into the unbounded gradient norms regarding state features, but other factors, e.g., representation ability, may also contribute to the training robustness. We leave the exploration towards this direction as future works.

In this paper, we explored the training robustness of distributional RL against both random and adversarial noisy state observations. After the convergence proof of distributional RL in the SN-MDP, we further uncover the stable gradient behavior of distributional RL loss as opposed to classical RL, accounting for its less vulnerability. Experimental observations coincides with our theoretical results.

4.7 Appendix

4.7.1 Theorem 7 with proof

Theorem 7. *(Convergence and Contraction of Bellman operators in the SN-MDP) Given a policy π, define the Bellman operator $\mathcal{T} : \mathbb{R}^{|S|} \to \mathbb{R}^{|S|}$ under random and adversarial states noises by $\mathcal{T}_r^\pi$ and $\mathcal{T}_a^\pi$, respectively. Denote a "merged" policy π' where $\pi'(a|s) = \sum_{v(s)} N(v(s)|s)\pi(a|v(s))$ and $\mathbf{S}(\pi)$ is a policy set given π. Then we have:*

(1) $\mathcal{T}_r^\pi$ is a contraction operator and can converge to $V_{\pi'}$, i.e., $\mathcal{T}_r^\pi \tilde{V}_{\pi\circ N} = \tilde{V}_{\pi\circ N} = V_{\pi'}$, where multiple policies $\pi_r \in \mathbf{S}(\pi)$ might exist, which satisfies

$$\sum_{v(s)} N(v(s)|s)\pi_r(a|v(s)) = \pi'(a|s). \tag{4.11}$$

(2) $\mathcal{T}_a^\pi$ is a contraction with the convergence satisfying $\mathcal{T}_a^\pi \tilde{V}_{\pi\circ N^*} = \min_N \tilde{V}_{\pi\circ N} = V_{\pi\circ N^*}$, where N^* is the optimal adversarial noise strategy. If the optimal policy

π_a exists, it satisfies $\pi_a(a|v^*(s)) = \pi(a|s)$ for each s and a, where $v^*(s)$ is the adversarial noisy state manipulated by $N^*(\cdot|s)$.

Proof. Our proof is partly based on Theorem 1 and 2 in [116], but adds more analysis on the converged policy especially under the random noisy states setting. The most important insight in the following proof is that the noise transition can be merged into the agent's policy, resulting in a new "merged" policy π'.

Proof of (1) Firstly, as the Bellman Equation under the random noisy states is right the general form in Eq. 4.3, it automatically satisfies that $\mathcal{T}_r^\pi \tilde{V}_{\pi\circ N} = \tilde{V}_{\pi\circ N}$ when it converges. As for the proof of contraction, based on our insight about the new "merged" policy π' where $\pi'(a|s) = \sum_{v(s)} N(v(s)|s)\pi(a|v(s))$, we can rewrite our Bellman Operator as:

$$\begin{aligned}\mathcal{T}_r^\pi \tilde{V}_{\pi\circ N}(s) &= \sum_a \pi'(a|s) \sum_{s'} p(s'|s,a) \Big[R(s,a,s') + \gamma \tilde{V}_{\pi\circ N}(s')\Big] \\ &= \boldsymbol{R}(s) + \gamma \sum_{s'} \boldsymbol{P}'_{s,s'} \tilde{V}_{\pi\circ N}(s')\end{aligned} \tag{4.12}$$

where $\boldsymbol{R}(s) = \sum_a \pi'(a|s) \sum_{s'} p(s'|s,a) R(s,a,s')$, and $\boldsymbol{P}'_{s,s'} = \sum_a \pi'(a|s) p(s'|s,a)$ determined by the "merged" policy π'. Then for two different value function $\tilde{V}^1_{\pi\circ N}$ and $\tilde{V}^2_{\pi\circ N}$ we have:

$$\begin{aligned}\|\mathcal{T}_r^\pi \tilde{V}^1_{\pi\circ N} - \mathcal{T}_r^\pi \tilde{V}^2_{\pi\circ N}\|_\infty &= \max_s |\gamma \sum_{s'} \boldsymbol{P}'_{s,s'} \tilde{V}^1_{\pi\circ N}(s') - \gamma \sum_{s'} \boldsymbol{P}'_{s,s'} \tilde{V}^2_{\pi\circ N}(s')| \\ &\le \gamma \max_s \sum_{s'} \boldsymbol{P}'_{s,s'} |\tilde{V}^1_{\pi\circ N}(s') - \tilde{V}^2_{\pi\circ N}(s')| \\ &\le \gamma \max_s \sum_{s'} \boldsymbol{P}'_{s,s'} \max_{s'} |\tilde{V}^1_{\pi\circ N}(s') - \tilde{V}^2_{\pi\circ N}(s')| \\ &= \gamma \max_s \sum_{s'} \boldsymbol{P}'_{s,s'} \|\tilde{V}^1_{\pi\circ N} - \tilde{V}^2_{\pi\circ N}\|_\infty \\ &= \gamma \|\tilde{V}^1_{\pi\circ N} - \tilde{V}^2_{\pi\circ N}\|_\infty\end{aligned} \tag{4.13}$$

Then according to the Banach fixed-point theorem, since $\gamma \in (0,1)$, $\tilde{V}_{\pi\circ N}$ converges to a unique fixed-point $V_{\pi'}$. However, even though the obtained policy π' satisfies that $\pi'(a|s) = \sum_{v(s)} N(v(s)|s)\pi(a|v(s))$ for each s, a, these

equations can not necessarily guarantee a unique π especially when these equations behind this condition are underdetermined. In such scenario, multiple policies π_r will exist as long as they satisfy the equations above.

Proof of (2) Firstly, based on Theorem 1 [116] that shows an optimal policy does not always exist, we assume that an optimal policy exists in the adversarial noisy state setting for the convenience of following analysis. Based on this assumption, we need to derive the explicit value function under the adversary. Inspired by [116], the proof insight is that the behavior of optimal adversary can be also viewed as finding another optimal policy, yielding a zero-sum two player game. Specifically, in the SN-MDP setting, the adversary selects an action $\hat{a} \in \mathcal{S}$ satisfying $\hat{a} = v(s)$, attempting to maximize its state-action value function $\tilde{Q}_{\pi_a}(s, \hat{a})$. Then the adversary's value function $\hat{V}_{\pi_a}(s)$ can be formulated as:

$$\begin{aligned}
\hat{V}_{\pi_a}(s) &= \max_{\hat{a}} \hat{Q}_{\pi_a}(s, \hat{a}) \\
&= \max_{\hat{a}} \sum_{s'} \hat{p}(s'|s, \hat{a})(\hat{R}(s, \hat{a}, s') + \gamma \hat{V}_{\pi_a}(s')) \\
&= \max_{v(s)} \sum_{s'} \sum_{a} \pi(a|v(s)) p(s'|s, a)(-R(s, a, s') + \gamma \hat{V}_{\pi_a}(s'))
\end{aligned} \tag{4.14}$$

where $\hat{p}(s'|s, \hat{a})$ is the transition dynamics of the adversary, satisfying $\hat{p}(s'|s, \hat{a}) = \sum_a \pi(a|v(s)) p(s'|s, a)$ from the perspective of the agent. $\hat{R}(s, \hat{a}, s')$ is the adversary's reward function while taking action $\hat{a}$, which is the opposite number of $R(s, a, s')$ given the action a. In addition, since both the adversary and agent can serve as a zero-sum two-player game, it indicates that $\tilde{V}_{\pi_a}(s) = -\hat{V}_{\pi_a}(s)$ for the agent's value function $\tilde{V}_{\pi_a}$ in the adversary setting. Then we rearrange the equation above as follows:

$$\begin{aligned}
\tilde{V}_{\pi_a}(s) &= -\hat{V}_{\pi_a}(s) \\
&= -\min_{N(\cdot|s)} \sum_{s'} \sum_{a} \pi'(a|s) p(s'|s, a)(-R(s, a, s') - \gamma \tilde{V}_{\pi_a}(s')) \\
&= \min_{v(s)} \sum_{s'} \sum_{a} \pi'(a|s) p(s'|s, a)(R(s, a, s') + \gamma \tilde{V}_{\pi_a}(s')) \\
&= \min_{N(\cdot|s)} \sum_{s'} \sum_{a} \pi'(a|s) p(s'|s, a)(r_{t+1} + \gamma \min_{N} \mathbb{E}_{\pi \circ N} \left[\sum_{k=0}^{\infty} r_{t+k+2} | s_{t+1} = s' \right]) \\
&= \min_{N} \tilde{V}_{\pi \circ N}(s)
\end{aligned} \tag{4.15}$$

Note that we optimize over N, which means we consider $N(\cdot|s)$ for each state s. Further, we derive the contraction of the Bellman operator $\mathcal{T}_a^\pi$. We rewrite our Bellman Operator $\mathcal{T}_a^\pi$ as:

$$\mathcal{T}_a^\pi \tilde{V}_{\pi\circ N}(s) = \min_N \tilde{V}_{\pi\circ N}(s) = \min_N \boldsymbol{R}(s) + \gamma \sum_{s'} \boldsymbol{P}'_{s,s'} \tilde{V}_{\pi\circ N}(s') \tag{4.16}$$

We firstly assume $\mathcal{T}_a^\pi \tilde{V}^1_{\pi_a}(s) \geq \mathcal{T}_a^\pi \tilde{V}^2_{\pi_a}(s)$, then we have:

$$\begin{aligned}
&\mathcal{T}_a^\pi \tilde{V}^1_{\pi\circ N}(s) - \mathcal{T}_a^\pi \tilde{V}^2_{\pi\circ N}(s) \\
&\leq \max_{N(\cdot|s)} \{\gamma \sum_{s'} \boldsymbol{P}'_{s,s'} \tilde{V}^1_{\pi\circ N}(s') - \gamma \sum_{s'} \boldsymbol{P}'_{s,s'} \tilde{V}^2_{\pi\circ N}(s')\} \\
&\leq \gamma \max_{N(\cdot|s)} \sum_{s'} \boldsymbol{P}'_{s,s'} |\tilde{V}^1_{\pi\circ N}(s') - \tilde{V}^2_{\pi\circ N}(s')| \\
&\leq \gamma \max_{N(\cdot|s)} \sum_{s'} \boldsymbol{P}'_{s,s'} \max_s |\tilde{V}^1_{\pi\circ N}(s') - \tilde{V}^2_{\pi\circ N}(s')| \\
&= \gamma \max_{N(\cdot|s)} \sum_{s'} \boldsymbol{P}'_{s,s'} \|\tilde{V}^1_{\pi\circ N} - \tilde{V}^2_{\pi\circ N}\|_\infty \\
&\leq \gamma \|\tilde{V}^1_{\pi\circ N} - \tilde{V}^2_{\pi\circ N}\|_\infty
\end{aligned} \tag{4.17}$$

where the first inequality holds as $\min_{x_1} f(x_1) - \min_{x_2} g(x_2) \leq \max_x (f(x) - g(x))$ and we extends this inequality into the Wasserstein distance in the proof of convergence of distributional RL setting in Appendix 4.7.2. The last inequality holds since only $\boldsymbol{P}'_{s,s'}$ depends on $N(\cdot|s)$ while the infinity norm is a constant, which is independent with the current $N(\cdot|s)$. Similarly, the other scenario can be still proved. Thus, we have:

$$\|\mathcal{T}_a^\pi \tilde{V}^1_{\pi\circ N} - \mathcal{T}_a^\pi \tilde{V}^2_{\pi\circ N}\|_\infty \leq \gamma \|\tilde{V}^1_{\pi\circ N} - \tilde{V}^2_{\pi\circ N}\|_\infty \tag{4.18}$$

Thus, we proved that $\mathcal{T}_a^\pi$ is still a contraction and converge to $\min_N \tilde{V}_{\pi\circ N}$. We denote it as $\tilde{V}_{\pi\circ N^*}$ In addition, based on the insight of the "merged" policy π'_a, we have $\pi'_a = \sum_{v(s)} N^*(v(s)|s)\pi(a|v(s)) = \pi(a|v^*(s))$ where the deterministic state $v^*(s)$ is the adversarial noisy state from the state s.

□

4.7.2 Proof of Theorem 5

Proof. The p-Wasserstein metric d_p is defined as

$$d_p = \left(\int_0^1 \left| F_{Z^*}^{-1}(\omega) - F_{Z_\theta}^{-1}(\omega) \right|^p d\omega \right)^{1/p}, \tag{4.19}$$

which minimizes the distance between the true value distribution Z^* and the parametric distribution Z_θ. F^{-1} is the inverse cumulative distribution function of a random variable with the cumulative distribution function as F. The following contraction proof is in the maximal form of d_p, denoted by $\bar{d}_p$.

Proof of (1) This contraction proof is similar to the original one [9] in the distributional RL without state observation noises. The only difference lies in the new transition operator $\mathcal{P}_r^\pi$, but it dose not change the main proof process. For two different random variables Z_N^1 and Z_N^2 about returns, we have:

$$\begin{aligned}
&\bar{d}_p(\mathfrak{T}_r^\pi Z_N^1, \mathfrak{T}_r^\pi Z_N^2) \\
&= \sup_{s,a} d_p(\mathfrak{T}_r^\pi Z_N^1(s,a), \mathfrak{T}_r^\pi Z_N^2(s,a)) \\
&= \sup_{s,a} d_p(R(s,a,S') + \gamma \mathcal{P}_r^\pi Z_N^1(s,a), R(s,a,S') + \gamma \mathcal{P}_r^\pi Z_N^2(s,a)) \\
&\leq \gamma \sup_{s,a} d_p(\mathcal{P}_r^\pi Z_N^1(s,a), \mathcal{P}_r^\pi Z_N^2(s,a)) \\
&\leq \gamma \sup_{s,a} \sup_{s',a'} d_p(Z_N^1(s',a'), Z_N^2(s',a')) \\
&= \gamma \sup_{s',a'} d_p(Z_1(s',a'), Z_2(s',a')) \\
&= \gamma \sup_{s,a} d_p(Z_N^1(s,a), Z_N^2(s,a)) \\
&= \gamma \bar{d}_p(Z_N^1, Z_N^2).
\end{aligned} \tag{4.20}$$

Thus, we conclude that $\mathfrak{T}_r^\pi : \mathcal{Z} \to \mathcal{Z}$ is a γ-contraction in $\bar{d}_p$.

Proof of (2) Recap the distributional Bellman optimality operator $\mathfrak{T}$ in MDP is defined as $\mathfrak{T}Z(s,a) :\overset{D}{=} R(s,a,S') + \gamma Z(S', \pi_Z(s'))$, where $S' \sim P(\cdot|s,a)$ and $\pi_Z(S') = \arg\max_{a'} \mathbb{E}[Z(S',a')]$. By contrast, in SN-MDP, Our greedy

adversarial rule $N^*(\cdot|s')$ is based on the greedy policy rule in distributional Bellman optimality operator, which attempts to find adversarial $N^*(\cdot|s')$ in order to minimize $\mathbb{E}\left[Z_N(s', a')\right]$, where $a' \sim \pi(\cdot|V(s'))$ and $V(s') \sim N(\cdot|s')$. As $N^*(\cdot|s')$ yields a deterministic state s^*, the agent always takes action based on s^*, which we denote as $A^* \sim \pi(\cdot|s^*)$. Therefore, we can obtain the state-action function $Q^\pi_{N^*}(s, a)$ under the adversary as

$$Q^\pi_{N^*}(s, a) = \min_N \mathbb{E}\left[Z^\pi_N(s, a)\right] = \mathbb{E}\left[Z^{\pi^*}(s, a)\right] \tag{4.21}$$

where $\pi^*(\cdot|s) = \pi(\cdot|s^*)$ for $\forall s$ that follows the adversarial policy A^*.

Next, to derive the contractive property of $\mathfrak{T}^\pi_a$, we denote two state-action valued distributions as $Z^1_N(s, a)$ and $Z^2_N(s, a)$. Then we have:

$$\begin{aligned}
\bar{d}_p(\mathfrak{T}^\pi_a Z^1_N, \mathfrak{T}^\pi_a Z^2_N) &= \sup_{s,a} d_p(\mathfrak{T}^\pi_a Z^1_N(s, a), \mathfrak{T}^\pi_a Z^2_N(s, a)) \\
&= \sup_{s,a} d_p(R(s, a, S') + \gamma \mathcal{P}^\pi_a Z^1_N(s, a), R(s, a, S') + \gamma \mathcal{P}^\pi_a Z^2_N(s, a)) \\
&\leq \gamma \sup_{s,a} \sum_{s'} P(s'|s, a) d_p(Z^1_N(s', A^*), Z^2_N(s', A^*)) \\
&= \gamma \sum_{s'} P(s'|s, a) d_p(Z^1_N(s', A^*), Z^2_N(s', A^*)) \\
&\leq \gamma \sup_{s'} d_p(Z^1_N(s', A^*), Z^2_N(s', A^*)) \\
&= \gamma \sup_{s'} d_p(\sum_{a'_*} \pi(a'_*|s^*) Z^1_N(s', a'_*), \sum_{a'_*} \pi(a'_*|s^*) Z^2_N(s', a'_*)) \\
&\leq \gamma \sup_{s'} \sum_{a'_*} \pi(a'_*|s^*) d_p(Z^1_N(s', a'_*), Z^2_N(s', a'_*)) \\
&\leq \gamma \sup_{s', a'_*} d_p(Z^1_N(s', a'_*), Z^2_N(s', a'_*)) \\
&= \gamma \sup_{s,a} d_p(Z^1_N(s, a), Z^2_N(s, a)) \\
&= \gamma \bar{d}_p(Z^1_N, Z^2_N)
\end{aligned} \tag{4.22}$$

Thus, we conclude that $\mathfrak{T}^\pi_a$ is still a γ-contraction in $\bar{d}_p$. □

4.7.3 Proof of Theorems 9 and 10

Proof. Firstly, we prove Theorem 9. We show the derivation details of the Histogram distribution loss starting from KL divergence between p and $q_{\mathbf{w}}$. p_i is the cumulative probability increment of target distribution $\mathfrak{T}Z_{\mathbf{w}}$ within the i-th bin, and $q_{\mathbf{w}}$ corresponds to a (normalized) histogram, and has density values $\frac{f_i^{\mathbf{w}}(\mathbf{x}(s))}{w_i}$ per bin. Thus, we have:

$$\begin{aligned}
\mathcal{L}(Z_{\mathbf{w}}, \mathfrak{T}Z_{\mathbf{w}}) &= -\int_a^b p(y) \log q_{\mathbf{w}}(y) dy \\
&= -\sum_{i=1}^{k} \int_{l_i}^{l_i+w_i} p(y) \log \frac{f_i^{\mathbf{w}}(\mathbf{x}(s))}{w_i} dy \\
&= -\sum_{i=1}^{k} \log \frac{f_i^{\mathbf{w}}(\mathbf{x}(s))}{w_i} \underbrace{\left(F_{\mathfrak{T}Z_{\mathbf{w}}}(l_i + w_i) - F_{\mathfrak{T}Z_{\mathbf{w}}}(l_i)\right)}_{p_i} \\
&\doteq -\sum_{i=1}^{k} p_i \log f_i^{\mathbf{w}}(\mathbf{x}(s))
\end{aligned} \tag{4.23}$$

where the last line holds as the width parameter w_i can be ignored and thus the loss function is proportion to the final term. Next, we compute the gradient of the Histogram distributional loss in the linear approximation case.

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{x}(s)} \sum_{j=1}^{k} p_j \log f_j^{\mathbf{w}}(\mathbf{x}(s)) &= \sum_{j=1}^{k} p_j \frac{1}{f_j^{\mathbf{w}}(\mathbf{x}(s))} \nabla f_j^{\mathbf{w}}(\mathbf{x}(s)) \\
&= \sum_{j=1}^{k} p_j \frac{1}{f_j^{\mathbf{w}}(\mathbf{x}(s))} f_j^{\mathbf{w}}(\mathbf{x}(s)) \sum_{i=1}^{k} \frac{\exp(\mathbf{x}(s)^\top \mathbf{w}_i)}{\sum_{p=1}^{k} \exp(\mathbf{x}(s)^\top \mathbf{w}_p)} (\mathbf{w}_j - \mathbf{w}_i) \\
&= \sum_{j=1}^{k} p_j \sum_{i=1}^{k} f_i^{\mathbf{w}}(\mathbf{x}(s)) (\mathbf{w}_j - \mathbf{w}_i) \\
&= \sum_{j=1}^{k} p_j \mathbf{w}_j - \sum_{i=1}^{k} f_i^{\mathbf{w}}(\mathbf{x}(s)) \mathbf{w}_i \\
&= \sum_{i=1}^{k} (p_i - f_i^{\mathbf{w}}(\mathbf{x}(s))) \mathbf{w}_i
\end{aligned} \tag{4.24}$$

Then, as we have $\|\mathbf{w}_i\| \leq l$ for $\forall i$, we bound the norm of its gradient

$$\begin{aligned} \|\frac{\partial}{\partial \mathbf{x}(s)} \sum_{j=1}^{k} p_j \log f_j^{\mathbf{w}}(\mathbf{x}(s))\| &\leq \sum_{i=1}^{k} \|(p_i - f_i^{\mathbf{w}}(\mathbf{x}(s)))\mathbf{w}_i\| \\ &= \sum_{i=1}^{k} |p_i - f_i^{\mathbf{w}}(\mathbf{x}(s))| \|\mathbf{w}_i\| \\ &\leq kl \end{aligned} \tag{4.25}$$

The last equality satisfies because $|p_i - f_i^{\mathbf{w}}(\mathbf{x}(s))|$ is less than 1 and even smaller. In summary, compared with the least squared loss in expectation-based RL, the histogram distributional loss in distributional RL has the bounded gradient norm regarding the state features $\mathbf{x}(s)$. This upper bound of gradient norm can mitigate the impact of the noises on state observations on the loss function, therefore yielding training robustness for distributional RL.

Next, we prove the Proposition 10. Its proof is similar to Proposition 9. Firstly, we know that $f_i^{\mathbf{w},\theta}(\mathbf{x}(s)) = \exp\left(\phi_{\mathbf{w}}(\mathbf{x}(s))^\top \theta_i\right) / \sum_{j=1}^{k} \exp\left(\phi_{\mathbf{w}}(\mathbf{x}(s))^\top \theta_j\right)$ and $\phi_{\mathbf{w}}(\cdot)$ is L-Lipschitz, i.e., $\|\phi_{\mathbf{w}}(x) - \phi_{\mathbf{w}}(y)\| \leq L\|x - y\|$. Then

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{x}(s)} \sum_{j=1}^{k} p_j \log f_j^{\mathbf{w},\theta}(\mathbf{x}(s)) \\ &= \sum_{j=1}^{k} p_j \frac{1}{f_j^{\mathbf{w},\theta}(\mathbf{x}(s))} f_j^{\mathbf{w},\theta}(\mathbf{x}(s)) \sum_{i=1}^{k} \frac{\exp(\mathbf{x}(s)^\top \mathbf{w}_i)}{\sum_{p=1}^{k} \exp(\mathbf{x}(s)^\top \mathbf{w}_p)} (\nabla_{\mathbf{x}} \phi_{\mathbf{w}}^\top \theta_j - \nabla_{\mathbf{x}} \phi_{\mathbf{w}}^\top \theta_i) \\ &= \sum_{j=1}^{k} p_j \sum_{i=1}^{k} f_i^{\mathbf{w},\theta}(\mathbf{x}(s)) (\nabla_{\mathbf{x}} \phi_{\mathbf{w}}^\top \theta_j - \nabla_{\mathbf{x}} \phi_{\mathbf{w}}^\top \theta_i) \\ &= \sum_{j=1}^{k} p_j \nabla_{\mathbf{x}} \phi_{\mathbf{w}}^\top \theta_j - \sum_{i=1}^{k} f_i^{\mathbf{w},\theta}(\mathbf{x}(s)) \nabla_{\mathbf{x}} \phi_{\mathbf{w}}^\top \theta_i \\ &= \sum_{i=1}^{k} (p_i - f_i^{\mathbf{w},\theta}(\mathbf{x}(s))) \nabla_{\mathbf{x}} \phi_{\mathbf{w}}^\top \theta_i \end{aligned} \tag{4.26}$$

Then, as we have $\|\theta_i\| \leq l$ for $\forall i$, we bound the norm of its gradient

$$\begin{aligned} \|\frac{\partial}{\partial \mathbf{x}(s)} \sum_{j=1}^{k} p_j \log f_j^{\mathbf{w},\theta}(\mathbf{x}(s))\| &\leq \sum_{i=1}^{k} \|(p_i - f_i^{\mathbf{w},\theta}(\mathbf{x}(s)))\nabla_{\mathbf{x}} \phi_{\mathbf{w}}^{\top} \theta_i\| \\ &= \sum_{i=1}^{k} |p_i - f_i^{\mathbf{w},\theta}(\mathbf{x}(s))| \|\nabla_{\mathbf{x}} \phi_{\mathbf{w}}^{\top} \theta_i\| \\ &\leq klL \end{aligned} \tag{4.27}$$

The last inequality holds because $|\phi_{\mathbf{w}}(x)^{\top}\theta_i - \phi_{\mathbf{w}}(y)^{\top}\theta_i| \leq \|\phi_{\mathbf{w}}(x) - \phi_{\mathbf{w}}(y)\| \|\theta_i\| \leq lL\|x - y\|$. Thus the function $\phi_{\mathbf{w}}^{\top}\theta_i$ can be viewed as Ll-Lipschitz continuous, indicating that $\|\nabla_{\mathbf{x}} \phi_{\mathbf{w}}^{\top} \theta_i\| \leq lL$. □

4.7.4 TD Convergence Under Noisy State Observations

Theorem 8. *(Conditions for TD Convergence under Noisy State Observations) Define* $\mathbf{P}$ *as the* $|\mathcal{S}| \times |\mathcal{S}|$ *matrix forming from* $p(s'|s)$, $\mathbf{D}$ *as the* $|\mathcal{S}| \times |\mathcal{S}|$ *diagonal matrix with* $\mu(s)$ *on its diagonal, and* $\mathbf{X}$ *as the* $|\mathcal{S}| \times d$ *matrix with* $\mathbf{x}(s)$ *as its rows, and* $\mathbf{E}$ *is the* $|\mathcal{S}| \times d$ *perturbation matrix with each perturbation vector* $\mathbf{e}(s)$ *as its rows. The stepsizes* $\alpha_t \in (0, 1]$ *satisfy* $\sum_{t=0}^{\infty} \alpha_t < \infty$ *and* $\sum_{t=0}^{\infty} \alpha_t^2 = 0$. *For noisy states, we consider the following three cases:* ***(i)*** $\mathbf{e}(s)$ *on current state features, i.e.,* $\mathbf{x}_t \leftarrow \mathbf{x}_t + \mathbf{e}_t$, ***(ii)*** $\mathbf{e}(s')$ *on next state features, i.e.,* $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_{t+1} + \mathbf{e}_{t+1}$, ***(iii)*** *the same* $\mathbf{e}$ *on both state features. We can attain that* $\mathbf{w}_t$ *converges to TD fixed point if the following conditions are satisfied, respectively.*

Case (i): *both* $\mathbf{A}$ *and* $(\mathbf{X} + \mathbf{E})^{\top}\mathbf{DPE}$ *are positive definite.* ***Case (ii):*** *both* $\mathbf{A}$ *and* $-\mathbf{X}^{\top}\mathbf{DPE}$ *are positive definite.* ***Case (iii):*** $\mathbf{A}$ *is positive definite.*

From the convergence conditions for the three cases in Theorem 8, it is clear that (iii) is the mildest. This is the same condition as that in the normal TD learning without noisy state observations. Note that the case (iii) can be viewed as the SN-MDP setting, whose convergence has been already rigorously analyzed in Section 4.3. In Section 4.5, our experiments demonstrate that both expectation-based and distribution RL are more likely to converge in case (iii) compared with case (i) and (ii).

In cases (i) and (ii), the positive definiteness of $\mathbf{X}^\top\mathbf{DPE} + \mathbf{E}^\top\mathbf{DPE}$ and $-\mathbf{X}^\top\mathbf{DPE}$ is crucial. We partition $(\mathbf{X}+\mathbf{E})^\top\mathbf{DPE}$ into $\mathbf{X}^\top\mathbf{DPE} + \mathbf{E}^\top\mathbf{DPE}$, where the first term has the opposite positive definiteness to $-\mathbf{X}^\top\mathbf{DPE}$, and the second term is positive definite [102]. Based on these observations, we discuss the subtle convergence relationship in cases (i) and (ii):

(1) If $-\mathbf{X}^\top\mathbf{DPE}$ is positive definite, which indicates that TD is convergent in case (ii), TD can still converge in case (i) **unless** the positive definiteness of $\mathbf{E}^\top\mathbf{DPE}$ dominates in $\mathbf{X}^\top\mathbf{DPE} + \mathbf{E}^\top\mathbf{DPE}$.

(2) If $-\mathbf{X}^\top\mathbf{DPE}$ is negative definite, TD is likely to diverge in case (ii). By contrast, TD will converge in case (i).

In summary, there exists a subtle trade-off of TD convergence in case (i) and (ii) if we approximately ignore the term $\mathbf{E}^\top\mathbf{DPE}$ in case (i). The key of it lies in the positive definiteness of the matrix $\mathbf{X}^\top\mathbf{DPE}$, which heavily depends on the task. In Section 4.5, we empirically verify that the convergence situations for current and next state observations are normally different. Which situation is superior is heavily dependent on the task.

Proof. To prove the convergence of TD under the noisy states, we use the results from [12] that require the condition about stepsizes α_t holds: $\sum_{t=0}^{\infty}\alpha_t < \infty$ and $\sum_{t=0}^{\infty}\alpha_t^2 = 0$. Our part proof is directly established on [102]. Particularly, the positive definiteness of $\mathbf{A}$ will determine the TD convergence. For linear TD(0), in the continuing case with $\gamma < 1$, $\mathbf{A}$ can be re-written as:

$$\begin{aligned}
\mathbf{A} &= \sum_s \mu(s) \sum_a \pi(a|s) \sum_{r,s'} p(r,s'|s,a)\mathbf{x}_t \left(\mathbf{x}_t - \gamma\mathbf{x}_{t+1}\right)^\top \\
&= \sum_s \mu(s) \sum_{s'} p(s'|s)\mathbf{x}_t \left(\mathbf{x}_t - \gamma\mathbf{x}_{t+1}\right)^\top \\
&= \sum_s \mu(s)\mathbf{x}_t (\mathbf{x}_t - \gamma \sum_{s'} p(s'|s)\mathbf{x}_{t+1})^\top \\
&= \mathbf{X}^\top\mathbf{D}\mathbf{X} - \mathbf{X}^\top\mathbf{D}\gamma\mathbf{P}\mathbf{X} \\
&= \mathbf{X}^\top\mathbf{D}(\mathbf{I} - \gamma\mathbf{P})\mathbf{X}
\end{aligned} \tag{4.28}$$

Then we use $\mathbf{A}_t$ to present the convergence matrix in the case (i) where the perturbation vector $\mathbf{e}_t$ is added onto the current state features, i.e., $\mathbf{x}_t \leftarrow$

$\mathbf{x}_t + \mathbf{e}_t$, while we use $\mathbf{A}_{t+1}$ and $\mathbf{A}_{t,t+1}$to present the counterparts in the case (ii) and (iii), respectively. Based on Eq. 4.28, in the case (iii), we have:

$$\begin{aligned}\mathbf{A}_{t,t+1} &= (\mathbf{X}+\mathbf{E})^\top \mathbf{D}(\mathbf{X}+\mathbf{E}) - (\mathbf{X}+\mathbf{E})^\top \mathbf{D}\gamma\mathbf{P}(\mathbf{X}+\mathbf{E}) \\ &= (\mathbf{X}+\mathbf{E})^\top \mathbf{D}(\mathbf{I}-\gamma\mathbf{P})(\mathbf{X}+\mathbf{E})\end{aligned} \tag{4.29}$$

From [102], we know that the inner matrix $\mathbf{D}(\mathbf{I}-\gamma\mathbf{P})$ is the key to determine the positive definiteness of $\mathbf{A}$. If we assume that $\mathbf{A}$ is positive definite, which also indicates that $\mathbf{D}(\mathbf{I}-\gamma\mathbf{P})$ is positive definite equivalently. As such, $\mathbf{A}_{t,t+1}$ is positive definite automatically, and thus the liner TD would converge to the TD fixed point. Next, in the case (i) we have:

$$\begin{aligned}\mathbf{A}_t &= (\mathbf{X}+\mathbf{E})^\top \mathbf{D}(\mathbf{X}+\mathbf{E}) - (\mathbf{X}+\mathbf{E})^\top \mathbf{D}\gamma\mathbf{P}\mathbf{X} \\ &= \mathbf{A} + \mathbf{X}^\top\mathbf{D}\mathbf{E} + \mathbf{E}^\top\mathbf{D}\mathbf{X} + \mathbf{E}^\top\mathbf{D}\mathbf{E} - \mathbf{E}^\top\mathbf{D}\gamma\mathbf{P}\mathbf{X} \\ &= (\mathbf{X}+\mathbf{E})^\top \mathbf{D}(\mathbf{I}-\gamma\mathbf{P})(\mathbf{X}+\mathbf{E}) + (\mathbf{X}+\mathbf{E})^\top \mathbf{D}\gamma\mathbf{P}\mathbf{E} \\ &= \mathbf{A}_{t,t+1} + \gamma(\mathbf{X}+\mathbf{E})^\top\mathbf{D}\mathbf{P}\mathbf{E} \\ &= \mathbf{A}_{t,t+1} + \gamma(\mathbf{X}^\top\mathbf{D}\gamma\mathbf{P}\mathbf{E} + \mathbf{E}^\top\mathbf{D}\gamma\mathbf{P}\mathbf{E})\end{aligned} \tag{4.30}$$

Similarly, in the case (ii), we can also attain:

$$\begin{aligned}\mathbf{A}_{t+1} &= \mathbf{X}^\top\mathbf{D}\mathbf{X} - \mathbf{X}^\top\mathbf{D}\gamma\mathbf{P}(\mathbf{X}+\mathbf{E}) \\ &= \mathbf{A} - \gamma\mathbf{X}^\top\mathbf{D}\mathbf{P}\mathbf{E}\end{aligned} \tag{4.31}$$

We know that the positive definiteness of $\mathbf{A}$ and $\mathbf{A}_{t,t+1}$ is only determined by the positive definiteness of the inner matrix $\mathbf{D}(\mathbf{I}-\gamma\mathbf{P})$. If we assume the positive definiteness of $\mathbf{A}$, i.e., the positive definiteness of $\mathbf{A}_{t,t+1}$ and $\mathbf{D}(\mathbf{I}-\gamma\mathbf{P})$, as $\gamma > 0$, what we only need to focus on are the positive definiteness of $\mathbf{X}^\top\mathbf{D}\mathbf{P}\mathbf{E}+\mathbf{E}^\top\mathbf{D}\mathbf{P}\mathbf{E}$ and $-\mathbf{X}^\top\mathbf{D}\mathbf{P}\mathbf{E}$. If they are positive definite, TD learning will converge under their cases, respectively. □

4.7.5 Sensitivity Analysis by Influence Function

Next, we conduct an outlier analysis by the *influence function*, a key facet in the robust statistics [49]. The influence function characterizes the effect that

the noise in particular observation has on an estimator, and can be utilized to investigate the impact of one particular state observation noise on the training of reinforcement learning algorithms. Specifically, suppose that F_ϵ is the contaminated distribution function that combines the clear data distribution F and an outlier x. The distribution F_ϵ can be defined as

$$F_\epsilon = (1-\epsilon)F + \epsilon\delta_x, \tag{4.32}$$

where δ_x is a probability measure assigning probability 1 to x. Let $\hat{\theta}$ be a regression estimator. The influence function of θ at F, $\psi : \mathcal{X} \to \Gamma$ is defined as

$$\psi_{\hat{\theta},F}(x) = \lim_{\epsilon \to 0} \frac{\hat{\theta}\left(F_\epsilon(x)\right) - \hat{\theta}(F)}{\epsilon}. \tag{4.33}$$

Mathematically, the influence function is the Gateaux derivative of θ at F in the direction δ_x. Owing to the fact that traditional value-based RL algorithms, e.g., DQN [74], can be viewed as a regression problem [28], the linear TD approximator also has a strong connection with regression problems. Based on this correlation, in the following Theorem 9, we quantitatively evaluate the influence function of TD learning in the case of linear function approximation.

Theorem 9. *(Influence Function Analysis in TD Learning with linear function approximation) Denote* $d_t = \mathbf{x}_t - \gamma\mathbf{x}_{t+1} \in \mathbb{R}^d$, *and* $\mathbf{A} \doteq \mathbb{E}\left[\mathbf{x}_t d_t^\top\right] \in \mathbb{R}^{d\times d}$. *Let* F_π *be the data distribution generated from the environment dynamics given a policy* π. *Consider an outlier pair* $(\mathbf{x}_t, \mathbf{x}_{t+1})$ *with the reward* R_{t+1}, *the influence function* ψ *of this pair on the estimator* $\mathbf{w}$ *is derived as*

$$\psi_{\mathbf{w},F_\pi}(\mathbf{x}_t, \mathbf{x}_{t+1}) = \mathbb{E}(\mathbf{A}^\top\mathbf{A})^{-1} d_t \mathbf{x}_t^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w}). \tag{4.34}$$

Please refer to Appendix 4.7.5 for the proof. Theorem 9 shows the quantitative impact of an outlier pair $(\mathbf{x}_t, \mathbf{x}_{t+1})$ on the learned parameter $\mathbf{w}$. Moreover, a corollary can be immediately obtained to make a precise comparison of the impacts of perturbations on current and next state features.

Corollary 1. *Given the same perturbation* η *on either current or next state features, i.e.,* $\mathbf{x}_t$, *and* $\mathbf{x}_{t+1}$, *at the step* t, *if we approximate* $\eta\eta^\top\mathbf{x}_t$ *and* $\eta\eta^\top\mathbf{w}$

as $\mathbf{0}$ *as* η *is small enough, the following relationship between the resulting variations of influence function,* $\Delta_{\mathbf{x}_t}\psi$ *and* $\Delta_{\mathbf{x}_{t+1}}\psi$*, holds:*

$$\gamma\Delta_{\mathbf{x}_t}\psi + \Delta_{\mathbf{x}_{t+1}}\psi = 2\gamma d_t \eta \mathbf{x}_t^\top (R_{t+1} - d_t^\top \mathbf{w}). \tag{4.35}$$

We provide the proof of Corollary 1 in Appendix 4.7.5. Under this equation, the sensitivity of noises on $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$, measured by $\Delta_{\mathbf{x}_t}\psi$ and $\Delta_{\mathbf{x}_{t+1}}\psi$, present a trade-off relationship as their weighted sum is definite. However, there is not an ordered relationship between $\Delta_{\mathbf{x}_t}\psi$ and $\Delta_{\mathbf{x}_{t+1}}\psi$. In summary, we conclude that the sensitivity of current and next state features against perturbations is normally divergent, and the degree of sensitivity is heavily determined by the task. These conclusions are similar to those we derived in the TD convergence part.

Proof. We combine the proof of Theorem 9 and Corollary 1 together. The TD fixed point $\mathbf{w}_{\text{TD}}$ to the system satisfies $\mathbf{A}\mathbf{w}_{\text{TD}} = \mathbf{b}$. Thus, the TD convergence point, i.e., TD fixed point, can be attained by solving the following regression problem:

$$\min_{\mathbf{w}} \|\mathbf{b} - \mathbf{A}\mathbf{w}\|^2 \tag{4.36}$$

To derive the influence function, consider the contaminated distribution which puts a little more weight on the outlier pair $(\mathbf{x}_t, \mathbf{x}_{t+1})$:

$$\begin{aligned}\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} (1-\epsilon)\mathbb{E}[(\mathbf{b} - \mathbf{A}\mathbf{w})^\top (\mathbf{b} - \mathbf{A}\mathbf{w})] + \\ \epsilon(y_b - x_A^\top \mathbf{w})^\top (y_b - x_A^\top \mathbf{w}),\end{aligned} \tag{4.37}$$

where $y_b = R_{t+1}\mathbf{x}_t$ and $x_b = d_t \mathbf{x}_t^\top$. We take the first condition:

$$(1-\epsilon)\mathbb{E}(2\mathbf{A}^\top\mathbf{A}\mathbf{w} - 2\mathbf{A}^\top\mathbf{b}) - 2\epsilon x_A(y_b - x_A^\top \mathbf{w}) = 0. \tag{4.38}$$

Then we arrange this equality and obtain:

$$(1-\epsilon)\mathbb{E}(\mathbf{A}^\top\mathbf{A} + x_A x_A^\top)\mathbf{w}_\epsilon = (1-\epsilon)\mathbb{E}(\mathbf{A}^\top\mathbf{b}) + \epsilon x_A y_b. \tag{4.39}$$

Then we take the gradient on ϵ and let $\epsilon = 0$, then we have:

$$(-\mathbb{E}(\mathbf{A}^\top \mathbf{A}) + x_A x_A^\top)\mathbf{w}_\epsilon + \mathbb{E}(\mathbf{A}^\top \mathbf{A})\psi_{\mathbf{w},F_\pi} = -\mathbb{E}(\mathbf{A}^\top \mathbf{b}) + x_A y_b. \tag{4.40}$$

We know that under the least square estimation, the closed-form solution of $\mathbf{w}_\epsilon$ is $\mathbb{E}(\mathbf{A}^\top \mathbf{A})^{-1}\mathbb{E}(\mathbf{A}^\top \mathbf{b})$. Thus, after the simplicity, we finally attain:

$$\begin{aligned} \psi_{\mathbf{w},F_\pi}(\mathbf{x}_t, \mathbf{x}_{t+1}) &= \mathbb{E}(\mathbf{A}^\top \mathbf{A})^{-1} x_A (y_b - x_A^\top \mathbf{w}) \\ &= \mathbb{E}(\mathbf{A}^\top \mathbf{A})^{-1} d_t \mathbf{x}_t^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w}). \end{aligned} \tag{4.41}$$

Next, we prove the Corollary. We only need to focus on the item $d_t \mathbf{x}_t^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w})$, which we denote as ψ_0. Then we use $\Delta_{x_t}\psi$ and $\Delta_{x_{t+1}}\psi$ to represent the change of ψ after adding perturbations η on $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$, respectively. In particular, since we approximate $\eta\eta^\top \mathbf{x}_t$ and $\eta\eta^\top \mathbf{w}$ as $\mathbf{0}$, then we have that the change of influence function for the perturbation η on the current state feature $\mathbf{x}_t$:

$$\begin{aligned} \Delta_{x_t}\psi &\approx (d_t + \eta)(\mathbf{x}_t^\top \mathbf{x}_t + 2\eta^\top \mathbf{x}_t)(R_{t+1} - d_t^\top \mathbf{w} - \eta^\top \mathbf{w}) - \psi_0 \\ &\approx -d_t \mathbf{x}_t^\top \mathbf{x}_t \eta^\top \mathbf{w} + 2 d_t \eta^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w}) + \eta \cdot \mathbf{x}_t^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w}) \\ &= 2 d_t \eta^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w}) - \frac{1}{\gamma}(\gamma d_t \mathbf{x}_t^\top \mathbf{x}_t \eta^\top \mathbf{w} - \gamma \eta \mathbf{x}_t^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w})). \end{aligned} \tag{4.42}$$

Then the influence function for the perturbation η on the next state feature $\mathbf{x}_{t+1}$ is:

$$\begin{aligned} \Delta_{x_{t+1}}\psi &= (d_t - \gamma\eta)\mathbf{x}_t^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w} + \gamma \eta^\top \mathbf{w}) - \psi_0 \\ &\approx \gamma d_t \mathbf{x}_t^\top \mathbf{x}_t \eta^\top \mathbf{w} - \gamma \eta \mathbf{x}_t^\top \mathbf{x}_t (R_{t+1} - d_t^\top \mathbf{w}). \end{aligned} \tag{4.43}$$

Finally, it is easy to observe that the following relationship holds:

$$\gamma \Delta_{x_t}\psi = 2\gamma d_t \eta x_t^\top (R_{t+1} - d_t^\top \mathbf{w}) - \Delta_{x_{t+1}}\psi. \tag{4.44}$$

□

4.7.6 Experimental Setup

Noise Strength. We use Gaussian noise with different standard deviations. In particular, for a better presentation to compare the difference, we select proper standard deviations as 0.05, 0.1 in Cart Pole, 0.01, 0.0125 in Mountain Car, 0.01, 0.05 in Breakout and 0.05 in Qbert. For the adversarial noises, we select the perturbation sizes ϵ as 0.05, 0.1 in Cart Pole, 0.01, 0.1 in Mountain Car, 0.005, 0.01 in Breakout, and 0.005 in Qbert.

Distributional Loss. After a linear search, in the QR-DQN, We set $\kappa = 1$ for the Huber quantile loss across all tasks due to its smoothness.

Cart Pole After a linear search, in the QR-DQN, we set the number of quantiles N to be 20, and evaluate both DQN and QR-DQN on 200,000 training iterations.

Mountain Car After a linear search, in the QR-DQN, we set the number of quantiles N to be 2, and evaluate both DQN and QR-DQN on 100,000 training iterations.

Breakout and Qbert After a linear search, in the QR-DQN, we set the number of quantiles N to be 200, and evaluate both DQN and QR-DQN on 12,000,000 training iterations.

4.7.7 Discussion about More Adversarial Attacks

We are investigating more advanced adversarial attacks to further demonstrate the robustness advantage of distributional RL algorithms. [116] proposed Robust SARSA (RS) attack and Maximal Action Difference (MAD) attack, however, these two advanced attacked are specifically designed for PPO algorithm. Meanwhile, Stochastic gradient Langevin dynamics (SGLD) and convex relaxation attacks proposed by [116] are for DDPG algorithm. PGD attacks, serveing as the most natural attack for value-based RL algorithms, are leveraged to

evaluate SA-DQN algorithm. We also probe the training robustness of distributional RL algorithms under the more advanced **MAD attack** [116] on Ant, where Figure 4.5 still suggests a similar robustness result of distributional RL.

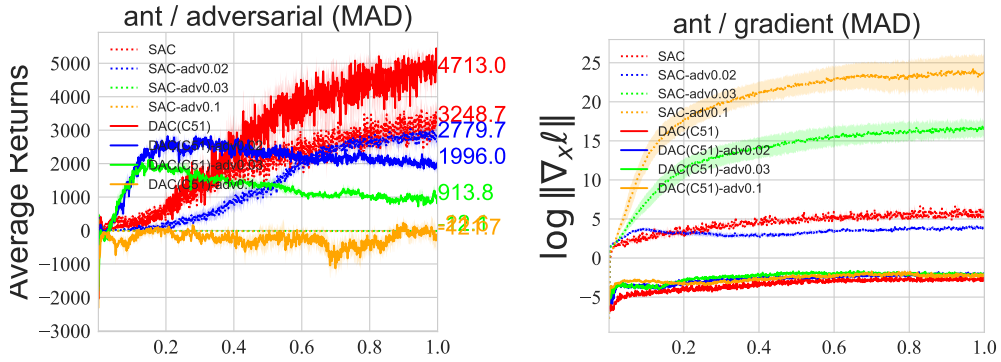


Figure 4.5: Robustness on MAD attack on Ant.

4.7.8 Experiments on D4PG

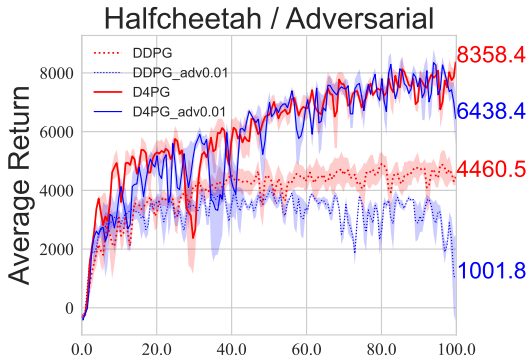


Figure 4.6: Average returns of DDPG and D4PG against **adversarial** state noises on Halfcheetah.

Chapter 5

Distributional Reinforcement Learning with Regularized Wasserstein Loss

5.1 Abstract

The empirical success of distributional reinforcement learning (RL) highly relies on the choice of distribution divergence equipped with an appropriate distribution representation. In this paper, we propose *Sinkhorn distributional RL (SinkhornDRL)*, which leverages Sinkhorn divergence—a regularized Wasserstein loss—to minimize the difference between current and target Bellman return distributions. Theoretically, we prove the contraction properties of SinkhornDRL, aligning with the interpolation nature of Sinkhorn divergence between Wasserstein distance and Maximum Mean Discrepancy (MMD). The introduced SinkhornDRL enriches the family of distributional RL algorithms, contributing to interpreting the algorithm behaviors compared with existing approaches by our investigation into their relationships. Empirically, we show that SinkhornDRL consistently outperforms or matches existing algorithms on the Atari games suite and particularly stands out in the multi-dimensional reward setting.

5.2 Introduction

Limitations of Typical Distributional RL Algorithms. Despite the gradual introduction of numerous algorithms, quantile regression-based algorithms [22, 21, 115, 85, 65, 86, 87] dominate attention and research in the realm of distributional RL. These algorithms utilize quantile regression to approximate the one-dimensional Wasserstein distance to compare two return distributions. Nevertheless, two major limitations hinder their performance improvement and wider practical deployment.

- *Inaccuracy in Capturing Return Distribution Characteristics.* The way of directly generating quantiles of return distributions via neural networks often suffers from the non-crossing issue [120], where the learned quantile curves fail to guarantee a non-decreasing property. This leads to abnormal distribution estimates and reduced model interpretability. The inaccurate distribution estimate is fundamentally attributed to the use of pre-specified statistics [85], while unrestricted statistics based on deterministic samples can be potentially more effective in complex environments [77].

- *Difficulties in Extension to Multi-dimensional Rewards.* Many RL tasks involve multiple sources of rewards [64, 23], hybrid reward architecture [104, 62], or sub-reward structures after reward decomposition [63, 117], which require learning multi-dimensional return distributions to reduce the intrinsic uncertainty of the environments. However, it remains elusive how to use quantile regressions to approximate a multi-dimensional Wasserstein distance, while circumventing the computational intractability issue in the related multi-dimensional output space.

Motivation of Sinkhorn Divergence: a Regularized Wasserstein loss. Sinkhorn divergence [95] has emerged as a theoretically principled and computationally efficient alternative for approximating Wasserstein distance. It has gained increasing attention in the field of optimal transport [7, 35, 31, 81] and has been successfully applied in various areas of machine learning [79, 36,

112, 30, 13]. By introducing entropic regularization, Sinkhorn divergence can efficiently approximate a multi-dimensional Wasserstein distance using computationally efficient matrix scaling algorithms [95, 81]. This makes it feasible to apply optimal transport distances to RL tasks with multi-dimensional rewards (see experiments in Section 5.6.3). Moreover, Sinkhorn divergence enables the leverage of samples to approximate return distributions instead of relying on pre-specified statistics, e.g., quantiles, thereby increasing the accuracy in capturing the full data complexity behind return distributions and naturally avoiding the non-crossing issues in distributional RL. Beyond addressing the two main limitations mentioned above, the well-controlled regularization introduced in Sinkrhorn divergence helps to find a "smoother" transport plan relative to Wasserstein distance, making it less sensitive to noises or small perturbations when comparing two return distributions (see Appendix 5.8.1 for the visualization). This regularization also aligns with the maximum-entropy principle [54, 24], which aims to maximize entropy while keeping the transportation cost constrained. Furthermore, the resulting strongly convex loss function [5] and the induced smoothness by regularization facilitate faster and more stable convergence in the deep RL setting (see more details in Sections 5.5 and 2.7).

Contributions. In this work, we propose a new family of distributional RL algorithms based on Sinkhorn divergence, a regularized Wasserstein loss, to address the limitations of quantile regression-based algorithms while promoting more stable training. As Sinkhorn divergence interpolates between Wasserstein distance and MMD [39, 31, 81], we probe this relationship in the RL context, characterizing the convergence properties of dynamic programming under Sinkhorn divergence and revealing the connections of different distances. Our study enriches the class of distributional RL algorithms, making them more effective for a broader range of scenarios and potentially inspiring advancement in other related areas of distribution learning. Our key contributions are summarized as follows:

1. **Algorithm.** We introduce a Sinkhorn distributional RL algorithm, called SinkhornDRL, which overcomes the primary shortcomings of pre-

dominantly utilized quantile regression-based algorithms. SinkhornDRL can be seamlessly integrated into existing model architectures and easily implemented.

2. **Theory.** We establish the properties of Sinkhorn divergence within distributional RL and derive the relevant convergence results for (multi-dimensional) distributional dynamic programming.

3. **Experiments.** We conduct an extensive comparison of SinkhornDRL with typical distributional RL algorithms across 55 Atari games, performing rigorous sensitivity analyses and computation cost assessments. We also verify the efficacy of SinkhornDRL in the multi-dimensional reward setting.

5.3 Preliminary Knowledge

We remain the notations of MDP, classical RL, and distributional RL as Chapters 2, 3, and 4. Next, we introduce the preliminary knowledge about divergences between probability measures.

Optimal Transport (OT) and Wasserstein / Earth Mover's Distance. The optimal transport (OT) metric W_c defines a powerful geometry to compare two probability measures (μ, ν), i.e., $W_c = \inf_{\Pi \in \mathbf{\Pi}(\mu,\nu)} \int c(x,y) \mathrm{d}\Pi(x,y)$, where c is the cost function, Π is the joint distribution with marginals (μ, ν), and the minimizer Π^* is called the *optimal transport plan* or *optimal coupling*. The p-Wasserstein distance $W_p = (\inf_{\Pi \in \mathbf{\Pi}(\mu,\nu)} \int \|x-y\|^p \mathrm{d}\Pi(x,y))^{1/p}$ is a special case of optimal transport with the Euclidean norm as the cost function. Relative to conventional divergences, including Hellinger, total variation or Kullback-Leibler divergences, the formulation of OT and Wasserstein distance inherently integrates the spatial or geometric relationships between data points and allows them to recover the full support of measures. This theoretical advantage comes, however, with a heavy computational price tag, especially in the high-dimensional space. Specifically, finding the optimal transport plan amounts

to solving a linear program and the cost scales at least in $\mathcal{O}(d^3 \log(d))$ when comparing two histograms of dimension d [20].

Maximum Mean Discrepancy [39]. Define two random variables X and Y. The squared Maximum Mean Discrepancy (MMD) MMD_k^2 with the kernel k is formulated as

$$\text{MMD}_k^2 = \mathbb{E}\left[k\left(X, X'\right)\right] + \mathbb{E}\left[k\left(Y, Y'\right)\right] - 2\mathbb{E}\left[k(X, Y)\right], \tag{5.1}$$

where $k(\cdot, \cdot)$ is a continuous kernel and X' (resp. Y') is a random variable independent of X (resp. Y). Mathematically, the "flat" geometry that MMD induces on the space of probability measures does not faithfully lift the ground distance [31], potentially inferior to OT when comparing two complicated distributions. However, MMD is cheaper to compute than OT with a smaller *sample complexity*, i.e., the number of samples for measures to approximate the true distance [35]. We provide more details of various distribution divergences as well as their existing contraction properties in Appendix 5.8.2.

Notations. We constantly use the *unrectified kernel* $k_\alpha = -\|x - y\|^\alpha$ in our algorithm analysis. With a slight abuse of notation, we also use Z_θ to denote θ parameterized return distribution.

5.4 Related Work

Based on the choice of distribution divergences and the distribution representation, distributional RL algorithms can be classified into three categories.

1. **Categorical Distributional RL.** As the first successful class, categorical distributional RL [9], e.g., C51, represents the return distribution using a categorical distribution with discrete fixed supports within a predefined interval.

2. **Quantile Regression (Wasserstein Distance) Distributional RL.** QR-DQN [22] employs quantile regression to approximate the one-dimensional

Wasserstein distance. It learns the quantile values for a series of fixed quantiles, offering greater flexibility in the support compared with categorical distributional RL. IQN [21] enhances this approach by utilizing an implicit model to produce more expressive quantile values, instead of fixed ones in QR-DQN, while FQF [115] further advances IQN by introducing a more expressive quantile network. However, as mentioned in Section 5.2, quantile regression distributional RL struggles with accurately capturing return distribution characteristics and handling multi-dimensional reward settings. SinkhornDRL, with the assistance of an entropy regularization, offers an alternative approach that addresses the two limitations simultaneously.

3. **MMD Distributional RL.** Rooted in kernel methods [39, 110], MMD-DQN [77] learns unrestricted statistics, i.e., samples, to represent the return distribution and optimizes under MMD, which can manage multi-dimensional rewards. However, the data geometry captured by MMD with a specific kernel may be limited, as it is highly sensitive to the characteristics of kernels and the induced Reproducing Kernel Hilbert space (RKHS) [36, 39, 34]. In contrast, SinkhornDRL is fundamentally based on OT, inherently capturing the spatial and geometric layout of return distributions. This enables SinkhornDRL to potentially surpass MMD-DQN by leveraging a richer representation of data geometry. In Section 2.7, we present extensive experiments to demonstrate the advantage of SinkhornDRL over MMD-DQN, particularly in the multi-dimensional reward scenario in Section 5.6.3.

5.5 Sinkhorn Distributional RL (SinkhornDRL)

The algorithmic evolution of distributional RL can be primarily viewed along two dimensions [77]. 1) Introducing new distributional RL families beyond the three established ones, leveraging alternative distribution divergences combined with suitable density estimation techniques. 2) Enhancing existing algorithms within a particular family by increasing their model capacity, e.g.,

IQN and FQF. Concretely, SinkhornDRL falls into the first dimension, aiming to expand the range of distributional RL algorithm families.

5.5.1 Sinkhorn Divergence and New Convergence Properties in Distributional RL

Sinkhorn divergence [95] efficiently approximates the optimal transport problem by introducing an entropic regularization. It aims at finding a sweet trade-off that simultaneously leverages the geometry property of Wasserstein distance (optimal transport distances) and the favorable sample complexity advantage and unbiased gradient estimates of MMD [36, 31]. For two probability measures μ and ν, the entropic regularized Wasserstein distance $\mathcal{W}_{c,\varepsilon}(\mu,\nu)$ is formulated as

$$\mathcal{W}_{c,\varepsilon}(\mu,\nu) = \min_{\Pi\in\mathbf{\Pi}(\mu,\nu)} \int c(x,y)\mathrm{d}\Pi(x,y) + \varepsilon \mathrm{KL}(\Pi|\mu\otimes\nu), \tag{5.2}$$

where the entropic regularization $\mathrm{KL}(\Pi|\mu\otimes\nu) = \int \log\left(\frac{\Pi(x,y)}{\mathrm{d}\mu(x)\mathrm{d}\nu(y)}\right)\mathrm{d}\Pi(x,y)$, also known as *mutual information*, makes the optimization strongly convex and differential [5, 31], allowing for efficient matrix scaling algorithms for approximation, such as Sinkrhon Iterations [95]. In statistical physics, $\mathcal{W}_{c,\varepsilon}(\mu,\nu)$ can be re-factored as a projection problem:

$$\mathcal{W}_{c,\varepsilon}(\mu,\nu) := \min_{\Pi\in\mathbf{\Pi}(\mu,\nu)} \mathrm{KL}\left(\Pi|\mathcal{K}\right), \tag{5.3}$$

where $\mathcal{K}$ is the Gibbs distribution and its density function satisfies $d\mathcal{K}(x,y) = e^{-c(x,y)/\varepsilon}d\mu(x)d\nu(y)$. This problem is often referred to as the "static Schrödinger problem" [57, 88] as it was initially considered in statistical physics. Formally, the Sinkhorn divergence is defined as

$$\overline{\mathcal{W}}_{c,\varepsilon}(\mu,\nu) = 2\mathcal{W}_{c,\varepsilon}(\mu,\nu) - \mathcal{W}_{c,\varepsilon}(\mu,\mu) - \mathcal{W}_{c,\varepsilon}(\nu,\nu), \tag{5.4}$$

which is smooth, positive definite, and metricizes the convergence in law [31]. This definition subtracts two self-distance terms to ensure non-negativity and metric properties.

Properties for Convergence. The contraction analysis of distributional Bellman operator $\mathfrak{T}^\pi$ under a distribution divergence d_p depends on its *scale sensitive* **(S)** and *sum invariant* **(I)** properties [10, 9]. We say d_p is scale sensitive (of order τ) if there exists a $\tau > 0$, such that for all random variables X, Y and a real value $a > 0$, $d_p(aX, aY) \leq |a|^\tau d_p(X, Y)$. d_p has the sum invariant property if whenever a random variable A is independent from X, Y, we have $d_p(A + X, A + Y) \leq d_p(X, Y)$. Based on these properties, [9] shows that $\mathfrak{T}^\pi$ is γ-contractive under the supremal from of Wasserstein distance W_p, which is regarding the first term of $\mathcal{W}_{c,\varepsilon}$ or directly letting $\varepsilon = 0$ in Eq. 5.2. When examining the regularized loss form of $\mathcal{W}_{c,\varepsilon}$, a natural question arises: *What is the influence of the incorporated regularization term on the contraction of $\mathfrak{T}^\pi$?* We begin to address this question in Proposition 2, focusing on the separate regularization term. Here, we define mutual information as $\mathrm{MI}_\Pi(\mu(s,a), \nu(s,a)) = \mathrm{KL}(\Pi|\mu(s,a) \otimes \nu(s,a))$ and its supremal form $\mathrm{MI}^\infty_\Pi(\mu, \nu) = \sup_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathrm{KL}(\Pi|\mu(s,a) \otimes \nu(s,a))$ given a joint distribution Π.

Proposition 11. *$\mathfrak{T}^\pi$ is non-expansive under MI^∞_Π for any non-trivial joint distribution Π.*

Please refer to Appendix 5.8.3 for the proof, where we investigate both **(S)** and **(I)** properties. The non-trivial Π rules out the independence case of μ and ν, where $\mathrm{KL}(\Pi|\mu \otimes \nu)$ would degenerate to zero. Although the non-expansive nature of the introduced regularization term, as shown in Proposition 2, may potentially slow the convergence in Sinkhorn divergence compared with W_p without the regularization, we will demonstrate that $\mathfrak{T}^\pi$ is still contractive under the full Sinkhorn divergence in Theorem 10. Before introducing Theorem 10, we first present the sum-invariant and a new variant of scale-sensitive properties in Proposition 12, which acts as the foundation for Theorem 10.

Proposition 12. *Considering $\mathcal{W}_{c,\varepsilon}$ with the unrectified kernel $k_\alpha := -\|x-y\|^\alpha$ as $-c$ ($\alpha > 0$) and a scaling factor $a \in (0, 1)$, $\mathcal{W}_{c,\varepsilon}$ is sum-invariant **(I)** and satisfies $\mathcal{W}_{c,\varepsilon}(a\mu, a\nu) \leq \Delta_\varepsilon(a, \alpha)\mathcal{W}_{c,\varepsilon}(\mu, \nu)$ **(S)** with a scaling constant $\Delta_\varepsilon(a, \alpha) \in (|a|^\alpha, 1)$ for any μ and ν in a finite set of probability measures.*

Proof Sketch. The detailed proof is provided in Appendix 5.8.4. Let Π^* be the optimal coupling of $\mathcal{W}_{c,\varepsilon}$, we define a ratio $\lambda_\varepsilon(\mu, \nu)$ that satisfies $\lambda_\varepsilon(\mu, \nu) =$

$\frac{\varepsilon \mathrm{KL}(\Pi^*|\mu\otimes\nu)}{\mathcal{W}_{c,\varepsilon}} \in (0,1)$ for a generally non-zero $\mathcal{W}_{c,\varepsilon}$. The ratio $\lambda_\varepsilon(\mu,\nu)$ measures the proportion of the entropic regularization term over the whole loss term $\mathcal{W}_{c,\varepsilon}$. Therefore, the contraction factor $\Delta_\varepsilon(a,\alpha)$ is defined as $\Delta_\varepsilon(a,\alpha) = |a|^\alpha(1-\sup_{\mu,\nu}\lambda_\varepsilon(\mu,\nu)) + \sup_{U,V}\lambda_\varepsilon(\mu,\nu)) \in (|a|^\alpha, 1)$ with $\sup_{\mu,\nu}\lambda_\varepsilon(\mu,\nu) < 1$, which is determined by the scale factor a, the order α, the hyperparameter ε, and the set of interested probability measures.

Contraction Guarantee and Interpolation Relationship. Proposition 12 reveals that $\mathcal{W}_{c,\varepsilon}$ with an unrectified kernel satisfies **(I)** and a variant of **(S)** properties. While the scaling constant $\Delta_\varepsilon(a,\alpha)$ in **(S)** has a complicated form, it remains strictly less than one, even considering a non-expansive nature of the entropic regularization as shown in Proposition 11. We denote the supremal form of Sinkhorn divergence as $\overline{\mathcal{W}}^\infty_{c,\varepsilon}(\mu,\nu) : \overline{\mathcal{W}}^\infty_{c,\varepsilon}(\mu,\nu) = \sup_{(s,a)\in\mathcal{S}\times\mathcal{A}}\overline{\mathcal{W}}_{c,\varepsilon}(\mu(s,a),\nu(s,a))$. In Theorem 10, we will integrate all these properties to demonstrate the contraction property of distributional dynamic programming under $\overline{\mathcal{W}}_{c,\varepsilon}$, specifically highlighting the interpolation property of Sinkhorn divergence between MMD and Wasserstein distance in the context of distributional RL.

Theorem 10. *Considering $\overline{\mathcal{W}}_{c,\varepsilon}(\mu,\nu)$ with an unrectified kernel $k_\alpha := -\|x-y\|^\alpha$ as $-c$ ($\alpha > 0$), where $\mu,\nu \in$ the distribution set of $\{Z^\pi(s,a)\}$ for $s \in \mathcal{S}$, $a \in \mathcal{A}$ in a finite MDP. We define the ratio $\overline{\lambda}_\varepsilon(\mu,\nu)$ as $\overline{\lambda}_\varepsilon(\mu,\nu) = \frac{\varepsilon KL(\Pi^*|\mu\otimes\nu)}{\overline{\mathcal{W}}_{c,\epsilon}(\mu,\nu)} \in (0,1)$ with $\sup_{\mu,\nu}\overline{\lambda}_\varepsilon(\mu,\nu) < 1$. Then, we have:*

1. *($\varepsilon \to 0$) $\overline{\mathcal{W}}_{c,\varepsilon}(\mu,\nu) \to 2W^\alpha_\alpha(\mu,\nu)$. When $\varepsilon = 0$, $\mathfrak{T}^\pi$ is γ^α-contractive under $\overline{\mathcal{W}}^\infty_{c,\varepsilon}$.*
2. *($\varepsilon \to +\infty$) $\overline{\mathcal{W}}_{c,\varepsilon}(\mu,\nu) \to MMD^2_{k_\alpha}(\mu,\nu)$. When $\varepsilon = +\infty$, $\mathfrak{T}^\pi$ is γ^α-contractive under $\overline{\mathcal{W}}^\infty_{c,\varepsilon}$.*
3. *($\varepsilon \in (0,+\infty)$), $\mathfrak{T}^\pi$ is at least $\overline{\Delta}_\varepsilon(\gamma,\alpha)$-**contractive** under $\overline{\mathcal{W}}^\infty_{c,\varepsilon}$, where $\overline{\Delta}_\varepsilon(\gamma,\alpha)$ is an MDP-dependent constant defined as $\overline{\Delta}_\varepsilon(\gamma,\alpha) = \gamma^\alpha(1-\sup_{\mu,\nu}\overline{\lambda}_\varepsilon(\mu,\nu)) + \sup_{\mu,\nu}\overline{\lambda}_\varepsilon(\mu,\nu)) \in (\gamma^\alpha, 1)$.*

Proof Sketch. The detailed proof of Theorem 10 can be found in Appendix 5.8.5. Theorem 10 (1) and (2) are follow-up conclusions in terms of the convergence

behavior of $\mathfrak{T}^\pi$ based on the interpolation relationship between Sinkhorn divergence with Wasserstein distance and MMD [36]. We also provide a rigorous analysis within the context of distributional RL for completeness. Our critical theoretical contribution is the part (3) for the general $\varepsilon \in (0, \infty)$, where we show that $\mathfrak{T}^\pi$ is at least a $\overline{\Delta}_\varepsilon(\gamma, \alpha)$-contractive operator. The contraction factor $\overline{\Delta}_\varepsilon(\gamma, \alpha) \in (\gamma^\alpha, 1)$ depends on the return distribution set $\{Z^\pi(s, a)\}$ of the considered MDP, and it is also a function of γ, ε and α. Due to the influence of the regularization term in Sinkhorn loss, $\overline{\Delta}_\varepsilon(\gamma, \alpha)$ is larger than $|\gamma|^\alpha$, the contraction factor for Wasserstein distance without the regularization. Thus, $\overline{\Delta}_\varepsilon(\gamma, \alpha)$ can be seen as an interpolation between γ^α and 1, with the coefficient $\sup_{\mu,\nu} \overline{\lambda}_\varepsilon(\mu, \nu) \in (0, 1)$ defined in Theorem 10. The ratio $\overline{\lambda}_\varepsilon(\mu, \nu)$ measures the proportion of the KL regularization term relative to $\overline{\mathcal{W}}_{c,\varepsilon}$. As $\varepsilon \to 0$ or $+\infty$, $\sup_{\mu,\nu} \overline{\lambda}_\varepsilon(\mu, \nu) \to 0$, leading to γ^α-contraction. This aligns with parts (1) and (2).

Consistency with Existing Contraction Conclusions. As Sinkhorn divergence interpolates between Wasserstein distance and MMD, its contraction property for $\varepsilon \in [0, \infty]$ also aligns well with the existing distributional RL algorithms when $c = -k_\alpha$. It is worth noting that using Gaussian kernels in the cost function does not yield concise or consistent contraction results like those in Theorem 10 (3). This conclusion is consistent with MMD-DQN [77] ($\varepsilon \to +\infty$), where $\mathfrak{T}^\pi$ is generally not a contraction operator under MMD with Gaussian kernels, as counterexamples exist (Theorem 2) in [77]. Guided by our theoretical results, we employ the rectified kernel k_α as the cost function and set $\alpha = 2$ in our experiments, ensuring that $\mathfrak{T}^\pi$ retains the

Algorithm	d_p **Distribution Divergence**	**Representation** Z_θ	**Convergence Rate of** $\mathfrak{T}^\pi$	**Sample Complexity of** d_p
C51	Cramér distance	Categorical Distribution	$\sqrt{\gamma}$	\
QR-DQN-1	Wasserstein distance	Quantiles	γ	$\mathcal{O}(n^{-\frac{1}{d}})$
MMD-DQN	MMD	Samples	$\gamma^{\alpha/2}$ (k_α)	$\mathcal{O}(n^{-1})$
SinkhornDRL (ours)	Sinkhorn divergence $(c = -k_\alpha)$	Samples	γ $(\varepsilon \to 0)$ $\gamma^{\alpha/2}$ $(\varepsilon \to \infty)$	$\mathcal{O}(n^{\frac{\frac{\varepsilon}{2}}{\varepsilon^{\lceil d/2 \rceil}\sqrt{n}}})$ $(\varepsilon \to 0)$ $\mathcal{O}(n^{-\frac{1}{2}})$ $(\varepsilon \to \infty)$

Table 5.1: Properties of different distribution divergences in typical distributional RL algorithms. d is the sample dimension and $\kappa = 2\beta d + \|c\|_\infty$, where the cost function c is β-Lipschitz [35]. Sample complexity is improved to $\mathcal{O}(1/n)$ using the kernel herding technique [17] in MMD.

contraction property guaranteed by Theorem 10 (3). In Table 5.1, we also summarize the main properties of distribution divergences in typical distributional RL algorithms, including the convergence rate of $\mathfrak{T}^\pi$ and sample complexity, i.e., the convergence rate of a given metric between a measure and its empirical counterpart as a function of the number of samples n.

5.5.2 Extension to Multi-dimensional Return Distributions

As the ability to extend to the multi-dimensional reward setting is one of the major advantages of SinkhornDRL over quantile regression-based algorithms, we next demonstrate that the joint distributional Bellman operator in the multi-dimensional reward case is contractive under Sinkhorn divergence $\overline{\mathcal{W}}^\infty_{c,\varepsilon}$. First, we define a d-dimensional reward function as $\mathbf{R} : \mathcal{S} \times \mathcal{A} \to P(\mathbb{R}^d)$, where d represents the number of reward sources. Consequently, we have joint return distributions of the d-dimensional return vector $\mathbf{Z}^\pi(s,a) = \sum_{t=0}^\infty \mathbf{R}(s_t, a_t)$, where $\mathbf{Z}^\pi(s,a) = (Z_1^\pi(s,a), \cdots, Z_d^\pi(s,a))^\top$. The joint distributional Bellman operator $\mathfrak{T}_d^\pi$ applied on the joint distribution of the random vector $\mathbf{Z}(s,a)$ is defined as $\mathfrak{T}_d^\pi \mathbf{Z}(s,a) :\overset{D}{=} \mathbf{R}(s,a) + \gamma \mathbf{Z}(s',a')$, where $s' \sim P(\cdot|s,a)$, $a' \sim \pi(\cdot|s')$.

Corollary 2. *For two joint distributions $\mathbf{Z}_1$ and $\mathbf{Z}_2$, $\mathfrak{T}_d^\pi$ is $\overline{\Delta}_\varepsilon(\gamma,\alpha)$-contractive under $\overline{\mathcal{W}}^\infty_{c,\varepsilon}$, i.e.,*

$$\overline{\mathcal{W}}^\infty_{c,\varepsilon}(\mathfrak{T}^\pi \mathbf{Z}_1, \mathfrak{T}^\pi \mathbf{Z}_2) \leq \overline{\Delta}_\varepsilon(\gamma,\alpha) \overline{\mathcal{W}}^\infty_{c,\varepsilon}(\mathbf{Z}_1, \mathbf{Z}_2). \tag{5.5}$$

Please refer to Appendix 5.8.6 for the proof. The contraction guarantee of Sinkhorn divergence enables us to effectively deploy our SinkhornDRL algorithm in various RL tasks that involve multiple sources of rewards [64, 23], hybrid reward architecture [104, 62], or sub-reward structures after reward decomposition [63, 117]. We compare SinkhornDRL with MMD-DQN in multiple reward sources setting in Section 5.6.3, where SinkhornDRL significantly outperforms MMD-DQN by leveraging its ability to capture richer data geometry, a key advantage of optimal transport distances.

5.5.3 SinkhornDRL Algorithm and Approximation

Equipping Sinkhorn Divergence and Particle Representation. The key to applying Sinkhorn divergence in distributional RL is to leverage the Sinkhorn loss $\overline{\mathcal{W}}_{c,\varepsilon}$ to measure the distance between the current action-return distribution $Z_\theta(s,a)$ and the target distribution $\mathfrak{T}^\pi Z_\theta(s,a)$. For each s,a pair, this yields $\overline{\mathcal{W}}_{c,\varepsilon}(Z_\theta(s,a), \mathfrak{T}^\pi Z_\theta(s,a))$. For the representation of $Z_\theta(s,a)$, we employ the unrestricted statistics, i.e., deterministic samples, akin to MMD-DQN, instead of predefined statistic functionals like quantiles in QR-DQN or categorical distributions in C51. More concretely, we use neural networks to generate samples to approximate the return distributions, expressed as $Z_\theta(s,a) := \{Z_\theta(s,a)_i\}_{i=1}^N$, where N is the number of generated samples. We refer to these samples $\{Z_\theta(s,a)_i\}_{i=1}^N$ as *particles*. We then use the Dirac mixture $\frac{1}{N}\sum_{i=1}^N \delta_{Z_\theta(s,a)_i}$ to approximate the true density function of $Z^\pi(s,a)$, thus minimizing the Sinkhorn divergence between the approximate distribution and its distributional Bellman target. A generic Sinkhorn distributional RL algorithm with particle representation is provided in Algorithm 2.

Efficient Approximation via Sinkhorn Iterations with Guarantee. By introducing an entropy regularization, Sinkhorn divergence renders optimal transport computationally feasible, especially in the high-dimensional space, via efficient algorithms, e.g., Sinkhorn Iterations [95, 36]. Notably, Sinkhorn iteration with L steps yields a differentiable and solvable efficient loss function as the main burden is the matrix-vector multiplication, which streams well on

Require: Number of generated samples N, the cost function c, hyperparameter ε and the target network Z_{θ^*}.
Input: Sample transition (s, a, r', s')
1: **Policy evaluation**: $a^* \sim \pi(\cdot|s')$
2: **Control**: $a^* \leftarrow \arg\max_{a' \in \mathcal{A}} \frac{1}{N}\sum_{i=1}^N Z_\theta\left(s', a'\right)_i$
3: TD update: $\mathfrak{T}Z_i \leftarrow r + \gamma Z_{\theta^*}\left(s', a^*\right)_i, \forall 1 \le i \le N$
Output: $\overline{\mathcal{W}}_{c,\varepsilon}\left(\{Z_\theta(s,a)_i\}_{i=1}^N, \{\mathfrak{T}Z_j\}_{j=1}^N\right)$

Algorithm 2: Generic Sinkhorn distributional RL Update

the GPU by simply adding extra differentiable layers on the typical deep neural network, such as a DQN architecture. *It has been proven that Sinkhorn iterations asymptotically converge to the true loss in a linear rate* [36, 32, 20, 53]. We provide a detailed description of Sinkhorn iterations in Algorithm 3 and a full version in Algorithm 4 of Appendix 5.8.7. In practice, selecting proper values of L and ε is crucial. To this end, we conduct a rigorous sensitivity analysis, detailed in Section 5.6.

Remark: Relationship with IQN and FQF. In the realm of distributional RL algorithms, it is important to highlight that QR-DQN and MMD-DQN are direct counterparts to SinkhornDRL within the first dimension of algorithmic evolution. In contrast, IQN and FQF enhance QR-DQN and position them in the second modeling dimension, which are orthogonal to our work. As discussed in [77], the techniques from IQN and FQF can naturally extend both MMD-DQN and SinkhornDRL. For instance, we can implicitly generate $\{Z_\theta(s,a)_i\}_{i=1}^N$ by applying a neural network to N samples of a base sampling distribution, as in IQN. We can also use a proposal network to learn the weights of each generated sample as in FQF. We leave these modeling extensions as future works and our current study focuses on rigorously investigating the simplest modeling choice via Sinkhorn divergence.

5.6 Experiments

We substantiate the effectiveness of SinkhornDRL as described in Algorithm 2 on the entire 55 Atari 2600 games. Without increasing the model capacity for a fair comparison, we leverage the same architecture as QR-DQN and MMD-DQN, and replace the quantiles output in QR-DQN with N particles (samples). In contrast to MMD-DQN, SinkhornDRL only changes the distribution divergence from MMD to Sinkhorn divergence. As such, the potential performance improvement of our algorithm is directly attributed to the theoretical advantages of Sinkhorn divergence over MMD.

Baseline Implementation. We choose DQN [74] and three typical distributional RL algorithms as classic baselines, including C51 [9], QR-DQN [22] and MMD-DQN [77]. For a fair comparison, we build SinkhornDRL and all baselines based on a well-accepted PyTorch implementation[1] of distributional RL algorithms. We re-implement MMD-DQN based on its original TensorFlow implementation[2], and keep the same setting. For example, our MMD-DQN still employs Gaussian kernels $k_h(x, y) = \exp(-(x - y)^2/h)$ with the same kernel mixture trick covering a range of bandwidths h as adopted in MMD-DQN [77].

SinkhornDRL Implementation and Hyperparameter Settings. For a fair comparison with QR-DQN, C51, and MMD-DQN, we use the same hyperparameters: the number of generated samples $N = 200$, Adam optimizer with $\text{lr} = 0.00005, \epsilon_{\text{Adam}} = 0.01/32$. In SinkhornDRL, we choose the number of Sinkhorn iterations $L = 10$ and smoothing hyperparameter $\varepsilon = 10.0$ in Section 5.6.1 after conducting sensitivity analysis in Section 5.6.2. Guided by the contraction guarantee analyzed in Theorem 10, we use *the unrectified kernel*, specifically setting $-c = k_\alpha$ and choosing $\alpha = 2$. This choice ensures *our implementation is consistent with the theoretical results regarding the contraction guarantee in Theorem 10 (3).* We evaluate all algorithms on 55 Atari games, averaging results over three seeds. The shade in the learning curves of each game represents the standard deviation.

5.6.1 Performance of SinkhornDRL

Learning Curves of Human Normalized Scores (HNS). We compare the learning curves of the Mean, Median, and Interquartile Mean (IQM) [2] across all considered distributional RL algorithms in Figure 5.1 summarized over 55 Atari games. The IQM (x%) computes the mean from the x% to $(1 - x)$% range of HNS, providing a robust alternative to the Mean that mitigates the impact of extremely high scores on specific games and is more statistically efficient than the Median. For computational feasibility, we evaluate

[1] `https://github.com/ShangtongZhang/DeepRL`
[2] `https://github.com/thanhnguyentang/mmdrl`

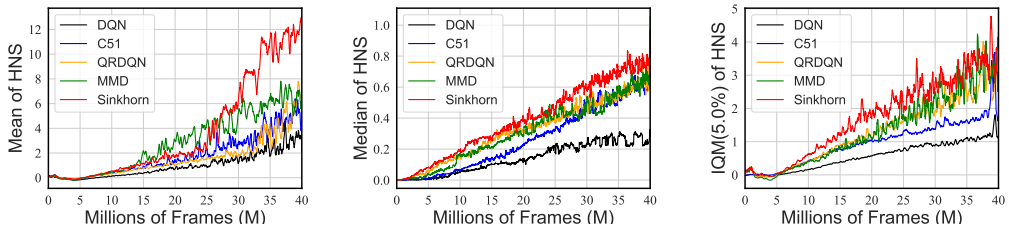
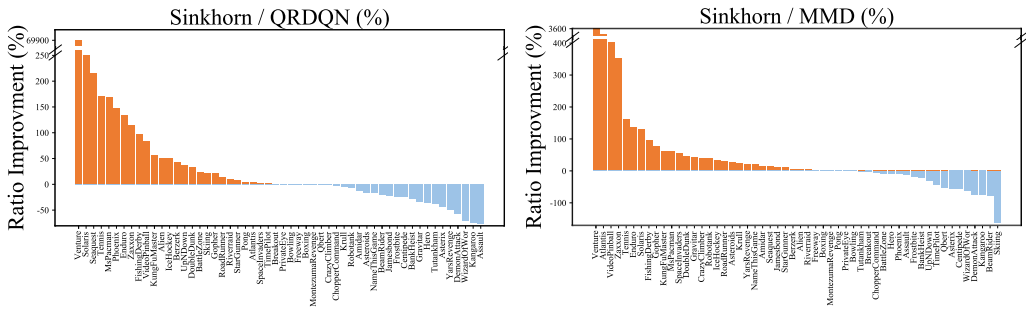


Figure 5.1: Mean (left), Median (middle), and IQM (5%) (right) of Human-Normalized Scores (HNS) summarized over 55 Atari games. We run 3 seeds for each algorithm.

the algorithms over 40M training frames. Our findings reveal that Sinkhorn-DRL achieves state-of-the-art performance in terms of mean, median, and IQM (5%) of HNS across most training phases. Notably, SinkhornDRL exhibits slower convergence during the early training phase, as indicated by the Mean of HNS (left panel of Figure 5.1). This slower initial convergence can be explained by the slower contraction factor $\overline{\Delta}_\varepsilon(\gamma, \alpha) > \gamma^\alpha$ in Theorem 10, as opposed to MMD-DQN. To ensure the reliability of our results, we also provide the learning curves for each Atari game in Figures 5.6 and 5.7 in Appendix 5.8.8. Furthermore, a table summarizing all raw scores is available in Table 5.2 in Appendix 5.8.9. This table highlights that SinkhornDRL achieves the highest numbers of best and second-best performance of all games among all baseline algorithms. Overall, we conclude that SinkhornDRL generally outperforms existing distributional RL algorithms.

Ratio Improvement Analysis across All Games. Given the interpolation nature of Sinkhorn divergence between Wasserstein distance and MMD, as analyzed in Theorem 10, a pertinent question arises: *In which environments does SinkhornDRL potentially perform better?* We empirically address this question by conducting a ratio improvement comparison between Sinkhorn-DRL and both QR-DQN and MMD-DQN across all games. Figure 5.2 showcases that SinkhornDRL surpasses both QR-DQN and MMD-DQN in more than half of the games and significantly excels at them in a large proportion of games. Notably, *the games where SinkhornDRL achieves considerable improvement tend to have larger action spaces and more complex dynamics.* In



(a) SinkhornDRL vs QR-DQN **(b)** SinkhornDRL vs MMD-DQN

Figure 5.2: Ratio improvement of return for SinkhornDRL over QR-DQN (left) and MMD-DQN (right) averaged over 3 seeds. The ratio improvement is calculated by (SinkhornDRL - QR-DQN) / QR-DQN in (a) and (SinkhornDRL - MMD-DQN) / MMD-DQN in (b), respectively.

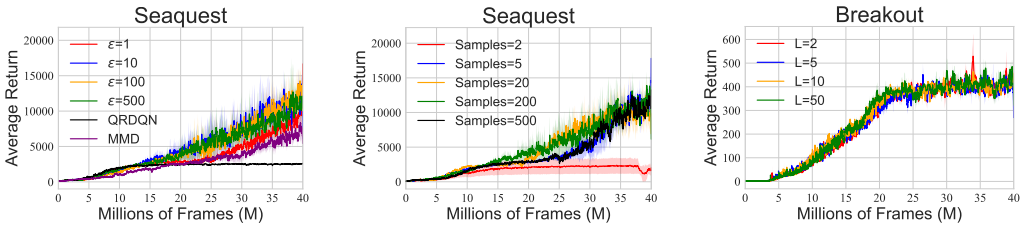
particular, as illustrated in Figure 5.2, these games include Venture, Seaquest, Solaris, Tennis, Phoenix, Atlantis, and Zaxxon. Most of these games have an 18-dimensional action space and intricate dynamics, except for Atlantis, which has a 4-dimensional action space and simpler dynamics where MMD-DQN is substantially inferior to SinkhornDRL. For a detailed comparison, we provide the features of all games, including the number of action spaces, and complexity of environment dynamics in Table 5.3 of Appendix 5.8.10.

In summary, compared with QR-DQN, the empirical success of SinkhornDRL can be attributed to several key factors: 1. *Enhanced return distribution representation:* SinkhornDRL captures return distribution characteristics more accurately by directly using samples, avoiding the non-crossing issue of learned quantile curves or the potential limitations of quantile representation. *2. Smooth transport plan and stable convergence.* The induced smoother transport plan (see Appendix 5.8.1 for visualization) and the inherent smoothness of Sinkhhorn divergence contribute to more stable convergence, leading to performance improvement. In contrast to MMD-DQN, the benefits of SinkhornDRL arise from its richer data representation capability when comparing return distributions, rooted in the OT nature. This is in comparison to the potentially restricted kernel-specific distances, such as MMD.

5.6.2 Sensitivity Analysis and Computational Cost

Sensitivity Analysis. In practice, a proper ε is preferable as an overly large or small ε will lead to numerical instability of Sinkhorn iterations in Algorithm 3 (see the discussion in Section 4.4 of [81]), therefore worsening its performance, as shown in Figure 5.3 (a). This implies that the potential interpolation nature of limiting behaviors between SinkhornDRL with QR-DQN and MMD-DQN revealed in Theorem 10 may not be able to be rigorously verified in numerical experiments. SinkhornDRL also requires a proper number of iterations L and samples N. For example, a small N, e.g., $N = 2$ in Seaquest in Figure 5.3 (b) leads to the divergence of algorithms, while an overly large N can degrade the performance and meanwhile increases the computational burden (Appendix 5.8.11). We conjecture that using larger networks to represent more samples is more likely to suffer from overfitting, yielding the instability in the RL training [11]. Therefore, we choose $N = 200$ to attain favorable performance and guarantee computational effectiveness simultaneously. We provide a more detailed sensitivity analysis and more results on StarGunner and Zaxxon in Appendix 5.8.11.

Computation Cost. In terms of the computation cost, SinkhornDRL slightly increases the computational overhead compared with C51, QR-DQN, and MMD-DQN. For instance, SinkhornDRL increases the average computational cost compared with MMD-DQN by around 20%. Due to the space limit, we provide more computation cost comparison in terms of L and N in Appendix 5.8.11.



(a) Hyper-parameter ε **(b)** Number of Samples **(c)** Sinkhorn Iterations L

Figure 5.3: Sensitivity analysis of SinkhornDRL on Breakout and Seaquest in terms of ε, number of samples, and number of iteration L. Learning curves are reported over three seeds.

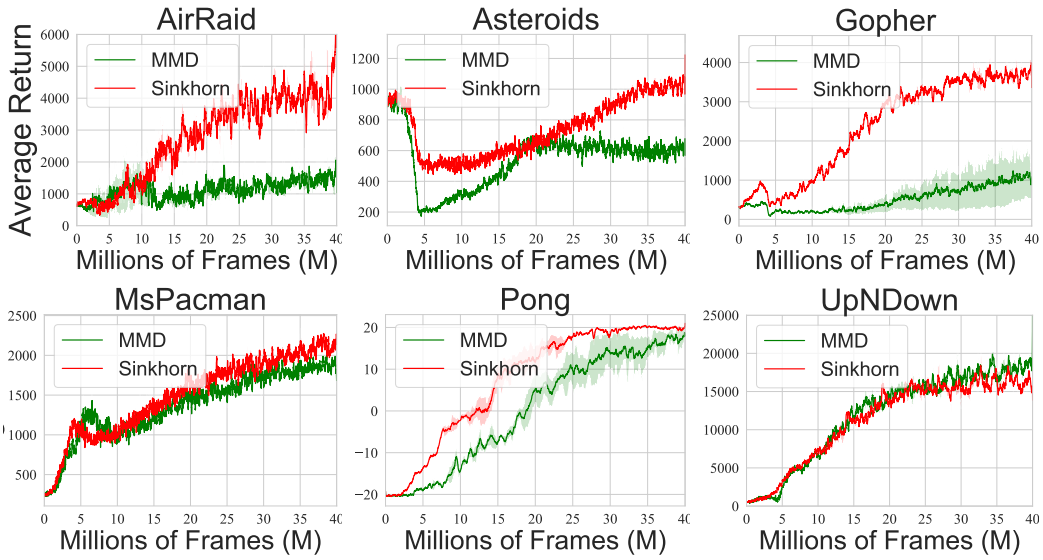


Figure 5.4: Performance of SinkhornDRL on six Atari games with multi-dimensional reward functions.

5.6.3 Modeling Joint Return Distribution for Multi-Dimensional Reward Functions

Many RL tasks involve modeling multivariate return distributions. Following the multi-dimensional reward setting in [117], we compare SinkhornDRL with MMD-DQN on six Atari games with multiple sources of rewards. In these tasks, the primitive scalar-based rewards are decomposed into reward vectors based on the respective reward structures (see Appendix 5.8.12 for more details). Figure 5.4 showcases that SinkhornDRL outperforms MMD-DQN in most cases for multi-dimensional reward functions. Of particular note, it remains an open question to directly approximate multi-dimensional Wasserstein distances via quantile regression or other efficient algorithms in RL tasks.

5.7 Conclusion, Limitations and Future Work

In this work, we propose a novel family of distributional RL algorithms based on Sinkhorn divergence that accomplishes competitive performance compared with the typical distributional RL algorithms on the Atari games suite. The-

oretical results about the properties of this regularized Wasserstein loss and its convergence guarantee in the context of RL are provided with rigorous empirical verification.

Limitations. While SinkhornDRL achieves competitive performance, it relatively increases the computational cost and requires tuning additional hyperparameters. This hints that the enhanced performance offered by SinkhornDRL may come with slightly greater efforts in practical deployment. Additionally, it remains elusive for a deeper connection between the theoretical properties of divergences and the practical performance of distributional RL algorithms given a specific environment.

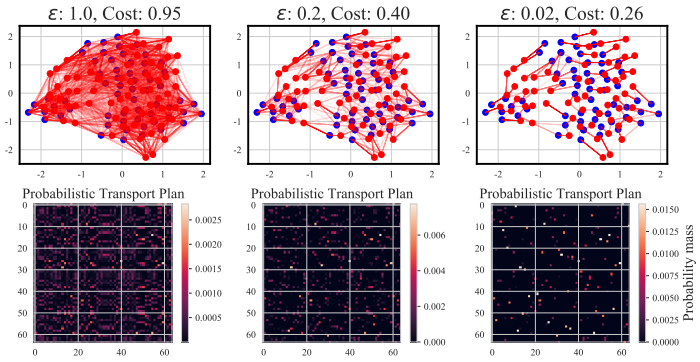
Future work. Along the two dimensions of distributional RL algorithm evolution, we can further improve Sinkhorn distributional RL by incorporating implicit generative models, including parameterizing the cost function and increasing model capacity. Moreover, Sinkhorn distributional RL also opens a door for new applications of Sinkhorn divergence and more optimal transport approaches in RL. It also becomes increasingly crucial to design a quantitative criterion for a given environment to recommend the choice of a specific distribution divergence before conducting costly experiments.

5.8 Appendix

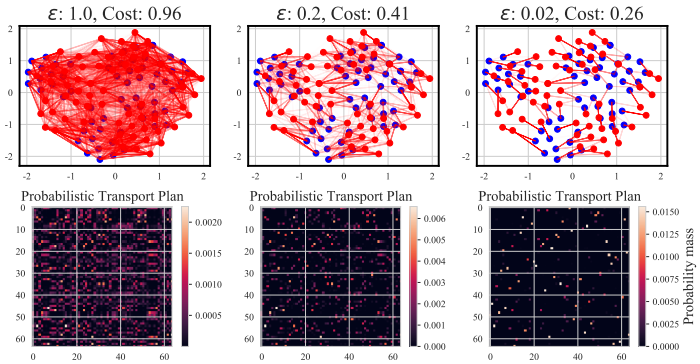
5.8.1 Smoother Transport Plan via Sinkhorn Divergence

We visualize the optimal transport plans by solving Sinkhorn divergence with different ε in well-trained SinkhornDRL models across three games in Figure 5.5 We evaluate (randomly selected 64) current and target state features to be compared and then apply t-SNE to reduce their dimension from 512 to 2 associated with a normalization for visualization. In each game of Figure 5.5, as we increase the regularization strength ε (from right to left), the resulting transport plans tend to be smoother, less concentrated, and more uniformly distributed by transporting the point mass between two distributions (in red

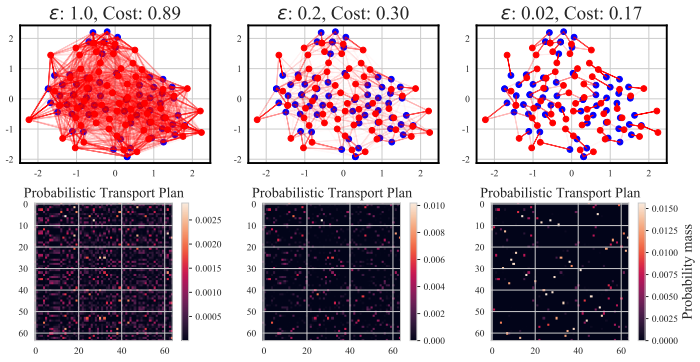
and blue).



(a) Enduro



(b) Qbert



(c) Seaquest

Figure 5.5: Optimal transport plans for via Sinkhorn Iterations in SinkhornDRL on three Atari games. The first row denotes the (two-dimensional) spatial transport plans across different data points, while the second row represents the heat map of the obtained transport plan (optimal coupling).

5.8.2 Definition of Distribution Divergences and Contraction Properties

Definition of distances. Given two random variables X and Y, one-dimensional p-Wasserstein metric W_p between the distributions of X and Y has a simplified form via the quantile functions:

$$W_p(X,Y) = \left(\int_0^1 \left| F_X^{-1}(\omega) - F_Y^{-1}(\omega) \right|^p d\omega \right)^{1/p} = \| F_X^{-1} - F_Y^{-1} \|_p, \tag{5.6}$$

which F^{-1} is the quantile function, also known as inverse cumulative distribution function, of a random variable with the cumulative distribution function as F. The supremal form of W_p, denoted by W_p^∞, is defined as

$$W_p^\infty(\mu,\nu) = \sup_{(s,a)\in\mathcal{S}\times\mathcal{A}} W_p^\infty(\mu(s,a),\nu(s,a)). \tag{5.7}$$

Further, ℓ_p distance [27] is defined as

$$\ell_p(X,Y) := \left(\int_{-\infty}^{\infty} |F_X(\omega) - F_Y(\omega)|^p \, \mathrm{d}\omega \right)^{1/p} = \| F_X - F_Y \|_p. \tag{5.8}$$

The ℓ_p distance and Wasserstein metric are identical at $p = 1$, but are otherwise distinct. Note that when $p = 2$, ℓ_p distance is also called Cramér distance [10] $d_C(X,Y)$. Also, Cramér distance has a different representation given by

$$d_C(X,Y) = \mathbb{E}|X - Y| - \frac{1}{2}\mathbb{E}\left|X - X'\right| - \frac{1}{2}\mathbb{E}\left|Y - Y'\right|, \tag{5.9}$$

where X' and Y' are the i.i.d. copies of X and Y. Energy distance [103, 121] is a natural extension of Cramér distance to the multivariate case, defined by

$$d_E(\mathbf{X},\mathbf{Y}) = \mathbb{E}\|\mathbf{X} - \mathbf{Y}\| - \frac{1}{2}\mathbb{E}\|\mathbf{X} - \mathbf{X}'\| - \frac{1}{2}\mathbb{E}\|\mathbf{Y} - \mathbf{Y}'\|, \tag{5.10}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are multivariate. Moreover, the energy distance is a special case of the maximum mean discrepancy (MMD), which is formulated as

$$\mathrm{MMD}(\mathbf{X}, \mathbf{Y}; k) = \left(\mathbb{E}\left[k\left(\mathbf{X}, \mathbf{X}'\right)\right] + \mathbb{E}\left[k\left(\mathbf{Y}, \mathbf{Y}'\right)\right] - 2\mathbb{E}[k(\mathbf{X}, \mathbf{Y})]\right)^{1/2}, \quad (5.11)$$

where $k(\cdot, \cdot)$ is a continuous kernel on $\mathcal{X}$. In particular, if k is a trivial kernel, also called the unrectified kernel, MMD degenerates to energy distance. Additionally, we further define the supreme MMD, which is a functional $\mathcal{P}(\mathcal{X})^{\mathcal{S}\times\mathcal{A}} \times \mathcal{P}(\mathcal{X})^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R}$ formulated as

$$\mathrm{MMD}_\infty(\mu, \nu) = \sup_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathrm{MMD}_\infty(\mu(s,a), \nu(s,a)). \quad (5.12)$$

We further summarize the convergence rates of the distributional Bellman operator $\mathfrak{T}^\pi$ under different distribution divergences.

- $\mathfrak{T}^\pi$ is γ-contractive under the supreme form of Wassertein distance W_p.
- $\mathfrak{T}^\pi$ is $\gamma^{1/p}$-contractive under the supreme form of ℓ_p distance.
- $\mathfrak{T}^\pi$ is $\gamma^{\alpha/2}$-contractive under MMD_∞ with $k_\alpha(x, y) = -\|x - y\|^\alpha$.

Proof of Contraction in Distributional Dynamic Programming.

- Contraction under the supreme form of W_p is provided in Lemma 3 [9].
- Contraction under supreme form of ℓ_p distance refers to Theorem 3.4 [27].
- Contraction under MMD_∞ is provided in Lemma 6 [77].

5.8.3 Proof of Proposition 2

Proof. We denote two marginal random variables U and V with the pdf $\mu(x)$ and $\nu(y)$. We next denote the $p_\Pi(x, y)$ as the pdf for Π in $\mathrm{MI}_\Pi(U, V) = \mathrm{KL}(\Pi|U \otimes V)$. We first prove that the $\mathrm{MI}_\Pi(U, V)$ is sum-invariant, which is based on the dual form of KL divergence via the variational representation [25, 3]:

$$D_{KL}(X, Y) = \sup_{f\in\mathcal{L}^b} \{\mathbb{E}_X[f(x)] - \log\left(\mathbb{E}_Y\left[e^{f(y)}\right]\right)\}, \quad (5.13)$$

where $\mathcal{L}^b$ is the space of bounded measurable functions. The mutual information involves two-dimensional random variables in the KL divergence. Let $U' = a + U$ and $V = a + V$ with pdf μ' and ν', we denote the joint distribution with margins $\mu'(x) = \mu(x-a)$ and $\nu'(y) = \nu(y-a)$ as $\Pi'(x,y)$ whose pdf $p_{\Pi'}$ satisfies $p_{\Pi'}(x,y) = p_\Pi(x-a, y-a)$. Based on the two-dimensional variational representation of KL divergence $\mathrm{MI}_\Pi(U,V) = \sup_{f\in\mathcal{L}^b}\{\mathbb{E}_\Pi[f(x,y)] - \log\left(\mathbb{E}_{U,V}\left[e^{f(x,y)}\right]\right)\}$, we have:

$$
\begin{aligned}
&\mathrm{MI}_\Pi(A+U, A+V) \\
&= \sup_{f\in\mathcal{L}^b}\{\mathbb{E}_{\Pi'}[f(x,y)] - \log\left(\mathbb{E}_{A+U,A+V}\left[e^{f(x,y)}\right]\right)\} \\
&\overset{(a)}{=} \sup_{f\in\mathcal{L}^b}\{\mathbb{E}_A\left[\mathbb{E}_{\Pi(x-a,y-a)}\left[f(x,y)\right]\right] - \log\left(\mathbb{E}_A\left[\mathbb{E}_{a+U,a+V}\left[e^{f(x,y)}\right]\right]\right)\} \\
&= \sup_{f\in\mathcal{L}^b}\{\mathbb{E}_A\left[\mathbb{E}_{\Pi(x,y)}\left[f(x+a,y+a)\right]\right] - \log\left(\mathbb{E}_A\left[\mathbb{E}_{U,V}\left[e^{f(x+a,y+a)}\right]\right]\right)\} \\
&\overset{(b)}{\leq} \sup_{f\in\mathcal{L}^b}\{\mathbb{E}_A\mathbb{E}_\Pi[f(x+a,y+a)] - \mathbb{E}_A\log\left(\mathbb{E}_{U,V}\left[e^{f(x+a,y+a)}\right]\right)\} \\
&= \sup_{f\in\mathcal{L}^b}\{\mathbb{E}_A[\mathbb{E}_\Pi[f(x+a,y+a)] - \log\left(\mathbb{E}_{U,V}\left[e^{f(x+a,y+a)}\right]\right)]\} \\
&\overset{(c)}{\leq} \mathbb{E}_A\sup_{f\in\mathcal{L}^b}\{\mathbb{E}_\Pi[f(x+a,y+a)] - \log\left(\mathbb{E}_{U,V}\left[e^{f(x+a,y+a)}\right]\right)\} \\
&\overset{(d)}{=} \mathbb{E}_A\sup_{g\in\mathcal{L}^b}\{\mathbb{E}_\Pi[g(x,y)] - \log\left(\mathbb{E}_{U,V}\left[e^{g(x,y)}\right]\right)\} \\
&= \mathrm{MI}_\Pi(U,V),
\end{aligned}
\tag{5.14}
$$

where (a) is by the independence of A between X, Y, and the joint cdf Π. For instance, in the one-dimensional setting, we have $\mathbb{E}_{Z=A+X}\left[f(z)\right] = \int_a\int_x f(x+a)p_A(a)p_X(x)dxda = \mathbb{E}_A\left[\mathbb{E}_X\left[f(x+a)\right]\right]$. (b) and (c) are by Jensen's inequality in terms of the convex function $-\log(x)$ and $\sup_f$, and (d) is because the translated cdf is still within $\mathcal{L}^b$.

Next, we show that MI_Π is NOT scale-sensitive or with the zero-order τ. This result is directly based on the similar property of KL divergence. With a slight abuse of notations, we denote $U' = aU$ and $V' = aV$, whose pdfs are $\mu'(x) = \frac{1}{a}\mu(\frac{x}{a})$ and $\nu'(y) = \frac{1}{a}\nu(\frac{y}{a})$, respectively. The scaled joint distribution Π' with the pdf $p_{\Pi'}$ satisfying $p_{\Pi'}(x,y) = \frac{1}{a^2}p_\Pi(x/a, y/a)$. Therefore, its marginal distributions are $\int_y \frac{1}{a^2}p_\Pi(x/a,y/a)dy = \frac{1}{a}\mu(\frac{x}{a})$ and $\int_x \frac{1}{a^2}p_\Pi(x/a,y/a)dy = \frac{1}{a}\nu(\frac{y}{a})$.

We thus have the following result:

$$\begin{aligned}
\mathrm{MI}_\Pi(aU, aV) &= \mathrm{KL}(\Pi'(x,y)|U' \otimes V') \\
&= \int p_{\Pi'}(x,y) \log \frac{p_{\Pi'}(x,y)}{\mu'(x)\nu'(y)} dxdy \\
&= \int \frac{1}{a^2} p_\Pi(x/a, y/a) \log \frac{\frac{1}{a^2} p_\Pi(x/a, y/a)}{\frac{1}{a^2}\mu(x/a)\nu(y/a)} dxdy \\
&= \int p_\Pi(x,y) \log \frac{p_\Pi(x,y)}{\mu(x)\nu(y)} dxdy \\
&= \mathrm{MI}_\Pi(U, V).
\end{aligned} \tag{5.15}$$

Putting the two properties together and given two return distributions $Z_1(s,a)$ and $Z_2(s,a)$, we have the non-expansive contraction property of the supremal form of MI_Π as follows.

$$\begin{aligned}
\mathrm{MI}^\infty_\Pi(\mathfrak{T}^\pi Z_1, \mathfrak{T}^\pi Z_2) &= \sup_{s,a} \mathrm{MI}_\Pi(\mathfrak{T}^\pi Z_1(s,a), \mathfrak{T}^\pi Z_2(s,a)) \\
&= \sup_{s,a} \mathrm{MI}_\Pi(R(s,a) + \gamma Z_1(s',a'), R(s,a) + \gamma Z_2(s',a')) \\
&\overset{(a)}{\leq} \mathrm{MI}_\Pi(\gamma Z_1(s',a'), \gamma Z_2(s',a')) \\
&\overset{(b)}{=} \mathrm{MI}_\Pi(Z_1(s',a'), Z_2(s',a')) \\
&\leq \sup_{s,a} \mathrm{MI}_\Pi(Z_1(s',a'), Z_2(s',a')) \\
&= \mathrm{MI}^\infty_\Pi(Z_1, Z_2),
\end{aligned} \tag{5.16}$$

where (a) relies on the sum invariant property of MI_Π and (b) utilizes the non-scale sensitive property of MI_Π. By applying the well-known Banach fixed point theorem, we have a unique return distribution when convergence of distributional dynamic programming under MI_Π for any non-trivial joint distribution Π.

□

5.8.4 Proof of Proposition 12

Sum Invariant Property Given two random variables U and V with the marginal distributions as μ and ν, and a random variable A that is independent

of them, we aim at proving

$$\mathcal{W}_{c,\varepsilon}(A+U, A+V) \leq \mathcal{W}_{c,\varepsilon}(U, V). \tag{5.17}$$

According to [81], we have the dual form of $\mathcal{W}_{c,\varepsilon}$:

$$\begin{aligned}
&\mathcal{W}_{c,\varepsilon}(U, V) \\
&= \sup_{\varphi,\psi} \left\{ \int_x \varphi(x)\mu_x dx + \int_y \psi(y)\nu_y dy - \varepsilon \int_{x,y} \exp \frac{\varphi(x)+\psi(y)-c(x,y)}{\varepsilon} \mu_x \nu_y dx dy \right\} \\
&= \sup_{\varphi,\psi} \left\{ \mathbb{E}_\mu \left[\varphi(x)\right] + \mathbb{E}_\nu \left[\psi(y)\right] - \varepsilon \mathbb{E}_{\mu,\nu} \left[\exp \frac{\varphi(x)+\psi(y)-c(x,y)}{\varepsilon} \right] \right\}
\end{aligned} \tag{5.18}$$

Therefore, we have:

$$\begin{aligned}
&\mathcal{W}_{c,\varepsilon}(A+U, A+V) \\
&= \sup_{\varphi,\psi} \left\{ \mathbb{E}_{A+U} \left[\varphi(x)\right] + \mathbb{E}_{A+V} \left[\psi(y)\right] - \varepsilon \mathbb{E}_{A+U,A+V} \left[\exp \frac{\varphi(x)+\psi(y)-c(x,y)}{\varepsilon} \right] \right\} \\
&\overset{(a)}{=} \sup_{\varphi,\psi} \left\{ \mathbb{E}_A \left[\mathbb{E}_\mu \left[\varphi(x+a)\right] + \mathbb{E}_\nu \left[\psi(y+a)\right] - \varepsilon \mathbb{E}_{\mu,\nu} \left[\exp \frac{\varphi(x+a)+\psi(y+a)-c(x,y)}{\varepsilon} \right] \right] \right\} \\
&\overset{(b)}{\leq} \mathbb{E}_A \left[\sup_{\varphi,\psi} \left\{ \mathbb{E}_\mu \left[\varphi(x+a)\right] + \mathbb{E}_\nu \left[\psi(y+a)\right] - \varepsilon \mathbb{E}_{\mu,\nu} \left[\exp \frac{\varphi(x+a)+\psi(y+a)-c(x,y)}{\varepsilon} \right] \right\} \right] \\
&\overset{(c)}{=} \sup_{f,g} \left\{ \mathbb{E}_\mu \left[f(x)\right] + \mathbb{E}_\nu \left[g(y)\right] - \varepsilon \mathbb{E}_{\mu,\nu} \left[\exp \frac{f(x)+g(y)-c(x,y)}{\varepsilon} \right] \right\} \\
&= \mathcal{W}_{c,\varepsilon}(U, V),
\end{aligned} \tag{5.19}$$

where (a) relies on the same techniques used in the proof of Eq. 5.14 in Appendix 5.8.3, (b) utilizes the Jensen inequality of sup, and (c) is based on the fact that the translation operator is still within the same functional space of φ, ψ.

A Variant of Scale Sensitive Property when $c = -k_\alpha$. Let Π^* be the optimal coupling for $\mathcal{W}_{c,\varepsilon}$, we define a ratio $\lambda_\varepsilon(U, V) = \frac{\varepsilon \mathrm{KL}(\Pi^* | \mu \otimes \nu)}{\mathcal{W}_{c,\varepsilon}} \in (0, 1)$ for any considered U, V with measures μ, ν to compare, where the denominator

$\mathcal{W}_{c,\varepsilon}$ is generally non-zero. We thus have the following result:

$$\mathcal{W}_{c,\varepsilon}(aU, aV) \leq \Delta_\varepsilon(a, \alpha)\mathcal{W}_{c,\varepsilon}(U, V), \tag{5.20}$$

where the scaling factor $\Delta_\varepsilon(a, \alpha)$ is defined as

$$\Delta_\varepsilon(a, \alpha) = |a|^\alpha(1 - \sup_{U,V} \lambda_\varepsilon(U, V)) + \sup_{U,V} \lambda_\varepsilon(U, V)) \in (|a|^\alpha, 1)$$

with $\sup_{U,V} \lambda_\varepsilon(U, V) > 0$. *The ratio* $\lambda_\varepsilon(U, V)$ *measures the proportion of the entropic regularization term over the whole divergence term* $\mathcal{W}_{c,\varepsilon}$, i.e., $\lambda_\varepsilon(U, V) = \frac{\varepsilon \mathrm{KL}(\Pi^*|\mu\otimes\nu)}{\mathcal{W}_{c,\varepsilon}} \in (0, 1)$. Under the mild assumption of a finite set of probability measures, we have $\sup_{U,V} \lambda_\varepsilon(U, V) > 0$. To elaborate the reason behind it, we first know that $\lambda_\varepsilon(U, V) < 1$ for any U and V with their measures on the probability measure set. If this set is finite, the ratio set that contains all $\{\lambda_\varepsilon(U, V)\}$ is also finite. Based on the fact that the real set is dense, we can directly find a positive lower bound λ^* for the ratio set, such that $\{\lambda_\varepsilon(U, V)\} \leq \lambda^* < 1$. This implies that $\sup_{U,V} \lambda_\varepsilon(U, V) = \max_{U,V} \lambda_\varepsilon(U, V) < 1$. Notably, this finite set property of the ratio avoids the extreme case that may lead to a conservative conclusion about a non-expansive distribution Bellman operator, which we will give more details later.

Scale-sensitive Property. By definition of Sinkhorn divergence [26, 81], the pdf of Gibbs kernel in the equivalent form of Sinkrhon divergence is $\mathcal{K}(U, V)$, which satisfies $\mathcal{K}(U, V) \propto e^{\frac{-c(x,y)}{\varepsilon}}\mu(x)\nu(y)$. In particular, the pdf of Gibbs kernel is defined as

$$\frac{d\mathcal{K}}{d(\mu \otimes \nu)}(x, y) = \frac{\exp(-c/\varepsilon)}{\int \exp(-c/\varepsilon) d(\mu \otimes \nu)},$$

where the denominator is the normalization factor. After a scaling transformation, the pdf of aU and aV with respect to x and y would be $\frac{1}{a}\mu(\frac{x}{a})$ and $\frac{1}{a}\nu(\frac{y}{a})$. Thus $\mathcal{K}(aU, aV) \propto e^{\frac{-c(x,y)}{\varepsilon}}\frac{1}{a}\mu(\frac{x}{a})\frac{1}{a}\nu(\frac{y}{a})$. In the following proof, we use the change variable formula (multivariate version) constantly, while changing the joint pdf $\pi(x, y)$ and keep the cost function term $c(x, y)$. In particu-

lar, we denote Π^* and Π^0 as the optimal joint distribution of $\mathcal{W}_{c,\varepsilon}(\mu,\nu)$ and $\mathcal{W}_{c,\varepsilon}(a\mu,a\nu)$. Then we have:

$$
\begin{aligned}
&\mathcal{W}_{c,\varepsilon}(aU,aV)\\
&=\int c(x,y)\mathrm{d}\Pi^0(x,y)+\varepsilon\mathrm{KL}(\Pi^0|a\mu\otimes a\nu)\\
&\leq\int c(x,y)\mathrm{d}\Pi^*(x,y)+\varepsilon\mathrm{KL}(\Pi^*|a\mu\otimes a\nu)\\
&\overset{c=-k_\alpha}{=}\int(x-y)^\alpha\frac{1}{a^2}\pi^*(\frac{x}{a},\frac{y}{a})\mathrm{d}x\mathrm{d}y+\varepsilon\int\frac{1}{a^2}\pi^*(\frac{x}{a},\frac{y}{a})\log\frac{\frac{1}{a^2}\pi^*(\frac{x}{a},\frac{y}{a})}{\frac{1}{a^2}\mu(\frac{x}{a})\nu(\frac{y}{a})}\mathrm{d}x\mathrm{d}y\\
&=|a|^\alpha\int(x-y)^\alpha\pi^*(x,y)\mathrm{d}x\mathrm{d}y+\varepsilon\int\pi^*(x,y)\log\frac{\pi^*(x,y)}{\mu(x)\nu(y)}\mathrm{d}x\mathrm{d}y\\
&=|a|^\alpha\int(x-y)^\alpha\pi^*(x,y)\mathrm{d}x\mathrm{d}y+(|a|^\alpha+1-|a|^\alpha)\varepsilon\int\pi^*(x,y)\log\frac{\pi^*(x,y)}{\mu(x)\nu(y)}\mathrm{d}x\mathrm{d}y\\
&=|a|^\alpha\mathcal{W}_{c,\varepsilon}(U,V)+(1-|a|^\alpha)\varepsilon\mathrm{KL}(\Pi^*|\mu\otimes\nu)\\
&=\Delta_\varepsilon^{U,V}(a,\alpha)\mathcal{W}_{c,\varepsilon}(U,V)
\end{aligned}
\tag{5.21}
$$

where $\Delta_\varepsilon^{U,V}(a,\alpha)=|a|^\alpha+(1-|a|^\alpha)\lambda_\varepsilon(U,V)=|a|^\alpha(1-\lambda_\varepsilon(U,V))+\lambda_\varepsilon(U,V)\in(|a|^\alpha,1)$ for $\varepsilon\in(0,+\infty)$ and $a<1$ due to the fact that $\lambda_\varepsilon(U,V)\in(0,1)$ for any non-trivial $\mathcal{W}_{c,\varepsilon}(U,V)$. The non-trivial $\mathcal{W}_{c,\varepsilon}(U,V)$ rules out the case when the regularization term is zero, e.g., $\epsilon=0$ or the optimal coupling is the product of two margins. In other words, $\Delta_\varepsilon^{U,V}(a,\alpha)$ is a function less than 1, which depends on the two margins, including their independence and distribution similarity, the scale factor a and the order α.

Ruling Out Extreme Cases in the Convergence via a Finite Set. However, the fact that $\Delta_\varepsilon^{U,V}(a,\alpha)<1$ can only guarantee a "conservative" non-expansive contraction rather than a desirable contraction of the distributional Bellman operator. This is because there will be extreme cases in the power of series in general, although it is very unlikely to occur given a certain MDP in practice. For example, denote the non-constant factor as q_k for the k-th distributional Bellman update, where $q_k<1$. We can construct a counterexample as $q_k=1-1/(k+2)^2$. In this case, $\Pi_{k=1}^{+\infty}q_k=(\frac{2}{3}\frac{4}{3})(\frac{3}{4}\frac{5}{4})\cdots>0$ instead of the convergence to 0 and the non-zero limit can not guarantee the

contraction. It also intuitively implies that iteratively applying distribution Bellman operator under $\mathcal{W}_{c,\varepsilon}$ may not lead to convergence *in general by considering all possible return distributions* given the non-constant factor $\Delta_\varepsilon^{U,V}(a,\alpha)$. Although we know these extreme cases are very unlikely to happen, we have to rule out these extreme cases for a rigorous proof. As we have the assumption of a finite set of probability measures, the set of $\{\lambda_\varepsilon(U,V)\}$ is also finite. As the real set is dense, we can always find a positive constant that can be used as the contraction factor. Alternatively, we can directly use the $\sup_{U,V}\lambda_\varepsilon(U,V)$ as the uniform upper bound across the whole set of interested probability measures. Under this condition, we can immediately find a universal upper bound of $\Delta_\varepsilon^{U,V}(a,\alpha)$:

$$\begin{aligned}\sup_{U,V}\Delta_\varepsilon^{U,V}(a,\alpha) &= |a|^\alpha + (1-|a|^\alpha)\sup_{U,V}\lambda_\varepsilon(U,V)\\ &= |a|^\alpha(1-\sup_{U,V}\lambda_\varepsilon(U,V)) + \sup_{U,V}\lambda_\varepsilon(U,V) \qquad (5.22)\\ &\dot{=} \Delta_\varepsilon(a,\alpha)\end{aligned}$$

where the upper bound $\sup_{U,V}\Delta_\varepsilon^{U,V}(a,\alpha)$ has an interpolation form, which can be viewed as the convex combination between $|a|^\alpha$ and 1 with the coefficient $\sup_{U,V}\lambda_\varepsilon(U,V)$ determined by the probability measure set. More importantly, $\sup_{U,V}\Delta_\varepsilon^{U,V}(a,\alpha)$ is strictly less than 1, which is guaranteed by the finite set of $\{\lambda_\varepsilon(U,V)\}$ thanks to a finite set of interested probability measures. Finally, we have the variant of scale-sensitive property as follows, where the factor $\Delta_\varepsilon(a,\alpha)$ depends on α, a and the probability measure set.

$$\mathcal{W}_{c,\varepsilon}(aU,aV) \le \Delta_\varepsilon(a,\alpha)\mathcal{W}_{c,\varepsilon}(U,V). \qquad (5.23)$$

5.8.5 Proof of Theorem 10

$\varepsilon \to 0$ and $c = -k_\alpha$. We study the uniform convergence when $\varepsilon \to 0$. The proof is summarized from the optimal transport literature [36, 31] and we here provide the detailed proof for completeness. On the one hand, $\mathcal{W}_{c,\varepsilon} \ge \int (x-y)^\alpha d\Pi^*(x,y)dxdy \ge W_\alpha^\alpha$ as $\mathrm{KL} \ge 0$. We want to provide the inequality on the other side. Denote Π' as the minimizer in the Wasserstein distance W_α^α.

For any $\delta > 0$, there always exists a joint distribution Π^δ such that

$$|\int (x-y)^\alpha d\Pi'(x,y) - \int (x-y)^\alpha d\Pi^\delta(x,y)| \leq \delta \tag{5.24}$$

and $\mathrm{KL}(\Pi^\delta|\mu\otimes\nu) < +\infty$, i.e., $\int (x-y)^\alpha d\Pi^\delta(x,y) - \int (x-y)^\alpha d\Pi'(x,y) \leq \delta$. One possible way to find Π^δ is provided in notes of Lecture 6 in Optimal Transport Course[3] and we invite interested readers for reference. It follows that

$$\begin{aligned} W_\alpha^\alpha \leq \mathcal{W}_{c,\varepsilon} &\leq \int (x-y)^\alpha d\Pi^\delta(x,y) + \varepsilon \mathrm{KL}(\Pi^\delta|\mu\otimes\nu) \\ &\leq \int (x-y)^\alpha d\Pi'(x,y) + \delta + \varepsilon \mathrm{KL}(\Pi^\delta|\mu\otimes\nu), \end{aligned} \tag{5.25}$$

where the RHS $\int (x-y)^\alpha d\Pi'(x,y) + \delta + \varepsilon \mathrm{KL}(\Pi^\delta|\mu\otimes\nu) \to \int (x-y)^\alpha d\Pi'(x,y) + \delta = W_\alpha^\alpha + \delta$ as $\varepsilon \to 0$. As $\delta > 0$ is arbitrary, combing the two sides, it shows that $\mathcal{W}_{c,\epsilon} \to W_\alpha^\alpha$ as $\varepsilon \to 0$. Thus, Sinkhorn divergence maintains the properties of Wasserstein distance when $\varepsilon \to 0$.

When $\varepsilon = 0$, it has been shown that W_α can guarantee a γ-contraction property for distributional Bellman operator [9]. The crux of proof is that W_α is γ-scale sensitive:

$$\begin{aligned} W_\alpha(aU, aV) &= \left(\inf_{\Pi \in \Pi(aU, aV)} \int a^\alpha (x-y)^p d\Pi(x,y) \right)^{1/\alpha} \\ &\leq a \left(\inf_{\Pi \in \Pi(U,V)} \int (x-y)^p d\Pi(x,y) \right)^{1/\alpha} \\ &= a W_\alpha(U,V), \end{aligned} \tag{5.26}$$

where the inequality comes from the change of optimal joint distribution. Therefore, $W_\alpha(aU, aV) \leq a W_\alpha(U,V)$ guarantees a γ-contraction property for the distributional Bellman operator. As such, for W_α^α, when $\varepsilon = 0$, it suggest that $\overline{\mathcal{W}}_{c,0} = W_\alpha^\alpha$ corresponds to a γ^α-contraction for the distributional Bellman operator $\mathfrak{T}^\pi$.

[3] `https://lchizat.github.io/ot2021orsay.html`

$\varepsilon \to \infty$ **and** $c = -k_\alpha$**.** Our complete proof is inspired by [82, 36]. Recap the definition of squared MMD is

$$\mathbb{E}\left[k\left(\mathbf{X}, \mathbf{X}'\right)\right] + \mathbb{E}\left[k\left(\mathbf{Y}, \mathbf{Y}'\right)\right] - 2\mathbb{E}[k(\mathbf{X}, \mathbf{Y})]. \tag{5.27}$$

When the kernel function k degenerates to an unrectified $k_\alpha(x, y) := -\|x-y\|^\alpha$ for $\alpha \in (0, 2)$, the squared MMD would degenerate to

$$2\mathbb{E}\|\mathbf{X} - \mathbf{Y}\|^\alpha - \mathbb{E}\|\mathbf{X} - \mathbf{X}'\|^\alpha - \mathbb{E}\|\mathbf{Y} - \mathbf{Y}'\|^\alpha. \tag{5.28}$$

where $\mathbf{X}, \mathbf{X}' \overset{\text{i.i.d.}}{\sim} \mu, \mathbf{Y}, \mathbf{Y}' \overset{\text{i.i.d.}}{\sim} \nu$ and $\mathbf{X}, \mathbf{X}', \mathbf{Y}, \mathbf{Y}'$ are mutually independent. On the other hand, by definition, we have the Sinkhorn loss as

$$\overline{\mathcal{W}}_{c,\infty}(\mu, \nu) = 2\mathcal{W}_{c,\infty}(\mu, \nu) - \mathcal{W}_{c,\infty}(\mu, \mu) - \mathcal{W}_{c,\infty}(\nu, \nu). \tag{5.29}$$

Denoting Π_ε be the unique minimizer for $\overline{\mathcal{W}}_{c,\varepsilon}$, it holds that $\Pi_\varepsilon \to \mu \otimes \nu$ as $\varepsilon \to \infty$, which is the product of two marginal distributions. That being said, $\mathcal{W}_{c,\infty}(\mu, \nu) \to \int c(x, y)\mathrm{d}\mu(x)\mathrm{d}\nu(y) + 0 = \int c(x, y)\mathrm{d}\mu(x)\mathrm{d}\nu(y)$. *One important proof insight here is although* $\varepsilon \to +\infty$*, the KL term tends to zero, which is faster than* ε*. Therefore, the whole regularization term still tends to 0 as* $\varepsilon \to +\infty$. If $c = -k_\alpha = \|x - y\|^\alpha$, we eventually have $\mathcal{W}_{-k_\alpha,\infty}(\mu, \nu) \to \int \|x - y\|^\alpha \mathrm{d}\mu(x)\mathrm{d}\nu(y) = \mathbb{E}\|\mathbf{X} - \mathbf{Y}\|^\alpha$, where μ and ν can be inherently correlated, although the minimizer degenerates to the product of the two marginal distributions. Finally, we can have

$$\overline{\mathcal{W}}_{-k_\alpha,\infty} \to 2\mathbb{E}\|\mathbf{X} - \mathbf{Y}\|^\alpha - \mathbb{E}\|\mathbf{X} - \mathbf{X}'\|^\alpha - \mathbb{E}\|\mathbf{Y} - \mathbf{Y}'\|^\alpha, \tag{5.30}$$

which is exactly the form of squared MMD with the unrectified kernel k_α. Now the key is to prove that $\Pi_\varepsilon \to \mu \otimes \nu$ as $\varepsilon \to \infty$. We now give the detailed proof.

Firstly, it is apparent that $\mathcal{W}_{c,\varepsilon}(\mu, \nu) \leq \int c(x, y)\mathrm{d}\mu(x)\mathrm{d}\nu(y)$ as $\mu \otimes \nu \in \Pi(\mu, \nu)$. Let $\{\varepsilon_k\}$ be a positive sequence that diverges to ∞, and Π_k be the corresponding sequence of unique minimizers for $\mathcal{W}_{c,\varepsilon}$. According to the optimality condition, it must be the case that $\int c(x, y)\mathrm{d}\Pi_k + \varepsilon_k \mathrm{KL}(\Pi_k, \mu \otimes \nu) \leq$

$\int c(x,y)\mathrm{d}\mu\otimes\nu + 0$ (when $\Pi(\mu,\nu)=\mu\otimes\nu$). Thus,

$$\mathrm{KL}\left(\Pi_k, \mu\otimes\nu\right) \leqslant \frac{1}{\varepsilon_k}\left(\int c\ \mathrm{d}\mu\otimes\nu - \int c\ \mathrm{d}\Pi_k\right) \to 0.$$

Besides, by the compactness of $\Pi(\mu,\nu)$, we can extract a converging subsequence $\Pi_{n_k}\to\Pi_\infty$. Since KL is weakly lower-semicontinuous, it holds that

$$\mathrm{KL}\left(\Pi_\infty, \mu\otimes\nu\right) \leqslant \lim_{k\to\infty}\inf \mathrm{KL}\left(\Pi_{n_k}, \mu\otimes\nu\right) = 0$$

Hence $\Pi_\infty = \mu\otimes\nu$. That being said that the optimal coupling is simply the product of the marginals, indicating that $\Pi_\varepsilon\to\mu\otimes\nu$ as $\varepsilon\to\infty$. As a special case, when $\alpha=1$, $\overline{\mathcal{W}}_{-k_1,\infty}(u,v)$ is equivalent to the energy distance

$$d_E(\mathbf{X},\mathbf{Y}) := 2\mathbb{E}\|\mathbf{X}-\mathbf{Y}\| - \mathbb{E}\|\mathbf{X}-\mathbf{X}'\| - \mathbb{E}\|\mathbf{Y}-\mathbf{Y}'\|. \tag{5.31}$$

In summary, if the cost function is the rectified kernel k_α, it is the case that $\overline{\mathcal{W}}_{-k_\alpha,\varepsilon}$ converges to the squared MMD as $\varepsilon\to\infty$. According to [77], $\mathfrak{T}^\pi$ is $\gamma^{\alpha/2}$-contractive in the supremal form of MMD with the unrectified kernel k_α. As $\overline{\mathcal{W}}_{c,\varepsilon}(\mu,\nu)\to\mathrm{MMD}^2_{k_\alpha}(\mu,\nu)$, which is a squared MMD instead of MMD, it implies that $\mathfrak{T}^\pi$ is γ^α-contractive under the squared MMD / $\overline{\mathcal{W}}_{c,+\infty}$.

$\varepsilon\in(0,+\infty)$ **and** $c=-\kappa_\alpha$ In the proof of Proposition 12, we have shown that the Sinkhorn loss $\mathcal{W}_{c,\varepsilon}$ satisfies the sum-invariant **(I)** and a new variant of scale-sensitive properties as follows:

$$\begin{aligned}\mathcal{W}_{c,\varepsilon}(A+U,A+V) &\leq \mathcal{W}_{c,\varepsilon}(U,V)\\ \mathcal{W}_{c,\varepsilon}(aU,aV) &\leq \Delta_\varepsilon(a,\alpha)\mathcal{W}_{c,\varepsilon}(U,V).\end{aligned} \tag{5.32}$$

The Sinkhorn divergence $\overline{\mathcal{W}}_{c,\varepsilon}$ is defined by additionally subtracting two self-distance terms ($\mathcal{W}_{c,\varepsilon}(\mu,\mu)$ and $\mathcal{W}_{c,\varepsilon}(\nu,\nu)$) based on $\mathcal{W}_{c,\varepsilon}(\mu,\nu)$ in order to guarantee the non-negativity, tri-angularity and metric properties. These two self-distance terms do not change the **(I)** and **(S)** properties when extending $\mathcal{W}_{c,\varepsilon}$ to $\overline{\mathcal{W}}_{c,\varepsilon}$, and some proof techniques can refer to Section 2 in [31]. The only difference is that the scaling factor will be $\overline{\Delta}_\varepsilon^{U,V}(a,\alpha)$, which is the counterpart

of Eq. 5.21 satisfying

$$\overline{\mathcal{W}}_{c,\varepsilon}(aU, aV) \leq \overline{\Delta}_{\varepsilon}^{U,V}(a,\alpha)\overline{\mathcal{W}}_{c,\varepsilon}(U,V). \tag{5.33}$$

where $\overline{\Delta}_{\varepsilon}^{U,V}(a,\alpha) = |a|^{\alpha}(1-\overline{\lambda}_{\varepsilon}(U,V)) + \overline{\lambda}_{\varepsilon}(U,V) \in (|a|^{\alpha},1)$ for $\varepsilon \in (0,+\infty)$ and $a < 1$ due to the fact that $\overline{\lambda}_{\varepsilon}(U,V) \in (0,1)$ for any non-trivial $\overline{\mathcal{W}}_{c,\varepsilon}(U,V)$. The new ratio $\overline{\lambda}_{\varepsilon}(U,V) = \frac{\varepsilon \mathrm{KL}(\Pi^*|\mu\otimes\nu)}{\overline{\mathcal{W}}_{c,\varepsilon}} \in (0,1)$ for any considered U, V with measures μ, ν in the interested probability measure set. In particular, in the context of distributional RL, the set of interested probability measures would be the return distribution set of $\{Z(s,a)\}$ for $s \in \mathcal{S}$ and $a \in \mathcal{A}$ in a given finite MDP. We now want to find the universal upper bound $\overline{\Delta}_{\varepsilon}(a,\alpha)$, defined by

$$\overline{\Delta}_{\varepsilon}(a,\alpha) = |a|^{\alpha}(1-\sup_{U,V}\overline{\lambda}_{\varepsilon}(U,V)) + \sup_{U,V}\overline{\lambda}_{\varepsilon}(U,V)) \in (|a|^{\alpha},1). \tag{5.34}$$

Following the proof in Appendix 5.8.4, the finite MDP guarantees a finite ratio set of $\{\overline{\lambda}_{\varepsilon}(U,V)\}$, and thus we can find a universal upper bound $\overline{\lambda}^*$ of the ratio set such that $\{\overline{\lambda}_{\varepsilon}(U,V)\} \leq \overline{\lambda}^* < 1$. This also implies that $\sup_{U,V}\overline{\lambda}_{\varepsilon}(U,V) \in (0,1)$ and thus the scaling factor $\overline{\Delta}_{\varepsilon}(a,\alpha) \in (|a|^{\alpha},1)$, which is strictly less than 1. Therefore, we have the **(I)** and **(S)** properties of $\overline{\mathcal{W}}_{c,\varepsilon}$:

$$\begin{aligned} \overline{\mathcal{W}}_{c,\varepsilon}(A+U, A+V) &\leq \overline{\mathcal{W}}_{c,\varepsilon}(U,V) \\ \overline{\mathcal{W}}_{c,\varepsilon}(aU, aV) &\leq \overline{\Delta}_{\varepsilon}(a,\alpha)\overline{\mathcal{W}}_{c,\varepsilon}(U,V). \end{aligned} \tag{5.35}$$

Putting all together, we now derive the convergence of distributional Bellman operator $\mathfrak{T}^{\pi}$ under the supreme form of $\overline{\mathcal{W}}_{c,\varepsilon}$, i.e., $\overline{\mathcal{W}}_{c,\varepsilon}^{\infty}$:

$$\begin{aligned} \overline{\mathcal{W}}_{c,\varepsilon}^{\infty}(\mathfrak{T}^{\pi}Z_1, \mathfrak{T}^{\pi}Z_2) &= \sup_{s,a}\overline{\mathcal{W}}_{c,\varepsilon}(\mathfrak{T}^{\pi}Z_1(s,a), \mathfrak{T}^{\pi}Z_2(s,a)) \\ &= \sup_{s,a}\overline{\mathcal{W}}_{c,\varepsilon}(R(s,a)+\gamma Z_1(s',a'), R(s,a)+\gamma Z_2(s',a')) \\ &\overset{(a)}{\leq} \overline{\mathcal{W}}_{c,\varepsilon}(\gamma Z_1(s',a'), \gamma Z_2(s',a')) \\ &\overset{(b)}{\leq} \overline{\Delta}_{\varepsilon}^{Z_1(s',a'),Z_2(s',a')}(\gamma,\alpha)\overline{\mathcal{W}}_{c,\varepsilon}(Z_1(s',a'), Z_2(s',a')) \\ &\leq \sup_{s',a'}\overline{\Delta}_{\varepsilon}^{Z_1(s',a'),Z_2(s',a')}(\gamma,\alpha)\sup_{s',a'}\overline{\mathcal{W}}_{c,\varepsilon}(Z_1(s',a'), Z_2(s',a')) \\ &= \overline{\Delta}_{\varepsilon}(\gamma,\alpha)\overline{\mathcal{W}}_{c,\varepsilon}^{\infty}(Z_1, Z_2) \end{aligned} \tag{5.36}$$

where the inequality (a) is based on the sum invariant property **(I)** of Sinkhorn divergence. (b) is based on the new variant of scale-sensitive property **(S)** of Sinkhorn divergence and the leverage of $c = -k_\alpha$. Notably, $\overline{\Delta}_\varepsilon(\gamma, \alpha) \in (|\gamma|^\alpha, 1)$ is an MDP-dependent constant, also determined by γ, ε and α. As such, we conclude that distributional Bellman operator is *at least* $\overline{\Delta}_\varepsilon(\gamma, \alpha)$-contractive, where the contraction factor $\overline{\Delta}_\varepsilon(\gamma, \alpha)$ is strictly less than 1 in a given finite MDP. Based on Banach fixed point theorem, we have a unique optimal return distribution by iteratively applying $\mathfrak{T}^\pi$ in distributional dynamic programming.

5.8.6 Proof of Corollary 2

Proof. The contraction conclusion that extends to the multi-dimensional return distributions is straightforward. As the definition of Sinkhorn divergence inherently allows the multi-dimensional measures, the sum-invariant and the variant of scale-sensitive properties hold naturally. Specifically, after recapping to proof of these properties, we only need to change $c(x, y) = (x - y)^\alpha$ to $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^\alpha$ and re-define two d-dimensional random vector $\mathbf{U}$ and $\mathbf{V}$ with the d-dimensional probability measure μ and ν. Therefore, the **(I)** and **(S)** properties in the multi-dimensional reward settings are:

$$\overline{\mathcal{W}}_{c,\varepsilon}(\mathbf{A} + \mathbf{U}, \mathbf{A} + \mathbf{V}) \leq \overline{\mathcal{W}}_{c,\varepsilon}(\mathbf{U}, \mathbf{V}),\ \overline{\mathcal{W}}_{c,\varepsilon}(a\mathbf{U}, a\mathbf{V}) \ \leq \overline{\Delta}_\varepsilon(a, \alpha)\overline{\mathcal{W}}_{c,\varepsilon}(\mathbf{U}, \mathbf{V}), \tag{5.37}$$

where $\mathbf{A}$ is a d-dimensional random vector independent of $\mathbf{U}$ and $\mathbf{V}$. By leveraging these two properties, we now derive the convergence of distributional Bellman operator $\mathfrak{T}_d^\pi$ under $\overline{\mathcal{W}}_{c,\varepsilon}^\infty$ in the joint return distribution setting. Given two d-dimensional return distributions $\mathbf{Z}_1$ and $\mathbf{Z}_2$, we have

$$\begin{aligned}
\overline{\mathcal{W}}_{c,\varepsilon}^\infty(\mathfrak{T}_d^\pi \mathbf{Z}_1, \mathfrak{T}_d^\pi \mathbf{Z}_2) &= \sup_{s,a} \overline{\mathcal{W}}_{c,\varepsilon}(\mathfrak{T}_d^\pi \mathbf{Z}_1(s,a), \mathfrak{T}_d^\pi \mathbf{Z}_2(s,a)) \\
&= \sup_{s,a} \overline{\mathcal{W}}_{c,\varepsilon}(\mathbf{R}(s,a) + \gamma \mathbf{Z}_1(s',a'), \mathbf{R}(s,a) + \gamma \mathbf{Z}_2(s',a')) \\
&\overset{(a)}{\leq} \overline{\mathcal{W}}_{c,\varepsilon}(\gamma \mathbf{Z}_1(s',a'), \gamma \mathbf{Z}_2(s',a')) \\
&\overset{(b)}{\leq} \overline{\Delta}_\varepsilon^{\mathbf{Z}_1(s',a'), \mathbf{Z}_2(s',a')}(\gamma, \alpha) \overline{\mathcal{W}}_{c,\varepsilon}(\mathbf{Z}_1(s',a'), \mathbf{Z}_2(s',a')) \\
&\leq \sup_{s',a'} \overline{\Delta}_\varepsilon^{\mathbf{Z}_1(s',a'), \mathbf{Z}_2(s',a')}(\gamma, \alpha) \sup_{s',a'} \overline{\mathcal{W}}_{c,\varepsilon}(\mathbf{Z}_1(s',a'), \mathbf{Z}_2(s',a')) \\
&= \overline{\Delta}_\varepsilon(\gamma, \alpha) \overline{\mathcal{W}}_{c,\varepsilon}^\infty(\mathbf{Z}_1, \mathbf{Z}_2)
\end{aligned} \tag{5.38}$$

where the inequality (a) is based on the sum invariant property **(I)** of Sinkhorn divergence that cancels the additive d-dimensional random vector $\mathbf{R}(s,a)$. (b) is based on the new variant of scale-sensitive property **(S)** of Sinkhorn divergence and the leverage of $c=-k_\alpha$, where the contraction factor $\overline{\Delta}_\varepsilon(\gamma,\alpha)$ will depend on the set of d-dimensional probability measures/distributions. Notably, the analysis of $\overline{\Delta}_\varepsilon(\gamma,\alpha)$ in the one-dimensional return setting established in Appendix 5.8.4 and Appendix 5.8.5 is also applicable in the multi-dimensional setting. □

5.8.7 Algorithm: Sinkhorn Iterations and Sinkhorn Distributional RL

Input: Two samples sequences $\{Z_i\}_{i=1}^N, \{\mathfrak{T}Z_j\}_{j=1}^N$, number of iterations L and hyperparameter ε.

1: $\hat{c}_{i,j}=c(Z_i,\mathfrak{T}Z_j)$ for $\forall i=1,...,N, j=1,...,N$
2: $\mathcal{K}_{i,j}=\exp(-\hat{c}_{i,j}/\varepsilon)$
3: $b_0 \leftarrow \mathbf{1}_N$
4: **for** $l=1,2,...,L$ **do**
5: $\quad a_l \leftarrow \frac{\mathbf{1}_N}{\mathcal{K}b_{l-1}}$, $b_l \leftarrow \frac{\mathbf{1}_N}{\mathcal{K}a_l}$
6: **end for**
7: $\widehat{\overline{\mathcal{W}}}_{c,\varepsilon}\left(\{Z_i\}_{i=1}^N,\{\mathfrak{T}Z_j\}_{j=1}^N\right)=\langle (K\odot\hat{c})b,a\rangle$

Return: $\widehat{\overline{\mathcal{W}}}_{c,\varepsilon}\left(\{Z_i\}_{i=1}^N,\{\mathfrak{T}Z_j\}_{j=1}^N\right)$

Algorithm 3: Sinkhorn Iterations to Approximate $\overline{\mathcal{W}}_{c,\varepsilon}\left(\{Z_i\}_{i=1}^N,\{\mathfrak{T}Z_j\}_{j=1}^N\right)$

Given two sample sequences $\{Z_i\}_{i=1}^N, \{\mathfrak{T}Z_j\}_{j=1}^N$ in the distributional RL algorithm, the optimal transport distance is equivalent to the form:

$$\min_{P\in\mathbb{R}_+^{N\times N}}\left\{\langle P,\hat{c}\rangle; P\mathbf{1}_N=\mathbf{1}_N, P^\top\mathbf{1}_N=\mathbf{1}_N\right\}, \tag{5.39}$$

where the empirical cost function is $\hat{c}_{i,j}=c(Z_i,\mathfrak{T}Z_j)$. By adding entropic regularization on optimal transport distance, Sinkhorn divergence can be viewed

to restrict the search space of P in the following scaling form:

$$P_{i,j} = a_i \mathcal{K}_{i,j} b_j, \tag{5.40}$$

where $\mathcal{K}_{i,j} = e^{-\hat{c}_{i,j}/\varepsilon}$ is the Gibbs kernel defined in Eq. 5.3. This allows us to leverage iterations regarding the vectors a and b. More specifically, we initialize $b_0 = \mathbf{1}_N$, and then the Sinkhorn iterations are expressed as

$$a_{l+1} \leftarrow \frac{\mathbf{1}_N}{\mathcal{K} b_l} \quad \text{and} \quad b_{l+1} \leftarrow \frac{\mathbf{1}_N}{\mathcal{K}^\top a_{l+1}}, \tag{5.41}$$

where $\frac{\cdot}{\cdot}$ indicates an entry-wise division. Combining Sinkhorn Iteration in Algorithm 3 and the generic update of Sinkhorn Distributional RL in Algorithm 2, we provide a full version of Sinkhorn Distributional RL in Algorithm 4.

Require: Number of generated samples N, the kernel k (e.g., unrectified kernel), discount factor $\gamma \in [0, 1]$, learning rate α, replay buffer M, main network Z_θ, target network Z_{θ^*}, number of iterations L, hyperparameter ε, and a behavior policy π based on Z_θ following an ϵ-greedy rule

Initialize θ and $\theta^* \leftarrow \theta$

for $t = 1, 2, \ldots$ **do**

  Take action $a_t \sim \pi(\cdot|s_t; \theta)$, receive reward $r_t \sim R(\cdot|s_t, a_t)$, and observe $s_{t+1} \sim P(\cdot|s_t, a_t)$

  Store (s_t, a_t, r_t, s_{t+1}) to the replay buffer M

  Randomly draw a batch of transition samples (s, a, r, s') from the replay buffer M

  Compute a greedy action: $a^* = \arg\max_{a' \in A} \frac{1}{N} \sum_{i=1}^{N} Z_{\theta^*}(s', a')_i$

  Compute the target Bellman return distribution: $\mathfrak{T}Z_i \leftarrow r + \gamma Z_{\theta^*}(s', a^*)_i, \forall 1 \leq i \leq N$

  Evaluate Sinkhorn divergence via Sinkhorn Iterations in Algorithm 3:

$$\overline{\mathcal{W}}_{c,\varepsilon}\left(\{Z_\theta(s,a)_i\}_{i=1}^N, \{\mathfrak{T}Z_j\}_{j=1}^N\right)$$

  Update the main network Z_θ: $\theta \leftarrow \theta - \alpha \nabla_\theta \overline{\mathcal{W}}_{c,\varepsilon}\left(\{Z_\theta(s,a)_i\}_{i=1}^N, \{\mathfrak{T}Z_j\}_{j=1}^N\right)$

  Periodically update the target network $\theta^* \leftarrow \theta$

end for

Algorithm 4: Sinkhorn Distributional RL

5.8.8 Learning Curves on 55 Atari Games

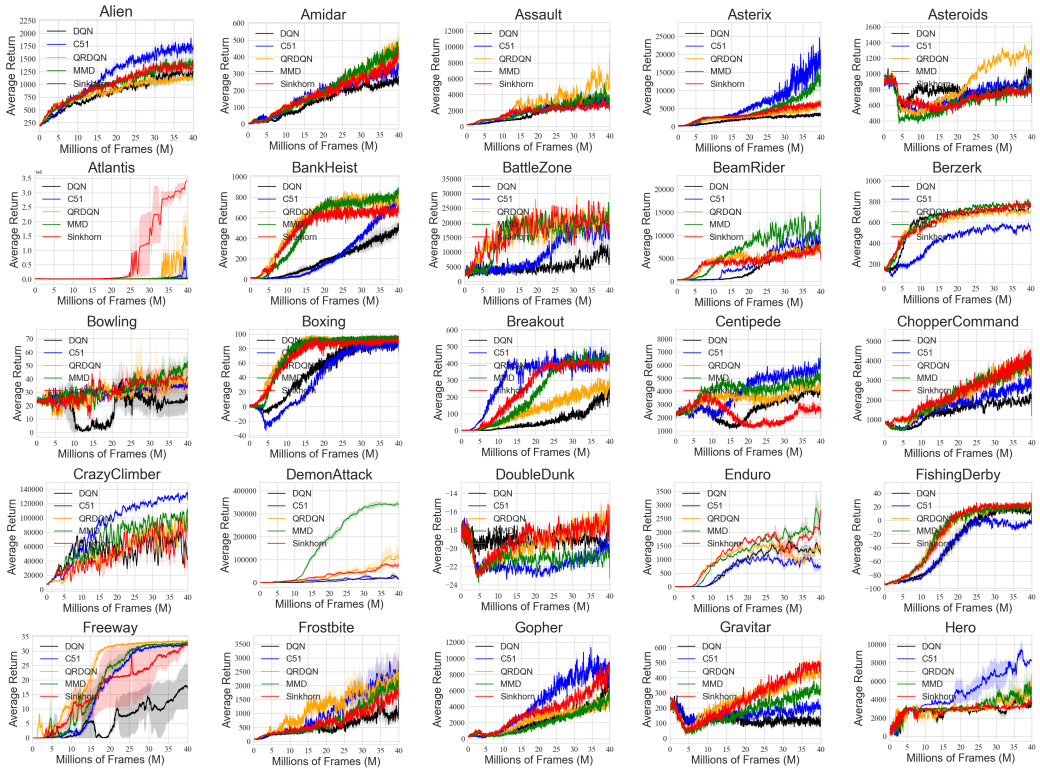


Figure 5.6: Part 1. Learning curves of SinkhornDRL on 55 Atari games after training 40M frames over 3 seeds.

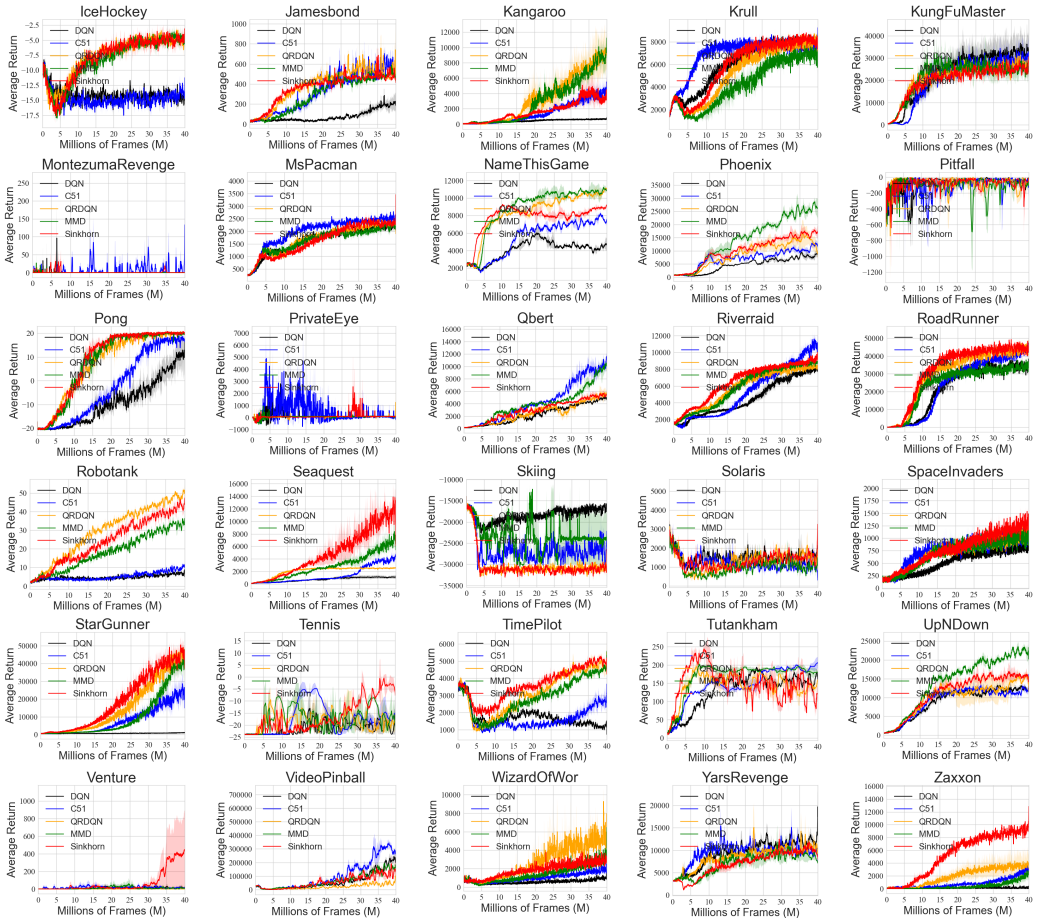


Figure 5.7: Part 2. Learning curves of SinkhornDRL on 55 Atari games after training 40M frames over 3 seeds.

5.8.9 Raw Score Table Across 55 Atari Games

GAMES	RANDOM	HUMAN	DQN	C51	QR-DQN	MMD-DQN	SinkhornDRL
Alien	211.9	7,127.7	1030	1510	1030	1480	**1560**
Amidar	2.34	1,719.5	341	424	**677**	510	588
Assault	283.5	742.0	3232	3647	**12943**	3295	2960
Asterix	268.5	8,503.3	3000	**34900**	11500	14900	6500
Asteroids	1008.6	47,388.7	1180	780	**1650**	1080	1370
Atlantis	22188	29,028.1	15500	84900	3316700	93600	**3447100**
BankHeist	14	753.1	570	960	**980**	880	700
BattleZone	3000	37,187.5	15000	19000	26000	**35000**	32000
BeamRider	414.3	16,926.5	8200	7476	7642	**25602**	6022
Berzerk	165.6	2,630.4	**970**	650	640	860	910
Bowling	23.48	160.7	54	43	**60**	**60**	**60**
Boxing	-0.69	12.1	94	90	**100**	**100**	**100**
Breakout	1.5	30.5	343	**452**	414	432	418
Centipede	2064.77	12,017.0	7551	4133	5388	**9342**	4070
ChopperCommand	794	7,387.8	1500	**3600**	3500	**3600**	3400
CrazyClimber	8043	35,829.4	94300	**153100**	139500	98500	137400
DemonAttack	162.25	1,971.0	31420	50240	240660	**407030**	105185
DoubleDunk	-18.14	-16.4	-16	-20	-18	-22	**-12**
Enduro	0.01	860.5	1387	1086	1972	1953	**4608**
FishingDerby	-93.06	-38.7	23	-1	31	31	**61**
Freeway	0.01	29.6	31	32	**34**	33	**34**
Frostbite	73.2	4,334.7	3330	**3690**	3470	3250	2640
Gopher	364	2,412.5	11400	**14780**	5440	3740	6620
Gravitar	226.5	3,351.4	350	350	**750**	350	500
Hero	551	30,826.4	3440	8535	**10155**	7195	6540
IceHockey	-10.3	0.9	-13	-10	-4	-3	**-2**
Jamesbond	27	302.8	350	600	**650**	450	500
Kangaroo	54	3,035.0	1300	6500	14600	**14800**	3600
Krull	1,566.59	2,665.5	8892	9336	**10053**	7762	9630
KungFuMaster	451	22,736.3	**46500**	38000	27900	26900	43600
MontezumaRevenge	0.0	4,753.3	1	**400**	1	1	1
MsPacman	242.6	6,951.6	3230	2440	1860	3130	**5120**
NameThisGame	2404.9	8,049.0	6160	5750	**13580**	9350	11250
Phoenix	757.2	7,242.6	9430	18780	9390	**25690**	23300
Pitfall	-265	6,463.7	1	1	1	1	1
Pong	-20.34	14.6	**21**	20	20	**21**	**21**
PrivateEye	34.49	69,571.3	100	100	100	100	100
Qbert	188.75	13,455.0	7425	**16375**	7800	16225	7750
RiverRaid	1575.4	17,118.0	8470	**13310**	8710	9190	9530
RoadRunner	7	7,845.0	45500	**60900**	52500	45600	59500
Robotank	2.24	11.9	8	11	**58**	39	54
Seaquest	88.2	42,054.7	1740	5940	2640	7370	**8350**
Skiing	-16267.9	-4,336.9	-13681	-20495	-29970	**-8986**	-23455
Solaris	2346.6	12,326.7	1640	660	2200	3380	**7720**
SpaceInvaders	136.15	1,668.7	940	**2480**	1170	770	1200
StarGunner	631	10,250.0	1200	17200	52900	52500	**57500**
Tennis	-23.92	-8.3	-23	-1	-7	-8	**5**
TimePilot	3682	5,229.2	800	4100	4400	**8000**	4500
Tutankham	15.56	167.6	201	213	**220**	141	137
UpNDown	604.7	11,693.2	14560	18440	13710	**27370**	18910
Venture	0.0	1,187.5	1	1	1	1	**700**
VideoPinball	15720.98	17,667.9	155165	**576843**	189460	69175	347700
WizardOfWor	534	4,756.5	1400	2400	**14300**	11500	4300
YarsRevenge	3271.42	54,576.9	**28048**	7882	17729	7520	9120
Zaxxon	8	9,173.3	1	3900	9100	4300	**19500**
Number of Best			4	12	15	13	**17**
Number of Second Best			6	7	10	8	**16**

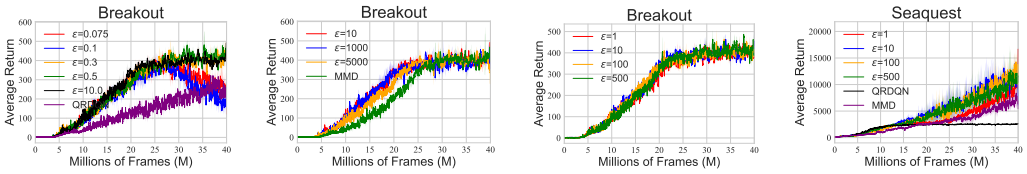
Table 5.2: Best score of all algorithms over 3 seeds across 55 Atari games after training 40M Frames. **Bold** denotes the best performance, while the underline represents the second best performance. The number of games with the best and second best performance substantiate the superiority of our SinkhornDRL across all considered baseline algorithms.

5.8.10 Features of Atari Games

GAMES	Action Space	Dynamics
Alien	18	Complex
Amidar	6	Simple
Assault	7	Complex
Asterix	18	Complex
Asteroids	4	Simple
Atlantis	4	Simple
BankHeist	18	Simple
BattleZone	18	Simple
BeamRider	18	Complex
Berzerk	18	Complex
Bowling	Continuous	Simple
Boxing	6	Simple
Breakout	4	Simple
Centipede	18	Complex
ChopperCommand	Continuous	Complex
CrazyClimber	18	Complex
DemonAttack	18	Complex
DoubleDunk	18	Simple
Enduro	9	Simple
FishingDerby	18	Simple
Freeway	3	Simple
Frostbite	18	Complex
Gopher	18	Simple
Gravitar	Continuous	Complex
Hero	18	Simple
IceHockey	Continuous	Simple
Jamesbond	18	Complex
Kangaroo	18	Complex
Krull	18	Complex
KungFuMaster	18	Complex
MontezumaRevenge	18	Complex
MsPacman	9	Simple
NameThisGame	18	Complex
Phoenix	18	Complex
Pitfall	18	Complex
Pong	3	Simple
PrivateEye	18	Complex
Qbert	6	Complex
Riverraid	18	Complex
RoadRunner	18	Simple
Robotank	9	Simple
Seaquest	18	Complex
Skiing	9	Simple
Solaris	18	Complex
SpaceInvaders	6	Simple
StarGunner	18	Complex
Tennis	18	Simple
TimePilot	18	Complex
Tutankham	18	Complex
UpNDown	18	Complex
Venture	18	Complex
VideoPinball	6	Simple
WizardOfWor	12	Complex
YarsRevenge	18	Complex
Zaxxon	18	Complex

Table 5.3: Number of Action space and difficulty of environmental dynamics of 55 Atari games.

5.8.11 Sensitivity Analysis and Computational Cost



(a) Small ε vs QR-DQN **(b)** Large ε vs MMD-DQN **(c)** ε on Breakout **(d)** ε on Seaquest

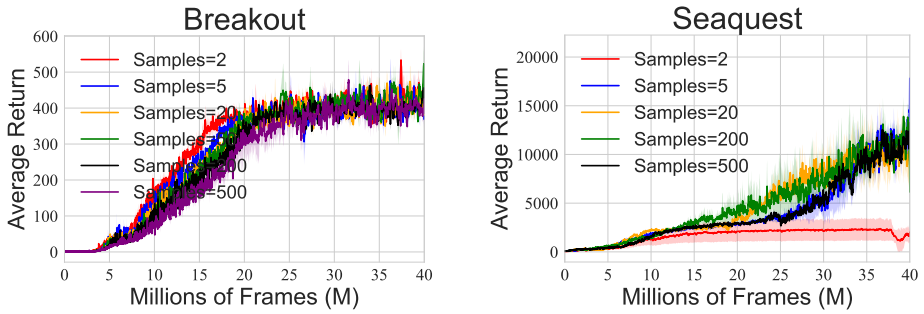
Figure 5.8: (a) Sensitivity analysis w.r.t. a small level of ε SinkhornDRL to compare with QR-DQN that approximates Wasserstein distance on Breakout. (b) Sensitivity analysis w.r.t. a large level of ε SinkhornDRL algorithm to compare with MMD-DQN on Breakout. All learning curves are reported over 2 seeds. (c) and (d) are results for a general ε on Breakout and Seaquest, respectively.

More results in Sensitivity Analysis

Decreasing ε. We argue that the limit behavior connection as stated in Theorem 10 may not be able to be verified rigorously via numeral experiments due to the numerical instability of Sinkhorn Iteration in Algorithm 3. From Figure 5.8 (a), we can observe that if we gradually decline ε to 0, SinkhornDRL's performance tends to degrade and approach QR-DQN. Note that an overly small ε will lead to a trivial almost 0 $\mathcal{K}_{i,j}$ in Sinkhorn iteration in Algorithm 3, and will cause $\frac{1}{0}$ numerical instability issue for a_l and b_l in Line 5 of Algorithm 3. In addition, we also conducted experiments on Seaquest, a similar result is also observed in Figure 5.8 (d). As shown in Figure 5.8 (d), the performance of SinkhornDRL is robust when $\varepsilon = 10, 100, 500$, but a small $\epsilon = 1$ tends to worsen the performance.

Increasing ε. Moreover, for breakout, if we increase ε, the performance of SinkhornDRL tends to degrade and be close to MMD-DQN as suggested in Figure 5.8 (b). It is also noted that an overly large ε will let the $\mathcal{K}_{i,j}$ explode to ∞. This also leads to the numerical instability issue in Sinkhorn iteration in Algorithm 3.

Samples N. We find that SinkhornDRL requires a proper number of samples N to perform favorably, and the sensitivity w.r.t N depends on the environment. As suggested in Figure 5.9 (a), a smaller N, e.g., $N = 2$ on breakout has already achieved favorable performance and even accelerates the convergence in the early phase, while $N = 2$ on Seaquest will lead to the divergence issue. Meanwhile, an overly large N worsens the performance across two games. We conjecture that using larger network networks to generate more samples may suffer from the overfitting issue, yielding the training instability [11]. In practice, we choose a proper number of samples, i.e., $N = 200$ across all games.

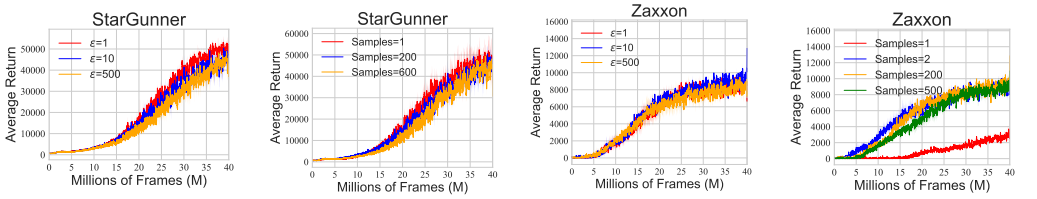


(a) Samples on Breakout **(b)** Samples on Seaquest

Figure 5.9: Sensitivity analysis of Sinkhorn in terms of the number of samples N on Breakout (a) and Seaquest (b).

More Games on StarGunner and Zaxxon. Beyond Breakout and Seaquest, we also provide sensitivity analysis on StarGunner and Zaxxon games in Figure 5.10. It suggests overly small samples, e.g., 1 and overall large samples tend to degrade the performance, especially on Zaxxon. Although the two games are robust to ε, and we find a small or large ε hurts the performance in Seaquest. Thus, considering all games, we set samples 200, and $\varepsilon = 10.0$ in a moderate range across all games, although a more careful tuning in each game will improve the performance further.

Comparison with the Computational Cost We evaluate the computational time every 10,000 iterations across the whole training process of all considered distributional RL algorithms and make a comparison in Figure 5.11.



(a) StarGunner: ε **(b)** StarGunner: Samples **(c)** Zaxxon: ε **(d)** Zaxxon: Samples

Figure 5.10: Sensitivity analysis of SinkhornDRL on StarGunner and Zaxxon in terms of ε, and number of samples. Learning curves are reported over 3 seeds.

It suggests that SinkhornDRL indeed increases around 50% computation cost compared with QR-DQN and C51, but only slightly increases the cost in contrast to MMD-DQN on both Breakout and Qbert games. We argue that this additional computational burden can be tolerant given the significant outperformance of SinkhornDRL in a large number of environments.

In addition, we also find that the number of Sinkhorn iterations L is negligible to the computation cost, while an overly large sample N, e.g., 500, will lead to a large computational burden as illustrated in Figure 5.12. This can be intuitively explained as the computation complexity of the cost function $c_{i,j}$ is $\mathcal{O}(N^2)$ in SinkhornDRL, which is particularly heavy in the computation if N is large enough.

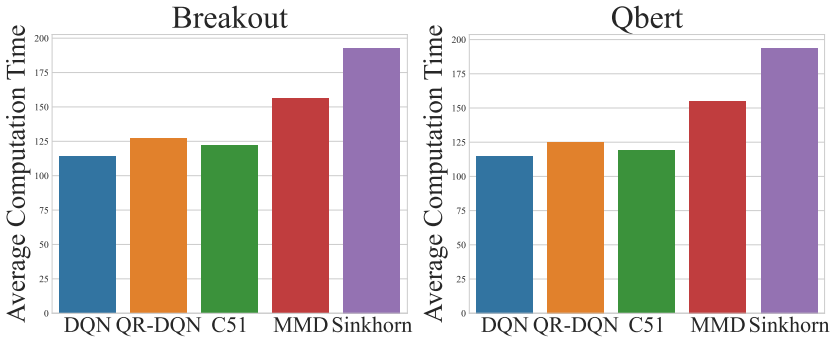


Figure 5.11: Average computational cost per 10,000 iterations of all considered distributional RL algorithm, where we select $\varepsilon = 10$, $L = 10$ and the number of samples $N = 200$ in SinkhornDRL algorithm.

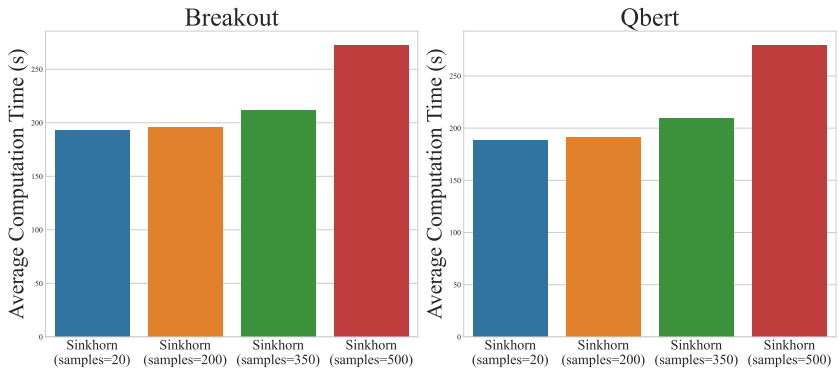


Figure 5.12: Average computational cost per 10,000 iterations of SinkhornDRL algorithm over different samples.

5.8.12 Experimental Setting in Multi-dimensional Return Distributions

Reward Structure and Decomposition. In practice, the reward function can be multi-dimensional [104, 62, 64, 23, 117, 63], where distributional RL is aimed at modeling multivariate return distribution with multiple reward sources. We follow the multi-dimensional return distribution setting in [117], which construct six Atari games with multiple sources of rewards by decomposing the scalar-valued primitive rewards into multi-dimension. For completeness, we introduce the respective reward structure and the decomposing method of the six considered Atari games, including AirRaid, Asteroids, Gopher, MsPacman, UpNDown, and Pong. The reward is decomposed while keeping the total reward unchanged.

- **AirRaid.** For primitive rewards, the agent kills different kinds of monsters and then receive discrete values of the rewards. The scalar-based primitive rewards are decomposed into four dimensions. The agent will get multi-dimensional rewards [100, 0, 0, 0], [0, 75, 0, 0], [0, 0, 50, 0],[0, 0, 0, 25], [0, 0, 0, 0] respectively for the primitive reward 100, 75, 50, 25 and 0.

- **Asteroids.** For primitive rewards, the agent kills different kinds of monsters and then receive values of the rewards. We denote the prim-

itive reward as r, and decompose it into the three-dimensional reward as $[r_1, r_2, r_3]$. If $(r - 20) \bmod 50 = 0$, we let $r_1 = 20$, otherwise $r_1 = 0$. If $(r - r_1 - 50) \bmod 100 = 0$, we let $r_2 = 50$, otherwise $r_2 = 0$. We let $r_3 = r - r_1 - r_2$.

- **Gopher.** For primitive rewards, the agent gets +80 reward for killing a monster and +20 reward after removing holes on the ground. We denote the primitive reward as r, and decompose it into the two-dimensions as $[r_1, r_2,]$. If $(r - 20) \bmod 100 = 0$, we let $r_1 = 20$, otherwise $r_1 = 0$. We let $r_2 = r - r_1$.

- **MsPacman.** The agent gets $\{+200, +400, +800, +1,600\}$ rewards after killing different monsters and +10 rewards after eating beans. In the reward decomposition, we decompose primitive reward denoted as r into four dimensions $[r_1, r_2, r_3, r_4]$. If $(r - 10) \bmod 50 = 0$, we let $r_1 = 10$, otherwise $r_1 = 0$. If $(r - r_1 - 50) \bmod 100 = 0$, we let $r_2 = 50$, otherwise $r_2 = 0$. If $(r - r_1 - r_2 - 100) \bmod 200 = 0$, we let $r_3 = 100$, otherwise $r_3 = 0$. We let $r_4 = r - r_1 - r_2 - r_3$.

- **Pong.** For primitive rewards, the agent gets +1 if it wins a round, and −1 for losing the round. We decompose the reward into two-dimension: the agent will get $[-1, 0]$ for a −1 reward, $[0, 1]$ for a +1 reward; otherwise, $[0, 0]$.

- **UpNDown.** For primitive rewards, the agent gets +400 reward for killing an energy car, +100 for reaching a flag, and +10 reward for being alive. We denote the primitive reward as r, and decompose it into the three-dimensional reward as $[r_1, r_2, r_3]$. If $(r - 10) \bmod 100 = 0$, we let $r_1 = 10$, otherwise $r_1 = 0$. If $(r - r_1 - 100) \bmod 200 = 0$, we let $r_2 = 100$, otherwise $r_2 = 0$. We let $r_3 = r - r_1 - r_2$.

Detailed Experimental Setup. Our implementation extends our code in one-dimensional return setting to multi-dimensional return scenario and adopts the key aspects in [117]. For instance, similar to [117], we leverage a clipping reward normalizer to clip the multi-dimensional rewards into $[-1, 1]$

after applying the reward decomposition procedure mentioned above to the primitive rewards. We keep the same model architecture except only modifying the output of the last layer from $(B, |\mathcal{A}|, N)$ to $(B, |\mathcal{A}|, D, N)$, where B is the batch size within each batch training, and D is the dimension of the decomposed mutivariate reward function in each game.

Baseline Algorithms. Quantile regression can be used to approximate 1-Wasserstein distance in one-dimensional setting [22] as the one-dimensional Wassertein distance has a closed-form expression via the quantile function. However, it remains elusive how to use quantile regression to approximate multi-dimensional Wasserstein distance. This is to say, it is still unclear how to extend the quantile regression distributional RL (QR-DQN) into multi-dimensional return distribution setting, resulting in no proper baseline in our experiment. Despite that, we directly compare SinkrhornDRL with MMD-DQN [77] as MMD is applicable and computationally tractable in the multi-dimensional setting. Notably, we did not introduce other baselines, such as Hybrid Reward Architecture (HRA) [104], or MD3QN [117]. This is because 1) [117] shows that their proposed MD3QN and HRA do not outperform MMD-DQN in most of the six Atari games. By contrast, as suggested in Figure 5.4, our SinkhornDRL has already surpassed MMD-DQN across almost all the considered games, and thus excels over MD3QN and HRA, correspondingly. 2) The primary focus of our study is the comprehensive advantages of SinkhornDRL over other distributional RL classes, especially in the more common setting within one-dimensional return distributions. The extension capability of SinkhornDRL into the multi-dimensional reward setting is one of its merits, which is not the primary focus of our study.

Chapter 6

Conclusion and Future Work

In conclusion, this thesis significantly advances the field of distributional reinforcement learning by investigating its theoretical advantages—through the lens of regularization, exploration, optimization, and robustness—and by innovating a novel algorithm class that incorporates insights from the optimal transport literature. We start by interpreting the advantages of being categorical distributional as a form of decomposed regularization effect, which promotes exploring states where the environmental uncertainty is largely underestimated. Further, we address the question: *how does return distribution in distributional RL help the optimization?* We examine this issue from the perspectives of uniform stability and acceleration effect in the optimization process. Additionally, we assess the training robustness of distributional RL against both random and adversarial noisy state observations, establishing the state-noisy MDP as a foundation for future robustness analyses in RL. Lastly, we propose a novel family of distributional RL algorithms based on Sinkhorn divergence, demonstrating competitive performance relative to the typical distributional RL algorithms.

This thesis also sets the stage for numerous promising research avenues. Firstly, it remains elusive whether it is feasible to extend the uncertainty-aware exploration, desirable optimization, and robustness properties in categorical distributional RL to more general algorithm classes. Although this extension is natural and instrumental, it is also considerably challenging, given that the analytical techniques in other classes, such as QR-DQN, are highly different

from CDRL. Moreover, it would also be crucial to investigate the advantages of distributional RL from additional perspectives, such as representation, and to develop more advanced distributional RL algorithms by diving deeper into the knowledge pools of probability, statistics, control, and optimal transport. More broadly, it has significant potential to apply distributional learning beyond the realm of RL to broader scenarios. Practically, it is equally essential to establish a criterion to determine which algorithms are likely to perform best under various conditions, thus guiding future developments and applications of distributional RL technologies.

Bibliography

[1] Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. Optimality and approximation with policy gradient methods in markov decision processes. In *Conference on Learning Theory*, pages 64–66. PMLR, 2020.

[2] Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320, 2021.

[3] Rohit Agrawal and Thibaut Horel. Optimal bounds between f-divergences and integral probability metrics. *Journal of Machine Learning Research*, 22(128):1–59, 2021.

[4] Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. Understanding the impact of entropy on policy optimization. In *International conference on machine learning*, pages 151–160. PMLR, 2019.

[5] Mokhtar Z Alaya, Maxime Berar, Gilles Gasso, and Alain Rakotomamonjy. Screening sinkhorn algorithm for regularized optimal transport. *Advances in Neural Information Processing Systems*, 32, 2019.

[6] Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *International Conference on Learning Representations*, 2017.

[7] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[8] Gabriel Barth-Maron, Matthew W Hoffman, David Budden, Will Dabney, Dan Horgan, Dhruva Tb, Alistair Muldal, Nicolas Heess, and Tim-

othy Lillicrap. Distributed distributional deterministic policy gradients. *International Conference on Learning Representations (ICLR)*, 2018.

[9] Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. *International Conference on Machine Learning (ICML)*, 2017.

[10] Marc G Bellemare, Ivo Danihelka, Will Dabney, Shakir Mohamed, Balaji Lakshminarayanan, Stephan Hoyer, and Rémi Munos. The cramer distance as a solution to biased wasserstein gradients. *arXiv preprint arXiv:1705.10743*, 2017.

[11] Johan Bjorck, Carla P Gomes, and Kilian Q Weinberger. Towards deeper deep reinforcement learning. *Advances in neural information processing systems (NeurIPS)*, 2021.

[12] Vivek S Borkar and Sean P Meyn. The ode method for convergence of stochastic approximation and reinforcement learning. *SIAM Journal on Control and Optimization*, 38(2):447–469, 2000.

[13] Tianshi Cao, Alex Bie, Arash Vahdat, Sanja Fidler, and Karsten Kreis. Don't generate me: Training differentially private generative models with sinkhorn divergence. *Advances in Neural Information Processing Systems*, 34:12480–12492, 2021.

[14] Johan Samir Obando Ceron and Pablo Samuel Castro. Revisiting rainbow: Promoting more insightful and inclusive deep reinforcement learning research. In *International Conference on Machine Learning*, pages 1373–1383. PMLR, 2021.

[15] Tong Chen, Jiqiang Liu, Yingxiao Xiang, Wenjia Niu, Endong Tong, and Zhen Han. Adversarial attack and defense in reinforcement learning-from ai security view. *Cybersecurity*, 2(1):11, 2019.

[16] Yu Chen, Xiangcheng Zhang, Siwei Wang, and Longbo Huang. Provable risk-sensitive distributional reinforcement learning with general function approximation. *International Conference on Machine Learning*, 2024.

[17] Yutian Chen, Max Welling, and Alex Smola. Super-samples from kernel herding. *UAI, 109–116. AUAI Press*, 2012.

[18] Taehyun Cho, Seungyub Han, Heesoo Lee, Kyungjae Lee, and Jungwoo Lee. Pitfall of optimism: Distributional reinforcement learning by randomizing risk criterion. *Advances in Neural Information Processing Systems*, 2023.

[19] Taehyun Cho, Seungyub Han, Heesoo Lee, Kyungjae Lee, and Jungwoo Lee. Pitfall of optimism: Distributional reinforcement learning by randomizing risk criterion. *Advances in Neural Information Processing Systems*, 36, 2024.

[20] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[21] Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. *International Conference on Machine Learning (ICML)*, 2018.

[22] Will Dabney, Mark Rowland, Marc G Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. *Association for the Advancement of Artificial Intelligence (AAAI)*, 2018.

[23] Christoph Dann, Yishay Mansour, and Mehryar Mohri. Reinforcement learning can be more efficient with multiple rewards. In *International Conference on Machine Learning*, pages 6948–6967. PMLR, 2023.

[24] John N Darroch and Douglas Ratcliff. Generalized iterative scaling for log-linear models. *The annals of mathematical statistics*, pages 1470–1480, 1972.

[25] Monroe D Donsker and SR Srinivasa Varadhan. Asymptotic evaluation of certain markov process expectations for large time—iii. *Communications on pure and applied Mathematics*, 29(4):389–461, 1976.

[26] Stephan Eckstein and Marcel Nutz. Quantitative stability of regularized optimal transport and convergence of sinkhorn's algorithm. *SIAM Journal on Mathematical Analysis*, 54(6):5922–5948, 2022.

[27] Odin Elie and Charpentier Arthur. *Dynamic Programming in Distributional Reinforcement Learning*. PhD thesis, Université du Québec à Montréal, 2020.

[28] Jianqing Fan, Zhaoran Wang, Yuchen Xie, and Zhuoran Yang. A theoretical analysis of deep q-learning. In *Learning for Dynamics and Control*, pages 486–489. PMLR, 2020.

[29] Jesse Farebrother, Jordi Orbay, Quan Vuong, Adrien Ali Taïga, Yevgen Chebotar, Ted Xiao, Alex Irpan, Sergey Levine, Pablo Samuel Castro, Aleksandra Faust, et al. Stop regressing: Training value functions via classification for scalable deep rl. *arXiv preprint arXiv:2403.03950*, 2024.

[30] Kilian Fatras, Thibault Séjourné, Rémi Flamary, and Nicolas Courty. Unbalanced minibatch optimal transport; applications to domain adaptation. In *International Conference on Machine Learning*, pages 3186–3197. PMLR, 2021.

[31] Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouvé, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2681–2690. PMLR, 2019.

[32] Joel Franklin and Jens Lorenz. On the scaling of multidimensional matrices. *Linear Algebra and its applications*, 114:717–735, 1989.

[33] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018.

[34] Kenji Fukumizu, Arthur Gretton, Gert Lanckriet, Bernhard Schölkopf, and Bharath K Sriperumbudur. Kernel choice and classifiability for rkhs embeddings of probability distributions. *Advances in neural information processing systems*, 22, 2009.

[35] Aude Genevay, Lénaic Chizat, Francis Bach, Marco Cuturi, and Gabriel Peyré. Sample complexity of sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1574–1583. PMLR, 2019.

[36] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pages 1608–1617. PMLR, 2018.

[37] Sina Ghiassian, Andrew Patterson, Shivam Garg, Dhawal Gupta, Adam White, and Martha White. Gradient temporal-difference learning with regularized corrections. In *International Conference on Machine Learning*, pages 3524–3534. PMLR, 2020.

[38] Florin Gogianu, Tudor Berariu, Mihaela C Rosca, Claudia Clopath, Lucian Busoniu, and Razvan Pascanu. Spectral normalisation for deep reinforcement learning: an optimisation perspective. In *International Conference on Machine Learning*, pages 3734–3744. PMLR, 2021.

[39] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. A kernel method for the two-sample-problem. *Advances in neural information processing systems*, 19, 2006.

[40] Ziwei Guan, Kaiyi Ji, Donald J Bucci Jr, Timothy Y Hu, Joseph Palombo, Michael Liston, and Yingbin Liang. Robust stochastic bandit algorithms under probabilistic unbounded adversarial attack. In *AAAI*, pages 4036–4043, 2020.

[41] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

[42] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pages 1352–1361. PMLR, 2017.

[43] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR, 2018.

[44] Seungyul Han and Youngchul Sung. A max-min entropy framework for reinforcement learning. *Advances in neural information processing systems (NeurIPS)*, 2021.

[45] Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, pages 1225–1234. PMLR, 2016.

[46] Hado Hasselt. Double q-learning. *Advances in neural information processing systems*, 23:2613–2621, 2010.

[47] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *NIPS Deep Learning Workshop*, 2015.

[48] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *Advances in Neural Information Processing Systems*, 2017.

[49] Peter J Huber. *Robust Statistics*, volume 523. John Wiley & Sons, 2004.

[50] Inaam Ilahi, Muhammad Usama, Junaid Qadir, Muhammad Umar Janjua, Ala Al-Fuqaha, Dinh Thai Hoang, and Dusit Niyato. Challenges and

countermeasures for adversarial attacks on deep reinforcement learning. *arXiv preprint arXiv:2001.09684*, 2020.

[51] Ehsan Imani and Martha White. Improving regression performance with distributional losses. In *International Conference on Machine Learning*, pages 2157–2166. PMLR, 2018.

[52] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *International Conference on Learning Representations*, 2016.

[53] Philippe Rigollet Jason Altschuler, Jonathan Weed. Near-linear time approximation algorithms for optimal transport via sinkhorn iteration, 2017.

[54] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.

[55] Qi Kuang, Zhoufan Zhu, Liwen Zhang, and Fan Zhou. Variance control for distributional reinforcement learning. *International Conference on Machine Learning*, 2023.

[56] Kimin Lee, Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Sunrise: A simple unified framework for ensemble learning in deep reinforcement learning. In *International Conference on Machine Learning*, pages 6131–6141. PMLR, 2021.

[57] Christian Léonard. A survey of the schr\" odinger problem and some of its connections with optimal transport. *arXiv preprint arXiv:1308.0215*, 2013.

[58] Alexander Li and Deepak Pathak. Functional regularization for reinforcement learning via learned fourier features. *Advances in Neural Information Processing Systems*, 34, 2021.

[59] 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.

[60] Shiau Hong Lim and Ilyas Malik. Distributional reinforcement learning for risk-sensitive policies. *Advances in Neural Information Processing Systems*, 35:30977–30989, 2022.

[61] Yen-Chen Lin, Zhang-Wei Hong, Yuan-Hong Liao, Meng-Li Shih, Ming-Yu Liu, and Min Sun. Tactics of adversarial attack on deep reinforcement learning agents. *arXiv preprint arXiv:1703.06748*, 2017.

[62] Zichuan Lin, Derek Yang, Li Zhao, Tao Qin, Guangwen Yang, and Tie-Yan Liu. Rd2: Reward decomposition with representation decomposition. *Advances in Neural Information Processing Systems*, 33:11298–11308, 2020.

[63] Zichuan Lin, Li Zhao, Derek Yang, Tao Qin, Tie-Yan Liu, and Guangwen Yang. Distributional reward decomposition for reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

[64] Daniel J Lizotte, Michael H Bowling, and Susan A Murphy. Efficient reinforcement learning with multiple reward functions for randomized controlled trial analysis. In *International Conference on Machine Learning*, volume 10, pages 695–702, 2010.

[65] Yudong Luo, Guiliang Liu, Haonan Duan, Oliver Schulte, and Pascal Poupart. Distributional reinforcement learning with monotonic splines. In *International Conference on Learning Representations*, 2021.

[66] Clare Lyle, Marc G Bellemare, and Pablo Samuel Castro. A comparative analysis of expected and distributional reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4504–4511, 2019.

[67] Xiaoteng Ma, Li Xia, Zhengyuan Zhou, Jun Yang, and Qianchuan Zhao. Dsac: Distributional soft actor critic for risk-sensitive reinforcement learning. *arXiv preprint arXiv:2004.14547*, 2020.

[68] Yecheng Ma, Dinesh Jayaraman, and Osbert Bastani. Conservative offline distributional reinforcement learning. *Advances in neural information processing systems*, 34:19235–19247, 2021.

[69] Borislav Mavrin, Shangtong Zhang, Hengshuai Yao, and Linglong Kong. Exploration in the face of parametric and intrinsic uncertainties. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2117–2119, 2019.

[70] Borislav Mavrin, Shangtong Zhang, Hengshuai Yao, Linglong Kong, Kaiwen Wu, and Yaoliang Yu. Distributional reinforcement learning for efficient exploration. *International Conference on Machine Learning (ICML)*, 2019.

[71] Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning*, pages 6820–6829. PMLR, 2020.

[72] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, et al. Learning to navigate in complex environments. *arXiv preprint arXiv:1611.03673*, 2016.

[73] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *International Conference on Learning Representations*, 2018.

[74] 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.

[75] Tetsuro Morimura, Masashi Sugiyama, Hisashi Kashima, Hirotaka Hachiya, and Toshiyuki Tanaka. Parametric return density estimation for reinforcement learning. *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2011.

[76] Rafael Müller, Simon Kornblith, and Geoffrey Hinton. When does label smoothing help? *Neural Information Processing Systems (NeurIPS)*, 2019.

[77] Thanh Tang Nguyen, Sunil Gupta, and Svetha Venkatesh. Distributional reinforcement learning with maximum mean discrepancy. *Association for the Advancement of Artificial Intelligence (AAAI)*, 2020.

[78] Yangchen Pan, Kirby Banman, and Martha White. Fuzzy tiling activations: A simple approach to learning sparse representations online. *International Conference on Learning Representations*, 2019.

[79] Giorgio Patrini, Rianne Van den Berg, Patrick Forre, Marcello Carioni, Samarth Bhargav, Max Welling, Tim Genewein, and Frank Nielsen. Sinkhorn autoencoders. In *Uncertainty in Artificial Intelligence*, pages 733–743. PMLR, 2020.

[80] Anay Pattanaik, Zhenyi Tang, Shuijing Liu, Gautham Bommannan, and Girish Chowdhary. Robust deep reinforcement learning with adversarial attacks. *Advances in Neural Information Processing Systems*, 2017.

[81] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[82] Aaditya Ramdas, Nicolás García Trillos, and Marco Cuturi. On wasserstein two-sample testing and related families of nonparametric tests. *Entropy*, 19(2):47, 2017.

[83] Martin Riedmiller. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *European conference on machine learning*, pages 317–328. Springer, 2005.

[84] Mark Rowland, Marc Bellemare, Will Dabney, Rémi Munos, and Yee Whye Teh. An analysis of categorical distributional reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 29–37. PMLR, 2018.

[85] Mark Rowland, Robert Dadashi, Saurabh Kumar, Rémi Munos, Marc G Bellemare, and Will Dabney. Statistics and samples in distributional reinforcement learning. *International Conference on Machine Learning (ICML)*, 2019.

[86] Mark Rowland, Rémi Munos, Mohammad Gheshlaghi Azar, Yunhao Tang, Georg Ostrovski, Anna Harutyunyan, Karl Tuyls, Marc G Bellemare, and Will Dabney. An analysis of quantile temporal-difference learning. *Journal of Machine Learning Research (JMLR)*, 2024.

[87] Mark Rowland, Yunhao Tang, Clare Lyle, Rémi Munos, Marc G Bellemare, and Will Dabney. The statistical benefits of quantile temporal-difference learning for value estimation. *International Conference on Machine Learning*, 2023.

[88] Ludger Rüschendorf and Wolfgang Thomsen. Closedness of sum spaces andthe generalized schrödinger problem. *Theory of Probability & Its Applications*, 42(3):483–494, 1998.

[89] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? *Advances in neural information processing systems*, 31, 2018.

[90] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[91] Qianli Shen, Yan Li, Haoming Jiang, Zhaoran Wang, and Tuo Zhao. Deep reinforcement learning with robust and smooth policy. In *International Conference on Machine Learning*, pages 8707–8718. PMLR, 2020.

[92] Chengchun Shi, Xiaoyu Wang, Shikai Luo, Hongtu Zhu, Jieping Ye, and Rui Song. Dynamic causal effects evaluation in a/b testing with a reinforcement learning framework. *Journal of the American Statistical Association*, pages 1–13, 2022.

[93] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[94] Rahul Singh, Qinsheng Zhang, and Yongxin Chen. Improving robustness via risk averse distributional reinforcement learning. *arXiv preprint arXiv:2005.00585*, 2020.

[95] Richard Sinkhorn. Diagonal equivalence to matrices with prescribed row and column sums. *The American Mathematical Monthly*, 74(4):402–405, 1967.

[96] Alexander L Strehl, Lihong Li, and Michael L Littman. Reinforcement learning in finite mdps: Pac analysis. *Journal of Machine Learning Research*, 10(11), 2009.

[97] Yang Sui, Yukun Huang, Hongtu Zhu, and Fan Zhou. Adversarial learning of distributional reinforcement learning. In *International Conference on Machine Learning*, pages 32783–32796. PMLR, 2023.

[98] Ke Sun, Bei Jiang, and Linglong Kong. How does return distribution in distributional reinforcement learning help optimization? *arXiv preprint arXiv:2209.14513*, 2022.

[99] Ke Sun, Yi Liu, Yingnan Zhao, Hengshuai Yao, Shangling Jui, and Linglong Kong. Exploring the training robustness of distributional reinforcement learning against noisy state observations. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2023.

[100] Ke Sun, Yingnan Zhao, Yi Liu, Shi Enze, Wang Yafei, Yan Xiaodong, Bei Jiang, and Linglong Kong. Interpreting distributional reinforcement

learning: A regularization perspective. *arXiv preprint arXiv:2110.03155*, 2021.

[101] Ke Sun, Yingnan Zhao, Yi Liu, Bei Jiang, and Linglong Kong. Distributional reinforcement learning via sinkhorn divergences. *arXiv preprint arXiv:2202.00769*, 2022.

[102] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An Introduction*. MIT press, 2018.

[103] Gábor J Székely. E-statistics: The energy of statistical samples. *Bowling Green State University, Department of Mathematics and Statistics Technical Report*, 3(05):1–18, 2003.

[104] Harm Van Seijen, Mehdi Fatemi, Joshua Romoff, Romain Laroche, Tavian Barnes, and Jeffrey Tsang. Hybrid reward architecture for reinforcement learning. *Advances in Neural Information Processing Systems*, 30, 2017.

[105] Kaiwen Wang, Owen Oertell, Alekh Agarwal, Nathan Kallus, and Wen Sun. More benefits of being distributional: Second-order bounds for reinforcement learning. *International Conference on Machine Learning*, 2024.

[106] Kaiwen Wang, Kevin Zhou, Runzhe Wu, Nathan Kallus, and Wen Sun. The benefits of being distributional: Small-loss bounds for reinforcement learning. *Advances in neural information processing systems*, 2023.

[107] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *International conference on machine learning*, pages 1995–2003. PMLR, 2016.

[108] Larry Wasserman. *All of nonparametric statistics*. Springer Science & Business Media, 2006.

[109] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[110] Li Kevin Wenliang, Grégoire Déletang, Matthew Aitchison, Marcus Hutter, Anian Ruoss, Arthur Gretton, and Mark Rowland. Distributional bellman operators over mean embeddings. *International Conference on Machine Learning*, 2024.

[111] Ronald J Williams and Jing Peng. Function optimization using connectionist reinforcement learning algorithms. *Connection Science*, 3(3):241–268, 1991.

[112] Eric Wong, Frank Schmidt, and Zico Kolter. Wasserstein adversarial examples via projected sinkhorn iterations. In *International Conference on Machine Learning*, pages 6808–6817. PMLR, 2019.

[113] Runzhe Wu, Masatoshi Uehara, and Wen Sun. Distributional offline policy evaluation with predictive error guarantees. *International Conference on Machine Learning*, 2023.

[114] Yi Xu, Yuanhong Xu, Qi Qian, Hao Li, and Rong Jin. Towards understanding label smoothing. *arXiv preprint arXiv:2006.11653*, 2020.

[115] Derek Yang, Li Zhao, Zichuan Lin, Tao Qin, Jiang Bian, and Tie-Yan Liu. Fully parameterized quantile function for distributional reinforcement learning. *Advances in neural information processing systems*, 32:6193–6202, 2019.

[116] Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Duane Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on observations. *Advances in Neural Information Processing Systems*, 2020.

[117] Pushi Zhang, Xiaoyu Chen, Li Zhao, Wei Xiong, Tao Qin, and Tie-Yan Liu. Distributional reinforcement learning for multi-dimensional reward functions. *Advances in Neural Information Processing Systems*, 34:1519–1529, 2021.

[118] Shangtong Zhang. Modularized implementation of deep rl algorithms in pytorch. `https://github.com/ShangtongZhang/DeepRL`, 2018.

[119] Shangtong Zhang and Hengshuai Yao. Quota: The quantile option architecture for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5797–5804, 2019.

[120] Fan Zhou, Jianing Wang, and Xingdong Feng. Non-crossing quantile regression for distributional reinforcement learning. *Advances in Neural Information Processing Systems*, 33, 2020.

[121] Florian Ziel. The energy distance for ensemble and scenario reduction. *arXiv preprint arXiv:2005.14670*, 2020.