Distributionally Robust Optimization for Reinforcement Learning

Jun Song

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

Doctor of Philosophy

University of Washington

2024

Reading Committee:

Chaoyue Zhao, Chair

Shuai Huang

Shan Liu

Program Authorized to Offer Degree:
Industrial & Systems Engineering

University of Washington

Abstract

Distributionally Robust Optimization for Reinforcement Learning

Jun Song

Chair of the Supervisory Committee:
Chaoyue Zhao
Industrial & Systems Engineering

Reinforcement learning (RL) has received remarkable success in many domains, including video games, board games, robotics and continuous control tasks. Despite the success and attention that RL has received during the past decades, it struggles with several issues that degrade its performance and lead to suboptimality. In model-based RL, the uncertainty in environment dynamics can significantly deteriorate the learnt agent's ability to recommend good actions. While in model-free RL, learnt agent's performance can be greatly affected by the restrictive parametric assumption on policy distribution.

In this dissertation, our goal is to utilize distributionally robust optimization (DRO) to overcome the above-mentioned limitations of RL, and to develop novel and practical RL algorithms with improved robustness and performance. To achieve the goal, we follow two main objectives. The first objective is to adopt DRO to add robustness to the uncertainty in the environment dynamics of the model-based RL. We propose a new Distributionally Robust Markov Decision Process (DRMDP) framework where the distribution of environment dynamics does not have predetermined parametric values, and we consider the worst-case probability distribution of these transition probabilities within a decision-dependent ambiguity set. The second objective is to utilize optimistic DRO to develop nonparametric policy optimization methods for the model-free RL. Since the policy learnt is not confined to the scope of parametric functions, this opens up the possibility of converging to a better optimality. Following this objective, we propose three different nonparametric policy optimization

frameworks, with Kullback–Leibler, Wasserstein and Sinkhorn constraints respectively to control the size of policy update. For each framework, we derive the closed-form policy update solution to the corresponding optimistic DRO problem using Lagrangian duality, and propose practical RL algorithms to perform the policy updates. We further improve the sample efficiency of the proposed nonparametric policy optimization frameworks, by incorporating human guidance through imitation learning techniques.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

First and foremost, I express my sincere gratitude to my remarkable PhD advisor, Chaoyue Zhao, for her invaluable guidance and mentorship throughout my doctoral journey. Chaoyue is an exceptional researcher, deeply knowledgeable in optimization under uncertainty, power systems and reinforcement learning. Her extensive knowledge has been instrumental in shaping my understanding of problem identification and formulation, solution methodologies, and the nuances of effective paper writing. Beyond her academic expertise, Chaoyue has proven to be an outstanding mentor, guiding me in selecting meaningful projects and providing the freedom to explore areas I am passionate about. Her ongoing support and advice regarding my post-PhD career have been a guiding force.

I extend my thanks to my dissertation committee members, Shuai Huang, Shan Liu, and Daniel Kirschen. Their expertise in healthcare, machine learning, and power systems has enriched the shaping and writing of my dissertation, offering valuable suggestions that have greatly contributed to its refinement.

Special acknowledgment goes to my undergraduate advisors, David Moore and Stuart Russell, for fostering my interest in scientific research. I am particularly grateful to David Moore for his teachings in statistical inference, sharing his personal PhD experiences, and guiding me in the PhD application process.

Gratitude also extends to colleagues and mentors from internships at Facebook/Meta Applied AI Research and Amazon SCOT. Mentors such as William Wei-Yu Tsai and Yudi Yang, along with colleagues like Wei Zhang, Xiaohan Wei, Qinyi Luo, Fan Lai, Jiachen Mao, Yuxi Hu, Xiuli Chao, German Riano and Muhong Zhang significantly contributed to my learning experience in recommendation systems, embedding techniques, and inventory optimization.

Collaborating with exceptional colleagues, including Lei Fan, Niao He, Xuegang Ban,

Shuai Huang, Xiuli Chao, Xiaodi Wu, Lijun Ding, Yudi Yang, William Wei-Yu Tsai, Qinyi Luo, Wei Zhang, Fan Lai, Jiachen Mao, Xiaohan Wei, Shuai Yang, Yuxi Hu, Congjing Zhang, William Yang and Jingxing Wang enriched my intellectual journey during my PhD. Engaging in discussions on quantum computing with Lei and Xiaodi, and receiving valuable proof techniques and writing assistance from Niao and Lijun, were particularly impactful aspects of this collaborative experience.

I express my appreciation to everyone at UW ISE for fostering an excellent graduate study environment. Special thanks to my friends and peers at UW, Xiaonan Sun, William Yang, Wengeng Pan, Yilun Xing, Serin Lee, Huasong Zhang, Weikun Hu, Luoyuan Chen, Yingqing Song, Yuan Wang, Yian Lin, Yi Yang, Xinyi Zhao and Yinsheng Wang. Their camaraderie has played a pivotal role in enriching the memorable years of my PhD, and I am truly grateful for their support and companionship.

Lastly, I extend heartfelt thanks to my parents for providing me with the best education. I am grateful to both my parents and parents-in-law for their unwavering support, and to my son for bringing joy into my life throughout my PhD journey. Above all, I dedicate this dissertation to my husband, X.S., whose encouragement, support, and love have been the cornerstone of my PhD journey. He is not only my motivation but the very reason behind the successful completion of this endeavor. His influence has intricately shaped the trajectory of both my academic and personal growth, raising me up to more than I can be.

DEDICATION

To my husband, X.S., who is my motivation and reason.

Chapter 1
INTRODUCTION

1.1 Motivation

Reinforcement learning (RL) has received remarkable success in many domains, including video games [170, 171], board games [108, 226], robotics [97, 98], and continuous control tasks [72, 221]. The core idea of RL is to learn the optimal actions of the agent performing in an uncertain interactive environment to maximize its expected cumulative reward. RL algorithms can be divided into two main categories: model-based RL [173] and model-free RL [170, 219, 222], depending on whether the agent uses the environment model (e.g., the actual or estimated environment dynamics) to find the optimal policy.

DRO for model-based RL

Though RL has received attention and success in the past decade, it often struggles with handling uncertainty. One type of uncertainty that causes concern for model-based RL is the uncertainty in environment dynamics. In model-based RL, agents are required to know the environment dynamics completely. For example, in a popular model-based RL approach called value iteration [25], the agent finds optimal actions by iteratively solving the Bellman equation that involves explicit values of the environment dynamics. However, in many real-world cases, the cost of interaction with the actual environment is high, and we can only take a limited number of real samples or use simulated samples. In either way, it can be challenging for model-based RL to estimate the environment dynamics accurately. The inaccurate estimation of environment dynamics can significantly deteriorate the learnt agent's ability to recommend good actions.

Two classical methods that are capable of addressing uncertainty are stochastic optimization [34, 225] and robust optimization [27, 28, 75]. Stochastic optimization assumes that the underlying uncertainty is a random variable following a known and fixed probability

distribution. However, in practice it can be hard to obtain a complete knowledge about the distribution of the uncertain parameter. On the contrary, robust optimization assumes that the decision maker has no distributional knowledge about the uncertain parameter despite its support. It then optimizes the objective function considering the worst-case realization of the uncertain parameter. However, robust optimization is often criticized for its overly conservativeness since the worst case rarely happens.

In this dissertation, we consider adopting distributionally robust optimization (DRO) to add robustness to the uncertainty in model-based RL. DRO finds a decision x that minimizes the expected cost under the most adversarial probability distribution of uncertain parameter, i.e., $\min_{x\in\mathcal{X}} \max_{\mathbb{P}\in\mathcal{D}} \mathbb{E}_{\mathbb{P}}[Q(x,\xi)]$. In DRO, the distribution $\mathbb{P}$ of uncertain parameter ξ is not precisely known but is assumed to fall into an ambiguity set $\mathcal{D}$. There are two main advantages of DRO over the classical stochastic and robust optimization. Firstly, unlike stochastic optimization, DRO allows the probability distribution of uncertain parameter to be ambiguous. That is, instead of fixing the distribution of uncertain parameter, we construct an ambiguity set of that distribution, which adds additional robustness to the uncertainty. Secondly, DRO optimizes the objective under the worst case probability distribution of the uncertain parameter. Unlike robust optimization, DRO considers the stochastic nature of uncertain parameters instead of totallying ignoring it. This allows the decision maker to find a solution that hedges against the uncertainty in model-based RL but not overly conservative as robust optimization.

DRO for model-free RL

Model-free RL does not model the environment, therefore is not affected by the uncertainty in environment dynamics. However, the performance of model-free RL can be greatly affected by the restrictive parametric assumption on policy distribution. Policy optimization [151, 219, 227] is a family of model-free RL algorithms that models the policy directly. In conventional policy optimization, the policy is usually represented as a particular parametric probability distribution $\pi_\theta(a|s) = P[a|s;\theta]$, such that the action a in state s is chosen stochastically following the policy π_θ controlled by parameter θ, for e.g., Gaussian [219, 222],

Beta [54] and Delta [151, 227] distribution functions. As indicated in [248], since parametric distributions are not convex in the distribution space, optimizing over such distributions results in local movements in the action space and thus leads to convergence to a sub-optimal solution. Also in practice, it is very difficult for machine learners to correctly predetermine the underlying distribution of the optimal policy. If we choose an incorrect distribution class for policy, we will never reach optimality as the optimal policy is excluded from exploration from the beginning.

In this dissertation, to overcome the limitation of parametric policy, we consider utilizing optimistic DRO to develop nonparametric policy optimization methods. Unlike the traditional DRO that considers the most adversarial probability distribution, the optimistic DRO aims to find the most optimistic (i.e., optimal) one to maximize the total reward function, i.e., $\max_{\pi' \in \mathcal{D}} \mathbb{E}_{a \sim \pi'}[A(s,a)]$. The optimistic DRO allows us to work on the space of policy distribution directly, and consider all admissible policies that are within the ambiguity sets with the goal of avoiding approximation errors. Since the policy learnt is not confined to the scope of parametric functions, this certainly opens up the possibility of converging to a better final policy. Besides, multiple types of ambiguity set $\mathcal{D}$ have been studied in previous DRO work, including moment based confidence set [65, 92, 314], ϕ divergences based confidence set [26, 117] and Wasserstein metric based confidence set [77, 309]. This enables us to explore different types of ambiguity sets to control the size of policy update in the proposed nonparametric policy optimization.

1.2 Research Objectives

The main focus of this dissertation is to utilize distributionally robust optimization (DRO) to develop novel and practical reinforcement learning (RL) algorithms with improved robustness and performance. Our motivations lead to the following objectives:

1. Model-based RL: Adopt DRO to add robustness to the uncertainty in environment dynamics

    - (Chapter 2) Distributionally Robust Markov Decision Process

      In model-based RL, the environment that the RL agent interacts with is formally

represented by Markov Decision Process. However, the lack of comprehensive knowledge on the environment dynamics calls for a more robust approach that is less dependent on assumptions that are prone to errors. Our main objective in this chapter is to build a new Distributionally Robust Markov Decision Process (DRMDP) framework where the distribution of transition dynamics does not have predetermined parametric values, and we consider the worst-case probability distribution of these transition probabilities within a decision-dependent ambiguity set. We also propose an efficient heuristic search model-based RL algorithm called Real-Time Dynamic Programming (RTDP) that is capable of solving the reformulated DRMDP model in an accurate, timely, and scalable manner. We evaluate the effectiveness of our approach on an epidemic control problem.

2. Model-free RL: Utilize optimistic DRO to develop nonparametric policy optimization algorithms

   - (Chapter 3) Nonparametric Kullback-Leibler Policy Optimization (KLPO)
     The majority of model-free RL methods cannot guarantee the stability and optimality of the learned policy, which is undesirable in safety-critical systems. In this chapter, we propose an innovative nonparametric policy optimization approach with Kullback-Leibler divergence constraint. Our approach ensures the stability of the policy update through trust region constraints, and improves optimality by removing the restrictive parametric assumption on policy representation that the majority of the RL literature adopts. We derive a closed-form expression of optimal policy update for each iteration and develop an efficient on-policy actor-critic algorithm to address the proposed constrained policy optimization problem. The effectiveness of our approach is demonstrated on a price-based demand response problem of the electricity market.
   - (Chapter 4) Wasserstein and Sinkhorn Policy Optimization (WPO, SPO)
     To stabilize policy optimization in model-free RL, trust-region methods based on Kullback-Leibler divergence are pervasively used. In this chapter, we exploit more

flexible metrics and examine two natural extensions of policy optimization with Wasserstein and Sinkhorn trust regions, namely Wasserstein policy optimization (WPO) and Sinkhorn policy optimization (SPO). Instead of restricting the policy to a parametric distribution class, we directly optimize the policy distribution and derive their closed-form policy updates based on the Lagrangian duality. Theoretically, we show that WPO guarantees a monotonic performance improvement, and SPO provably converges to WPO as the entropic regularizer diminishes. Moreover, we prove that with a decaying Lagrangian multiplier to the trust region constraint, both methods converge to the global optimality. Experiments across tabular domains, robotic locomotion, and continuous control tasks are conducted to demonstrate the performance improvement of the proposed WPO and SPO.

- (Chapter 5) Expert-Guided Wasserstein Policy Optimization (EGWPO)
  In general, model-free RL methods face the challenge of sample inefficiency, requiring a substantial amount of data to refine their policies. The WPO framework, as a model-free RL approach introduced in Chapter 4, is no exception to this limitation. In this chapter, we propose an enhancement to address the sample inefficiency of WPO by integrating human guidance through the application of Generative Adversarial Imitation Learning (GAIL) [113]. This results in the creation of the Expert-Guided WPO (EGWPO), combining the strengths of both WPO and GAIL. By merging expert knowledge with reinforcement signals, EGWPO aims to augment the learning efficiency of the original WPO. Additionally, by leveraging the nonparametric policy representation inherent in WPO, EGWPO seeks to mitigate the sub-optimality issue associated with the original GAIL. To evaluate the effectiveness of our approach, we apply it to a challenging whole-building HVAC control problem.
- (Chapter 6) Future Work

  – **Application of KLPO, WPO, and SPO to large-scale systems:** The innovative deep RL methodologies introduced in our previous research, namely

KLPO (Chapter 3), WPO, and SPO (Chapter 4), have shown remarkable robustness and superior performance compared to traditional RL. These methodologies offer substantial potential for applications across diverse domains, especially in addressing challenges characterized by large-scale complexities. Looking ahead, we anticipate employing the WPO approach to address various challenges within large-scale energy systems. For example, it shows great promise for enhancing energy efficiency in large-scale building energy management systems (BEMS). Additionally, it can efficiently determine optimal dynamic energy dispatch strategies within large-scale integrated energy systems (IES). Our future research will focus on these applications, aiming to comprehensively demonstrate the effectiveness of our proposed RL approach.

– **Extension of EGWPO for HVAC control in real-world smart homes:** The effectiveness of the EGWPO, as proposed in Chapter 5, has been successfully demonstrated in its application to the simulated EnergyPlus building system [127]. This involved guidance from a simulated PPO expert [223]. Looking ahead, our research endeavors will extend to deploying EGWPO for HVAC control in real-world smart homes located in Texas, where high summer temperatures present unique challenges. In our upcoming implementations, we will not only rely on simulated experts but also integrate guidance from actual human operators. This multi-faceted approach ensures the robust and adaptive application of EGWPO, enhancing its performance and applicability in real-world scenarios.

Chapter 2

DECISION-DEPENDENT DISTRIBUTIONALLY ROBUST MARKOV DECISION PROCESS METHOD IN DYNAMIC EPIDEMIC CONTROL

The environment that the model-based reinforcement learning (RL) agent interacts with is formally represented by Markov Decision Process (MDP). However, the lack of comprehensive knowledge on the environment dynamics calls for a more robust approach that is less dependent on assumptions that are prone to errors. Our main objective in this chapter is to build a new Distributionally Robust Markov Decision Process (DRMDP) framework where the distribution of transition dynamics does not have predetermined parametric values, and we consider the worst-case probability distribution of these transition probabilities within a decision-dependent ambiguity set. We also propose an efficient model-based heuristic RL algorithm called Real-Time Dynamic Programming (RTDP) that is capable of solving the reformulated DRMDP model in an accurate, timely, and scalable manner. We evaluate the effectiveness of our approach on an epidemic control problem, to find optimal vaccination and transmission-reducing intervention strategies to combat disease spreading according to the Susceptible-Exposed-Infectious-Recovered (SEIR) model. We compare the performance of the DRMDP model with RTDP to the standard MDP formulation and the results show that the DRMDP yields a lower proportion of individuals that are infected and susceptible to disease at a lower cost. We additionally perform a sensitivity analysis to observe which parameters affect DRMDP performance the most.

2.1 Introduction

Infectious disease is a major contributing factor to human morbidity and mortality and it has a devastating impact on both human welfare and the economy. The COVID-19 outbreak has caused millions of infections and deaths worldwide, and has led to a 3.2% global economy recession in 2020 [120]. The SARS outbreak in 2003 was another major epidemic that incurred a worldwide loss of about 50 billion dollars [132].

In this light, various mathematical models have been studied in the past few decades to understand the epidemic progression dynamics and to develop cost-effective interventions to control the spread of diseases, from the 2003 SARS outbreaks [e.g., 83, 283], to recent COVID-19 pandemic [e.g., 90, 96, 115, 163, 197, 231, 245, 313]. Although these approaches provide powerful insights into building strategies to reduce the impact of epidemics on a macroscopic level, they are not specifically structured to assist real-time public health decision-making through rapidly evolving epidemics.

To address this issue, Markov Decision Process (MDP) has been proposed as a method to develop dynamic control policies in the stochastic environment of infectious disease propagation [192, 212, 258, 282]. In MDP-based epidemic control models, one important component is to model the transition probabilities, which are to characterize disease spreading dynamics through a population. However, in practice, it is difficult to estimate the transition probabilities accurately. Inaccurate estimation of transition probabilities can significantly deteriorate a model's ability to recommend effective intervention strategies. However, there are a limited amount of MDP-based approaches that can cope with uncertainties in transition probabilities in epidemic control.

One general method that is capable of addressing uncertainty in MDP is robust MDP [121, 184, 185, 274], which considers the worst-case realization of the uncertain parameter, and it has been applied to tackling uncertainties in transition probabilities in epidemic control problems. For example, [30] formulate a discrete-time epidemic model as a robust MDP and solve it with parameter-wise robust reinforcement learning. In [30], some environmental parameters, which are used to model the transition probabilities, are considered as uncertain parameters, and the solutions are based on the worst-case scenario of environmental parameters. However, robust MDP is often criticized as its overly conservativeness, since the worst-case happens rarely.

In this paper, we utilize a distributionally robust optimization approach to address the MDP model for epidemic control. There are two main advantages for the proposed Distributionally Robust Markov Decision Process (DRMDP) model. First, the DRMDP model allows the probability distribution of transition probabilities to be ambiguous. That is, instead of fixing the probability distribution of transition probabilities, we construct an

ambiguity set of the distribution, to embrace the uncertainty of transition probabilities and consequently increase the model's robustness. Second, the DRMDP model minimizes the total expected health and economic loss under the worst-case probability distribution of transition probabilities. This allows the model to give a conservative solution that hedges against the uncertainty of transition probabilities but not overly conservative as robust MDP, since DRMDP considers the stochastic nature of the random parameters instead of totally ignoring it as robust MDP does.

Different shapes and sizes of the ambiguity set of the distribution will affect the computational complexity and the robustness of the final solution. Among the few existing DRMDP studies, there are two main approaches to ambiguity set construction. The two approaches are to construct the ambiguity set based on the moment information [53, 280, 287, 294], or distribution information [53, 188, 286]. In epidemic settings, limited information, especially in the early stage of epidemic evolution, makes it hard to construct an ambiguity set with density information. However, we can efficiently obtain estimates for the moments of epidemic model parameters in an Susceptible-Exposed-Infectious-Recovered (SEIR) model, so we utilize moment information to construct the ambiguity set in this paper.

In traditional distributionally robust optimization (DRO) settings, the ambiguity set of distributions is assumed to be exogenous, i.e., the distributions of random parameters are independent of what decisions have been made. However, during an epidemic, public health decisions can directly affect the spread of infectious diseases, which can be reflected by the transition probabilities. Therefore, to model the uncertainty of transition probabilities in dynamic epidemic control, we need to consider a decision-dependent (i.e., endogenous) ambiguity set.

Previous work using DRO under endogenous uncertainty mainly focuses on single-stage or two-stage settings [24, 162, 187, 210, 299]. For example, [24] reformulate a two-stage endogenous distributionally robust facility location problem as a mixed-integer programming (MIP) using a McCormick envelope and linear decision rules. DRO under endogenous uncertainty with a multistage setting has not been explored until recently [179, 295]. When extending endogenous DRO to a multistage setting, it remains challenging to design an efficient multistage algorithm. [179] model a differential equation based epidemic system as

partially observable DRMDP, where a first moment ambiguity set of transition-observation probabilities is employed. Though partially observable DRMDP is defined with a continuous belief, [179] is not directly applicable to the continuous state case. In our paper, we consider the DRMDP epidemic model with a general state space, and we adopt a modified Real-Time Dynamic Programming (RTDP) algorithm to efficiently solve for optimal policies based on the corresponding DRMDP problem.

To summarize, we utilize DRMDP under endogenous uncertainty to address the dynamic epidemic control problem. We highlight our contributions as follows:

1. We propose DRMDP formulations under a decision-dependent ambiguity set to model the epidemic control problem. The ambiguity set is robust to the setting where the true transition probabilities are unknown. We then derive a mixed-integer programming (MIP) reformulation of the DRMDP.

2. We develop a modified Real-Time Dynamic Programming (RTDP) method to efficiently compute optimal policies based on our DRMDP with a general state space. Since it computes an optimal partial epidemic control policy only for the states that are reachable from the initial state, this algorithm is significantly more efficient compared to the traditional value iteration algorithm which solves for the entire state space.

3. The numerical experiments verify that, as compared to the classic MDP, our DRMDP algorithm finds a better policy (i.e., a policy with lower total discounted health and economic loss) under misspecified distributions of transition probabilities. Furthermore, our results show that the DRMDP is more effective than the classic MDP in controlling the number of infectives.

2.2 Preliminaries and Problem Setup

Markov Decision Processes (MDP) are commonly used to solve sequential decision-making problems. In general, a finite-horizon discounted Markov Decision Process (MDP) is defined as a tuple $< T, \lambda, \mathcal{S}, \mathcal{A}, \boldsymbol{p}, \boldsymbol{r} >$, where T is the time horizon, λ is the discount factor, $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\boldsymbol{p}$ is the transition probability between states depending on

the action taken, and $\boldsymbol{r}$ is the reward associated with the state and the action taken. In this section, we model our problem as a continuous-state MDP model. We present the different MDP components in the context of the dynamic epidemic control problem below:

- **State**: The Susceptible-Exposed-Infectious-Recovered (SEIR) model is a classic model to describe the influenza epidemic, but can be generalized model to any infectious disease. The state of disease spread at stage t is defined as $s^t = (p_S(t), p_E(t), p_I(t))$, where $p_S(t)$ denotes the proportion of susceptible individuals in the population at stage t, $p_E(t)$ denotes the percentage of exposed individuals at stage t, $p_I(t)$ denotes the percentage of infectious individuals at stage t. The state space can be defined as $\mathcal{S} = \{(p_S, p_E, p_I) \in \mathbb{R}^3_+ \mid p_S + p_E + p_I \leq 1\}$. We assume the population size N stays constant throughout the epidemic, and that susceptible individuals immediately gain lifelong immunity from vaccination.

- **Action**: We consider two categories of actions for controlling the spread of influenza: vaccination and transmission-reducing intervention. At each stage t, the decision maker will decide the proportion of susceptibles to vaccinate. $y_V(t) \in \{0, \ldots, L\}$ represents the scale of susceptibles to vaccinate at stage t, where $y_V(t) = i$ corresponds to vaccinating $\frac{i}{L} \times 100\%$ susceptibles at stage t. The transmission-reducing interventions are interventions that can be employed or lifted during the pandemic to reduce transmission, such as social distancing, wearing face masks, quarantining, closing schools etc. At each stage t, $y_R(t) \in \{0, \ldots, M\}$ represents the scale of transmission-reducing interventions based on its strength, i.e., 0 represents no transmission-reducing action, and M represents the strongest transmission-reducing action. The action at stage t can be defined as $a^t = (y_V(t), y_R(t))$, and the action space can be defined as $\mathcal{A} = \{(y_V, y_R) \in \mathbb{N}^2 | y_V \leq L, y_R \leq M\}$.

- **Transition Probabilities**: The transition probability $p_{as}(s') = P(s'|s, a)$ represents the probability of transitioning from the state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$ in stage t to the next state $s' \in \mathcal{S}$ in stage $t + 1$. Its value depends on the stochastic epidemiological processes and on control measures implemented by the policy maker. We define the

vector $\boldsymbol{p}_{as} = (p_{as}(s'), s' \in \mathcal{S})^T$ as a collection of transition probabilities from the state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$ to each $s' \in \mathcal{S}$. In other words, the s' component of $\boldsymbol{p}_{as}$ is simply $p_{as}(s')$. Note that because S is a continuous state space, $\boldsymbol{p}_{as}$ is an infinite-dimensional vector.

We will derive formulas for the transition probabilities using the underlying structure of the SEIR model. Let $n_B(t)$ denote the number of susceptible individuals who become exposed during time t, $n_C(t)$ denote the number of exposed individuals who become infectious during time t and $n_D(t)$ denote the number of infectious individuals who recover during time t. The discrete-time stochastic SEIR model in [144] specifies the following relationships:

$$\begin{aligned} Np_S(t+1) &= Np_S(t)(1 - \frac{y_V(t)}{L}) - n_B(t), \\ Np_E(t+1) &= Np_E(t) + n_B(t) - n_C(t), \\ Np_I(t+1) &= Np_I(t) + n_C(t) - n_D(t), \end{aligned}$$

with $n_B(t) \sim \text{Bin}(Np_S(t) \times (1 - \frac{y_V(t)}{L}), \phi(t))$, $n_C(t) \sim \text{Bin}(Np_E(t), \rho_C)$, and $n_D(t) \sim \text{Bin}(Np_I(t), \rho_D)$, where $\text{Bin}(.,.)$ denotes the binomial distribution, $\phi(t) = 1 - \exp(-(1 - \alpha(t))\mu p_I(t)\beta)$, $\rho_C = 1 - \exp(-l_C)$, and $\rho_D = 1 - \exp(-l_D)$. The parameter μ denotes the contact rate when no transmission reduction method is used, and the parameter $\alpha(t)$ denotes the fraction reduction in the contact rate from transmission-reduction intervention. We assume that $\alpha(t) = \alpha_0 y_R(t)/M$, where α_0 represents the maximum possible fractional reduction in the contact rate, which means $\alpha(t)$ has a linear relationship with the scale of the transmission-reducing method. The parameter β denotes the probability that a susceptible individual becomes infected upon contact with an infectious individual. Lastly, the parameters l_C, l_D denote the mean incubation period and the mean infectious period respectively.

The nominal transition probability from s^t to s' given a^t can be expressed as:

$$\begin{aligned} &p^0_{a^t s^t}(s') = P(s' = (p_S, p_E, p_I) | s^t = (p_S(t), p_E(t), p_I(t)), a^t = (y_V(t), y_R(t))) \\ &= P(n_B(t) = Np_S(t) \times (1 - \frac{y_V(t)}{L}) - Np_S) \end{aligned}$$

$$\times P(n_C(t) = Np_S(t) \times (1 - \frac{y_V(t)}{L}) + Np_E(t) - Np_S - Np_E)$$
$$\times P(n_D(t) = Np_S(t) \times (1 - \frac{y_V(t)}{L}) + Np_E(t) + Np_I(t) - Np_S - Np_E - Np_I).$$

- **Rewards**: To represent the economic and health impact in each state for each action, we use reward matrices $\boldsymbol{r} \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}$. In this section, we will derive an expression for the components in the nominal reward matrix. We define the reward at stage t to consist of the following components:

  1. $c_V(t) := Q\frac{y_V(t)}{L}Np_S(t)$ is cost of vaccinations at stage t, where Q is the unit price of vaccine.
  2. $c_R(t) := k_R y_R(t)$ is cost of implementing transmission-reduction method at stage t, where k_R is a positive multiplier.
  3. $c_I(t) := W\mathbb{E}[Np_I(t)+n_C(t)-n_D(t)]$ is the expected total health loss and treatment cost due to infections at stage t, where W is the health loss plus the treatment cost associated with a single infection. This formula can be seen as the cost for the expected number of infected people in the next time period.

  The nominal reward can then be expressed as:

$$r^0_{a^t s^t} = -c_V(t) - c_R(t) - c_I(t). \tag{2.1}$$

In this paper, we propose and solve a distributionally robust policy under endogenous transition probability uncertainty. That is, the distribution μ_{as} of the transition probability $\boldsymbol{p}_{as}$ is not precisely known but is assumed to belong to an ambiguity set $\mathcal{D}_{as} \subseteq \mathcal{P}(\Delta(\mathcal{S}))$, where $\Delta(\mathcal{S})$ is the probability simplex of set $\mathcal{S}$, and $\mathcal{P}(\Delta(\mathcal{S}))$ represents the set of all probability distributions with support $\Delta(\mathcal{S})$. The objective is to find a policy $\pi : \mathcal{S} \to \mathcal{A}$ that determines corresponding vaccination and transmission-reducing intervention actions for different proportions of susceptible, exposed, and infectious individuals for each stage t. In this paper, to enhance the robustness of the model, we aim to find the optimal policy to maximize reward under the worst-case distribution $\mu_{as} \in \mathcal{D}_{as}$. That is, we consider a dynamic adversarial game between the public health decision-maker and nature, where at each stage,

the public health decision-maker selects a epidemic control action $a \in \mathcal{A}$ to maximize total expected future reward while nature selects the distribution μ_{as} of $\boldsymbol{p}_{as}$ to minimize total expected future reward given the decision maker's action a. Note here $\mathcal{D}_{as}$ is an endogenous (or, decision-dependent) ambiguity set, i.e., it is depending on the action a at state s. Most of the literature assumes the ambiguity set is exogenous (decision independent), however, this is not a reasonable assumption in our setting because control actions taken in each stage will significantly affect the spread of infectious diseases, and therefore affect the transmission probabilities. For example, if we take the action to vaccinate as many susceptible individuals as possible, close school, and employ social distancing, this should decrease the probability of entering a state with a higher proportion of infective individuals.

We will now derive an expression for the expected total reward in the adversarial game setting. Let $h^t = (s^1, a^1, \mu_{a^1 s^1}, \ldots, s^{t-1}, a^{t-1}, \mu_{a^{t-1} s^{t-1}}, s^t)$ be the history of states and actions until stage t and H^t denote the set of all histories until stage t. The set of all history-dependent control policies for the decision maker is denoted by $\Pi = \{\pi = (\pi^1, \ldots, \pi^{T-1}) \mid \pi^t : H^t \to \mathcal{A},\ \forall t \in \{1, \ldots, T-1\}\}$. Let $\tilde{h}^t = (s^1, a^1, \mu_{a^1 s^1}, \ldots, s^{t-1}, a^{t-1},\ \mu_{a^{t-1} s^{t-1}}, s^t, a^t)$ be the extended history until stage t, with action a_t, and $\tilde{H}^t$ denote the set of all extended histories until stage t. The set of nature's admissible policies are defined as $\Gamma = \{\gamma = (\gamma^1, \ldots, \gamma^{T-1}) \mid \gamma^t : \tilde{H}^t \to \mathcal{D}_{a^t s^t},\ \forall t \in \{1, \ldots, T-1\}\}$. Given a strategy pair $(\pi, \gamma) \in (\Pi \times \Gamma)$, we define the expected total rewards as

$$R_s[\pi, \gamma] = \mathbb{E}_\gamma \left[\sum_{t=1}^{T-1} \left(\lambda^{t-1} r_{a^t s^t} \right) + \lambda^{T-1} \bar{R}(s^T) \mid s^1 = s \right], \tag{2.2}$$

where $a^t = \pi^t(h^t)$, $\mathbb{E}_\gamma$ denotes the expectation with respect to the probability measure induced by nature's strategy γ, s is the initial state and $\bar{R}$ is a bounded terminal reward function. To obtain the values of the instantaneous reward, $r_{a^t s^t}$, we use linear regression with the nominal reward $r^0_{a^t s^t}$ defined in (2.1). We describe this procedure in more detail in Section 2.4.

Our problem can be modeled as a zero-sum two-player dynamic game problem, where the public health decision maker's objective improves if and only if nature's objective gets worse. Thus, the desired epidemic control policy can be obtained by solving the following

optimization problem:

$$\max_{\pi \in \Pi} \min_{\gamma \in \Gamma} R_s[\pi, \gamma]. \tag{2.3}$$

2.3 Discretization

In this section, we describe the process of discretizing the state space and transition probabilities for our DRMDP model.

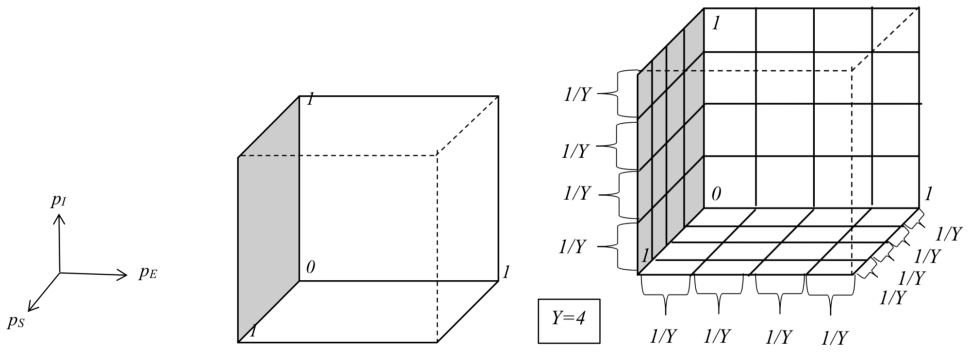
To discretize the state space, we consider the cube $H_{\mathcal{S}}$, which contains the continuous state space $\mathcal{S}$. The cube $H_{\mathcal{S}}$ is defined as $H_{\mathcal{S}} := \{(p_S, p_E, p_I) \in \mathbb{R}^3_+ \mid p_S \leq 1, p_E \leq 1, p_I \leq 1\}$. It is important to note that by construction, $H_{\mathcal{S}}$ may contain some combinations of $(\tilde{p}_S, \tilde{p}_E, \tilde{p}_I)$ that are not in the actual state space $\mathcal{S}$. However, defining $H_{\mathcal{S}}$ as a cube allows us to easily partition it into smaller cubes with equal volume. Each smaller cube has an edge length of $\frac{1}{Y}$, resulting in a total of Y^3 equal-volume cubes.

To further partition each small cube into simplexes, we adopt the Kuhn triangulation method [175]. Kuhn triangulation is commonly used in the discretization of MDPs due to its efficient computation of interpolation weights [60, 178]. By applying Kuhn triangulation, we divide each small cube into six equal-volume simplexes.

As a result of this discretization process, the state space after discretization is represented by $\tilde{\mathcal{S}} = \{(\tilde{p}_S, \tilde{p}_E, \tilde{p}_I) \mid \tilde{p}_S, \tilde{p}_E, \tilde{p}_I \in \{\frac{0}{Y}, \frac{1}{Y}, \ldots, \frac{Y}{Y}\}\}$. For simplicity, we use the notation $\xi_1, \ldots, \xi_{|\tilde{\mathcal{S}}|}$ to represent the discrete states in $\tilde{\mathcal{S}}$. It is important to note that any state $s \in \mathcal{S} \backslash \tilde{\mathcal{S}}$ is included in exactly one simplex, and every corner state $\xi \in \tilde{\mathcal{S}}$ is included in at least one simplex.

We denote the union of simplexes that include the corner state $\xi \in \tilde{\mathcal{S}}$ as $U(\xi)$, and the simplex that includes state $s \in \mathcal{S} \backslash \tilde{\mathcal{S}}$ as $B(s)$. Moreover, every state within a simplex B can be expressed as a convex combination of the corner states of that simplex. The set of corner states for simplex B is denoted as $C(B)$.

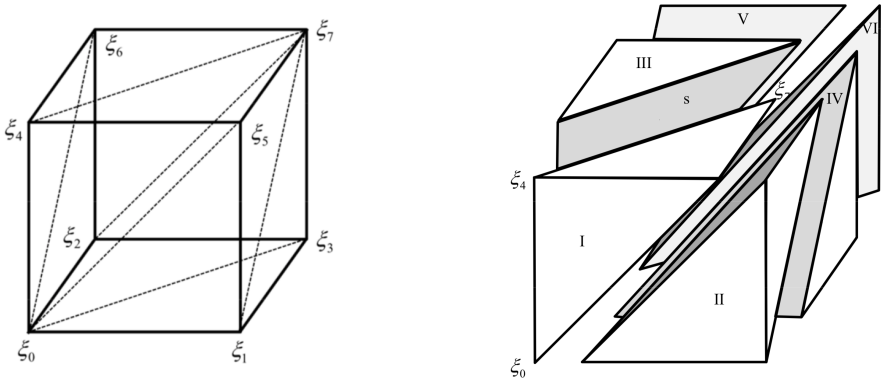
Figure 2.1a shows the representation of the state space as a unit cube, and Figure 2.1b shows the partitioning of the state space for the case when $Y = 4$, which shows the unit cube split into 64 equal volume cubes in this example.



(a) (b)

Figure 2.1: Partition of State Space where $Y = 4$

Figure 2.2 represents the Kuhn triangulation of one of the smaller cubes shown in Figure 2.1b. Figure 2.2a shows the lines in which each simplex is divided by. And Figure 2.2b shows the separated 6 simplexes. From the example in Figure 2.2b, we can see that $B(s) = III$, $U(\xi_0) = \{I, II, III, IV, V, VI\}$, $U(\xi_1) = \{II, IV\}$, $C(I) = \{\xi_0, \xi_4, \xi_5, \xi_7\}$, and $C(II) = \{\xi_0, \xi_1, \xi_5, \xi_7\}$.



(a) (b)

Figure 2.2: Kuhn Triangulation [186]

Therefore, the convex combination that represents $s \in B$ is

$$s = \sum_{\xi \in C(B(s))} \theta_s^\xi \xi, \text{where} \sum_{\xi \in C(B(s))} \theta_s^\xi = 1.$$

θ_s^ξ is the weight of each ξ in the convex combination. Based on the state space discretization, we define the discrete nominal reward and discrete terminal reward to be: $\tilde{r}^0_{a\xi_i} = r^0_{a\xi_i}$ if $\xi_i \in \mathcal{S}$, and $\tilde{r}^0_{a\xi_i} = 0$ otherwise; $\tilde{q}_R(\xi_i) = \bar{R}(\xi_i)$ if $\xi_i \in \mathcal{S}$, and $\tilde{q}_R(\xi_i) = 0$ otherwise, respectively. And we define the discrete nominal transition probability between $\xi_i, \xi_j \in \tilde{\mathcal{S}}$:

- If $\xi_i \in \mathcal{S}$:

$$\tilde{p}^0_{a\xi_i}(\xi_j) := \int_{s \in U(\xi_j) \cap \mathcal{S}} \theta_s^{\xi_j} p^0_{a\xi_i}(s) ds. \tag{2.4}$$

- If $\xi_i \notin \mathcal{S}$: $\tilde{p}^0_{a\xi_i}(\xi_j) = 1$ if $\xi_i = \xi_j$; $\tilde{p}^0_{a\xi_i}(\xi_j) = 0$ if $\xi_i \neq \xi_j$.

We note that the discrete nominal transition probability is well defined since $\sum_{\xi \in \tilde{\mathcal{S}}} \tilde{p}^0_{a\xi_i}(\xi) = \sum_{\xi \in \tilde{\mathcal{S}}} \int_{s \in U(\xi) \cap \mathcal{S}} \theta_s^\xi p^0_{a\xi_i}(s) ds = \int_{s \in \mathcal{S}} \sum_{\xi \in C(B(s))} \theta_s^\xi p^0_{a\xi_i}(s) ds = \int_{s \in \mathcal{S}} p^0_{a\xi_i}(s) ds = 1$.

2.4 Model Reformulation

In this section, we consider DRMDP formulations with a decision-dependent ambiguity set, which can handle the case where only limited information of the epidemic statistics is available. To solve the decision-dependent uncertainty, we adopt the approach used in [24, 295] and reformulate the distributionally robust Bellman equation as a mixed integer programming (MIP) using McCormick or unary envelopes with linear decision rules.

2.4.1 Distributionally Robust Bellman Equation

We first rewrite the expected reward-to-go function (2.3) by replacing the value function $R_\xi[\pi, \gamma]$ by (2.2):

$$V^t(\xi) = \max_{\pi \in \Pi} \min_{\gamma \in \Gamma} \mathbb{E}_\gamma \left[\sum_{i=t}^{T-1} \lambda^{i-t} \tilde{r}_{a^i \xi^i} + \lambda^{T-t} \tilde{q}_R(\xi^T) \mid \xi^t = \xi \right].$$

We assume that without loss of generality, maximizing the expected reward under worst admissible transition probability distribution is equivalent to solving the following distributionally

robust Bellman equations:

$$V^t(\xi) = \max_{a\in\mathcal{A}} \min_{\mu_{a\xi}\in\mathcal{D}_{a\xi}} \mathbb{E}_{\boldsymbol{p}_{a\xi}\sim\mu_{a\xi}}[\tilde{r}_{a\xi} + \lambda \boldsymbol{p}_{a\xi}^{\mathrm{T}} \boldsymbol{V}^{t+1}], \tag{2.5}$$

$$Q^t(\xi, a) = \min_{\mu_{a\xi}\in\mathcal{D}_{a\xi}} \mathbb{E}_{\boldsymbol{p}_{a\xi}\sim\mu_{a\xi}}[\tilde{r}_{a\xi} + \lambda \boldsymbol{p}_{a\xi}^{\mathrm{T}} \boldsymbol{V}^{t+1}],$$

where $V^t(.)$ and $Q^t(.,.)$ represent the distributionally robust state-value function and action-value function respectively, and $\boldsymbol{V}^{t+1} = (V^{t+1}(\xi'), \xi' \in \tilde{\mathcal{S}})^T$. The optimal action for the decision maker at stage t then is:

$$a^* = \operatorname*{argmax}_{a\in\mathcal{A}} Q^t(\xi, a),$$

whereas the optimal distribution chosen by the nature is:

$$\mu^* = \operatorname*{argmin}_{\mu_{a\xi}\in\mathcal{D}_{a\xi}} \mathbb{E}_{\boldsymbol{p}_{a\xi}\sim\mu_{a\xi}}[\tilde{r}_{a\xi} + \lambda \boldsymbol{p}_{a\xi}^{\mathrm{T}} \boldsymbol{V}^{t+1}].$$

2.4.2 Ambiguity Set with First Moment Information

In practice, it is often the case that one has limited information about the transition probability distribution $\mu_{a\xi}$. In such situations, one can rely on the estimates of transition probability based on historical records or expertise domain knowledge. However, even if the mean value of transition probabilities can be estimated, the true distribution is still ambiguous. Therefore, we construct an ambiguity set to handle this uncertainty. We consider such an ambiguity set where the mean vector of the transition probabilities is restricted by decision-dependent bounds, and the true distribution of the transition probabilities can run adversely within the ambiguity set. The ambiguity set is constructed as follows:

$$\mathcal{D}_{a\xi} := \Big\{ \mu_{a\xi} \in \mathcal{P}(\Delta(\tilde{\mathcal{S}})) \ : \boldsymbol{p}_{a\xi} \sim \mu_{a\xi}, \tag{2.6a}$$

$$\tilde{\boldsymbol{\eta}}_{a\xi}^L \leq \mathbb{E}[\boldsymbol{p}_{a\xi}] \leq \tilde{\boldsymbol{\eta}}_{a\xi}^U \Big\}, \tag{2.6b}$$

where $\tilde{\boldsymbol{\eta}}_{a\xi}^L$ and $\tilde{\boldsymbol{\eta}}_{a\xi}^U$ are the decision dependent lower and upper bounds, respectively.

2.4.3 Reformulation of Bellman Equations

In order to reformulate the Bellman equation (2.5) into tractable formulations, under the setting of ambiguity set shown in (2.6), we relax the hard constraint (2.6b) into a soft

constraint and adjust the objective function by penalizing constraint violations. Then (2.5) with the ambiguity set in (2.6) can be reformulated as:

$$\min_{\mu_{a\xi}, \boldsymbol{x}\geq 0} \tilde{r}_{a\xi} + \int_{\Delta(\tilde{\mathcal{S}})} \lambda \boldsymbol{p}_{a\xi}^{\mathrm{T}} \boldsymbol{V}^{t+1} d\mu_{a\xi}(\boldsymbol{p}_{a\xi}) + k\mathbf{1}^{\mathrm{T}}\boldsymbol{x} \tag{2.7a}$$

$$s.t. \quad \int_{\Delta(\tilde{\mathcal{S}})} d\mu_{a\xi}(\boldsymbol{p}_{a\xi}) = 1, \tag{2.7b}$$

$$\int_{\Delta(\tilde{\mathcal{S}})} \boldsymbol{p}_{a\xi} d\mu_{a\xi}(\boldsymbol{p}_{a\xi}) - \tilde{\boldsymbol{\eta}}_{a\xi}^{U} \leq \boldsymbol{x}, \tag{2.7c}$$

$$\tilde{\boldsymbol{\eta}}_{a\xi}^{L} - \int_{\Delta(\tilde{\mathcal{S}})} \boldsymbol{p}_{a\xi} d\mu_{a\xi}(\boldsymbol{p}_{a\xi}) \leq \boldsymbol{x}, \tag{2.7d}$$

where the objective (2.7a) consists of the initial reward, the expected future reward over probability distribution $\mu_{a\xi}$ and a penalty term $k\mathbf{1}^{\mathrm{T}}\boldsymbol{x}$. Here, k represents a user-specified penalty coefficient to penalize the violation of constraint (2.6b). (2.7b) represents the constraint (2.6a), (2.7c) and (2.7d) are relaxation of (2.6b). When $k \to \infty$, (2.7) will be equivalent to (2.5) as x will be 0, i.e., no violation for constraint (2.6b).

By utilizing the Lagrangian dualization approach, we reformulate the (2.7), and we show the reformulation in the following theorem:

Theorem 1. *If for any $a \in \mathcal{A}$, the ambiguity set defined in* (2.6) *is nonempty, then* (2.7) *can be reformulated as:*

$$\begin{aligned} V^t(\xi) = \max_{a\in\mathcal{A}, \boldsymbol{w}, \boldsymbol{u}, q} \; & \tilde{r}_{a\xi} + q - \boldsymbol{w}^T \tilde{\boldsymbol{\eta}}_{a\xi}^{U} + \boldsymbol{u}^T \tilde{\boldsymbol{\eta}}_{a\xi}^{L} \\ s.t. \quad & q\mathbf{1} \leq \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}, \\ & \boldsymbol{w} + \boldsymbol{u} \leq k\mathbf{1}, \\ & \boldsymbol{w}, \boldsymbol{u} \geq \mathbf{0}. \end{aligned} \tag{2.8}$$

Here, $\mathbf{1}, \boldsymbol{V}^{t+1}, \boldsymbol{w}, \boldsymbol{u}, \tilde{\boldsymbol{\eta}}_{a\xi}^{U}$ and $\tilde{\boldsymbol{\eta}}_{a\xi}^{L}$ are vectors with length $|\tilde{\mathcal{S}}|$. Therefore, the notation $\boldsymbol{w}^T \tilde{\boldsymbol{\eta}}_{a\xi}^{U}$ can be understood as

$$\boldsymbol{w}^T \tilde{\boldsymbol{\eta}}_{a\xi}^{U} := \sum_{\xi' \in \tilde{\mathcal{S}}} w(\xi') \tilde{\eta}_{a\xi}^{U}(\xi'),$$

where $w(\xi')$ and $\tilde{\eta}_{a\xi}^{U}(\xi')$ represent the value of $\boldsymbol{w}^T$ and $\tilde{\boldsymbol{\eta}}_{a\xi}^{U}$ for state ξ', respectively. $\boldsymbol{u}^T \tilde{\boldsymbol{\eta}}_{a\xi}^{L}$ is defined similarly. The proof of this theorem is provided in the online supplement.

To approximate (2.8), we adopt the linear decision rule [114], that is, we assume that $\tilde{\boldsymbol{\eta}}^U_{as}$, $\tilde{\boldsymbol{\eta}}^L_{as}$ and $\tilde{r}_{as}$ are linear functions of a:

$$\tilde{\eta}^U_{a\xi}(\xi') = \rho^\xi_0(\xi') + \sum_{i=1}^{N_a} \rho^\xi_i(\xi') a_i, \quad \forall \xi' \in \tilde{\mathcal{S}}, \tag{2.9}$$

$$\tilde{\eta}^L_{a\xi}(\xi') = \sigma^\xi_0(\xi') + \sum_{i=1}^{N_a} \sigma^\xi_i(\xi') a_i, \quad \forall \xi' \in \tilde{\mathcal{S}}, \tag{2.10}$$

$$\tilde{r}_{a\xi} = \epsilon^\xi_0 + \sum_{i=1}^{N_a} \epsilon^\xi_i a_i, \tag{2.11}$$

where a_i represents the i-th dimension of action a, and N_a is the total dimension of each a in $\mathcal{A}$ ($N_a = 2$ in the case where we consider the two types of actions: vaccination and transmission-reduction intervention). To obtain the coefficients $\boldsymbol{\epsilon}^\xi$ in (2.11), we apply linear regression with training data $\{\alpha_1, \ldots, \alpha_n\}$ and target values $\{\tilde{r}^0_{\alpha_1\xi}, \ldots, \tilde{r}^0_{\alpha_n\xi}\}$, where $\tilde{r}^0_{a\xi}$ is the discrete nominal reward. Similarly, we apply linear regression with training data $\{\alpha_1, \ldots, \alpha_n\}$ and target values $\{\tilde{\boldsymbol{\eta}}^U_{\alpha_1\xi}, \ldots, \tilde{\boldsymbol{\eta}}^U_{\alpha_n\xi}\}$ and $\{\tilde{\boldsymbol{\eta}}^L_{\alpha_1\xi}, \ldots, \tilde{\boldsymbol{\eta}}^L_{\alpha_n\xi}\}$ to obtain coefficients $\boldsymbol{\rho}^\xi$ and $\boldsymbol{\sigma}^\xi$ in (2.9) and (2.10) respectively. Here we set the target values $\tilde{\boldsymbol{\eta}}^U_{\alpha_i\xi} = \tilde{\boldsymbol{p}}^0_{\alpha_i\xi} + \boldsymbol{\delta}$ and $\tilde{\boldsymbol{\eta}}^L_{\alpha_i\xi} = \tilde{\boldsymbol{p}}^0_{\alpha_i\xi} - \boldsymbol{\delta}$ for $i = 1, \ldots, n$, where $\boldsymbol{\delta} > 0$ is a pre-defined error bound and $\tilde{\boldsymbol{p}}^0_{a\xi}$ is the discrete nominal transition probability defined in (2.4).

Thus, $\boldsymbol{w}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}^U_{a\xi}$, and $\boldsymbol{u}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}^L_{a\xi}$ (which are terms of the objective of (2.8)) can be expressed as:

$$\boldsymbol{w}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}^U_{a\xi} = \sum_{\xi' \in \tilde{\mathcal{S}}} \Big(\rho^\xi_0(\xi') + \sum_{i=1}^{N_a} \rho^\xi_i(\xi') a_i \Big) w(\xi'),$$

$$\boldsymbol{u}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}^L_{a\xi} = \sum_{\xi' \in \tilde{\mathcal{S}}} \Big(\sigma^\xi_0(\xi') + \sum_{i=1}^{N_a} \sigma^\xi_i(\xi') a_i \Big) u(\xi').$$

To linearize the bilinear terms $a_i w(\xi')$ and $a_i u(\xi')$, we adopt two different methods: McCormick envelope relaxation and exact unary expansion. The corresponding mixed integer programming (MIP) formulations are presented in Corollary 1 and Corollary 2, respectively, which we introduce below.

McCormick Envelope Relaxation: We replace the bilinear terms $a_i w(\xi')$ with $m^0_i(\xi')$ and $a_i u(\xi')$ with $m^1_i(\xi')$ by using McCormick envelopes [167] $M^0_i(\xi')$ and $M^1_i(\xi')$ respectively for all $i = 1, \ldots, N_a$, $\xi' \in \tilde{\mathcal{S}}$. We also utilize upper and lower bounds for decision variables

$a_i, w(\xi')$, and $u(\xi')$, which we denote as $\bar{a}_i, \bar{w}_{\xi'}, \bar{u}_{\xi'}$ and $\underline{a}_i, \underline{w}_{\xi'}, \underline{u}_{\xi'}$, respectively. In our setting, it is clear that $\bar{a}_1 = L, \bar{a}_2 = M, \underline{a}_1 = \underline{a}_2 = 1, \bar{w}_{\xi'} = \bar{u}_{\xi'} = k$, and $\underline{w}_{\xi'} = \underline{u}_{\xi'} = 0 \ \ \forall \xi' \in \tilde{\mathcal{S}}$, which leads us to the following corollary:

Corollary 1. *If for any $a \in \mathcal{A}$, the ambiguity set defined in* (2.6) *is nonempty, then the Bellman equation* (2.5) *can be approximated as the following MIP formulation:*

$$V^t(\xi) = \max_{a \in \mathcal{A}, \boldsymbol{w}, \boldsymbol{u}, q} \tilde{r}_{a\xi} + q - \sum_{\xi' \in \tilde{\mathcal{S}}} \left(\rho_0^{\xi}(\xi') w(\xi') + \sum_{i=1}^{N_a} \rho_i^{\xi}(\xi') m_i^0(\xi') \right) + \sum_{\xi' \in \tilde{\mathcal{S}}} \left(\sigma_0^{\xi}(\xi') u(\xi') + \sum_{i=1}^{N_a} \sigma_i^{\xi}(\xi') m_i^1(\xi') \right) \tag{2.12a}$$

$$s.t. \quad q\mathbf{1} \leq \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}, \tag{2.12b}$$

$$\boldsymbol{w} + \boldsymbol{u} \leq k\mathbf{1}, \tag{2.12c}$$

$$\boldsymbol{w}, \boldsymbol{u} \geq \mathbf{0}, \tag{2.12d}$$

$$(m_i^0(\xi'), a_i, w(\xi')) \in M_i^0(\xi'), \quad \forall i \in [N_a], \xi' \in \tilde{\mathcal{S}}, \tag{2.12e}$$

$$(m_i^1(\xi'), a_i, u(\xi')) \in M_i^1(\xi'), \quad \forall i \in [N_a], \xi' \in \tilde{\mathcal{S}}, \tag{2.12f}$$

where $[N_a]$ denotes the set $\{1, \ldots, N_a\}$, and

$$\begin{aligned}
M_i^0(\xi') = \{&(m_i^0(\xi'), a_i, w(\xi')) : \\
&m_i^0(\xi') \geq \underline{a}_i w(\xi') + a_i \underline{w}_{\xi'} - \underline{a}_i \underline{w}_{\xi'}, m_i^0(\xi') \geq \bar{a}_i w(\xi') + a_i \bar{w}_{\xi'} - \bar{a}_i \bar{w}_{\xi'}, \\
&m_i^0(\xi') \leq \bar{a}_i w(\xi') + a_i \underline{w}_{\xi'} - \bar{a}_i \underline{w}_{\xi'}, m_i^0(\xi') \leq a_i \bar{w}_{\xi'} + \underline{a}_i w(\xi') - \underline{a}_i \bar{w}_{\xi'}.\}, \\
M_i^1(\xi') = \{&(m_i^1(\xi'), a_i, u(\xi')) : \\
&m_i^1(\xi') \geq \underline{a}_i u(\xi') + a_i \underline{u}_{\xi'} - \underline{a}_i \underline{u}_{\xi'}, m_i^1(\xi') \geq \bar{a}_i u(\xi') + a_i \bar{u}_{\xi'} - \bar{a}_i \bar{u}_{\xi'}, \\
&m_i^1(\xi') \leq \bar{a}_i u(\xi') + a_i \underline{u}_{\xi'} - \bar{a}_i \underline{u}_{\xi'}, m_i^1(\xi') \leq a_i \bar{u}_{\xi'} + \underline{a}_i u(\xi') - \underline{a}_i \bar{u}_{\xi'}.\}.
\end{aligned}$$

Exact Unary Expansion: We note that the McCormick method used here provides additional relaxation of the original problem in (2.8). To have a more exact formulation, we utilize unary expansion [100] in the following corollary:

Corollary 2. *If for any $a \in \mathcal{A}$, the ambiguity set defined in* (2.6) *is nonempty, then the*

Bellman equation (2.5) *can be approximated as the following MIP formulation:*

$$V^t(\xi) = \max_{a\in\mathcal{A},\boldsymbol{w},\boldsymbol{u},q} \tilde{r}_{a\xi} + q - \sum_{\xi'\in\tilde{\mathcal{S}}}\left(\rho_0^{\xi}(\xi')w(\xi') + \sum_{i=1}^{N_a}\rho_i^{s}(\xi')\sum_{j=1}^{A_i}\tau_j^i\hat{m}_j^{0i}(\xi')\right) + \sum_{\xi'\in\tilde{\mathcal{S}}}\left(\sigma_0^{\xi}(\xi')u(\xi') + \sum_{i=1}^{N_a}\sigma_i^{\xi}(\xi')\sum_{j=1}^{A_i}\tau_j^i\hat{m}_j^{1i}(\xi')\right) \tag{2.13a}$$

$$s.t. \quad q\mathbf{1} \le \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}, \tag{2.13b}$$

$$\boldsymbol{w} + \boldsymbol{u} \le k\mathbf{1}, \tag{2.13c}$$

$$\boldsymbol{w}, \boldsymbol{u} \ge \mathbf{0}, \tag{2.13d}$$

$$\sum_{j=1}^{A_i}\psi_j^{0i} = 1, \ \sum_{j=1}^{A_i}\psi_j^{1i} = 1, \quad \forall i \in [N_a], \tag{2.13e}$$

$$\psi_j^{0i}, \psi_j^{1i} \in \{0,1\}, \quad \forall i \in [N_a], \ j = 1, ..., A_i \tag{2.13f}$$

$$(\hat{m}_j^{0i}(\xi'), \psi_j^{0i}, w(\xi')) \in \hat{M}_j^{0i}(\xi'), \quad \forall i \in [N_a], \xi' \in \tilde{\mathcal{S}}, \tag{2.13g}$$

$$(\hat{m}_j^{1i}(\xi'), \psi_j^{1i}, u(\xi')) \in \hat{M}_j^{1i}(\xi'), \quad \forall i \in [N_a], \xi' \in \tilde{\mathcal{S}}, \tag{2.13h}$$

where

$$\hat{M}_j^{0i}(\xi') = \{(\hat{m}_j^{0i}(\xi'), \psi_j^{0i}, w(\xi')) : \hat{m}_j^{0i}(\xi') \ge 0, \hat{m}_j^{0i}(\xi') \ge w(\xi') + \bar{w}(\xi')(\psi_j^{0i} - 1)\},$$
$$\hat{M}_j^{1i}(\xi') = \{(\hat{m}_j^{1i}(\xi'), \psi_j^{1i}, u(\xi')) : \hat{m}_j^{1i}(\xi') \le u(\xi'), \hat{m}_j^{1i}(\xi') \le \bar{u}(\xi')\psi_j^{1i}\},$$
$$\forall i \in [N_a], \ j = 1, ..., A_i,$$

where A_i represents the number of actions for action type i, τ_j^i is the j^{th} action for action type i (in the epidemic control setting, τ_j^i is the j^{th} item in $(0, 1, 2, ..., A_i - 1)$ for $i \in \{1, 2\}$), and ψ_j^{0i}, ψ_j^{1i} are auxillary binary variables.

The main difference between (2.12) and (2.13) is that (2.13) introduces binary variables, ψ_j^i, to represent whether or not the discrete action is chosen. Therefore, we can represent the action chosen as the following:

$$a^i = \sum_{j=1}^{A_i}\tau_j^i\psi_j^i, \quad \text{where } \psi_j^i = \begin{cases} 1 \text{ if } \tau_j^i \text{ is chosen} \\ 0 \text{ otherwise} \end{cases}$$

Thus, we can represent the bilinear terms in (2.13) as:

$$a^i w(\xi') = \sum_{j=1}^{A_i} \tau_j^i \hat{m}_j^{0i}, \text{ and } a^j u(\xi') = \sum_{j=1}^{A_i} \tau_j^i \hat{m}_j^{1i}$$

for all $i \in [N_a]$ where

$$\hat{m}_j^{0i} = \begin{cases} w(\xi') \text{ if action } j \text{ is chosen} \\ 0 \text{ otherwise} \end{cases}, \hat{m}_j^{1i} = \begin{cases} u(\xi') \text{ if action } j \text{ is chosen} \\ 0 \text{ otherwise} \end{cases} \tag{2.14}$$

The logic of (2.14) is enforced by the unary envelopes $\hat{M}_j^{0i}$ and $\hat{M}_j^{1i}$, using binary indicator variables ψ_j^{0i}, ψ_j^{1i}. This allows (2.13) to be an exact reformulation compared to (2.12), which is a relaxation of (2.5). However, introducing binary variables adds computational complexity so there is an inherent trade-off between (2.12) and (2.13). The realization of this trade-off is discussed further in Section 2.5.

2.5 Real-Time Dynamic Programming

To compute optimal policies based on the DRMDP model of the environment, we utilize the Real-Time Dynamic Programming (RTDP) algorithm [22]. RTDP is an online algorithm that combines heuristics search and dynamic programming to solve MDPs. Unlike traditional dynamic programming, RTDP does not require evaluating the entire state space to find an optimal solution. Instead, it computes a partial optimal policy for states reachable from the initial state, making it computationally efficient, especially for large state spaces.

It is important to note that the original RTDP algorithm iteratively solves the Bellman equation $\max_{a \in \mathcal{A}} \tilde{r}_{a\xi} + \lambda p_{a\xi}^{\mathrm{T}} V^{t+1}$, whereas in our case, we consider the distributionally robust Bellman equation (2.5).

In the context of a finite-horizon MDP, visiting the same state at different stages can lead to different values of expected reward-to-go. To address this, we adopt the assumption proposed in [280] and treat multiple visits to a state as visiting different states. Therefore, in the following algorithm, we use the notation (ξ, t) to denote the visit to state ξ at stage t, and we represent the corresponding expected reward-to-go as $V(\xi, t) = V^t(\xi)$.

Algorithm 1: RTDP with admissible heuristics $h((\xi, t))$

Input: number of iterations $niter$, initial state $\xi_{init} \in \tilde{\mathcal{S}} \cap \mathcal{S}$

Initialize $V((\xi, t)) = h((\xi, t))$ for all $\xi \in \tilde{\mathcal{S}}$, $t \in \{1, \ldots, T\}$

for $i = 1, \ldots, niter$ **do**

  Set $\xi = \xi_{init}$

  for $t = 1, \ldots, T-1$ **do**

    Solve the MIP relaxation in (2.12) or (2.13) to obtain optimal $V^*((\xi, t))$ and a^*

    Update $V((\xi, t)) \leftarrow V^*((\xi, t))$

    Sample the next state $\xi' \in \tilde{\mathcal{S}} \cap \mathcal{S}$, and set $\xi = \xi'$

  end

end

In the provided algorithm, note that ξ_{init} is defined outside the "for" loops, indicating that for each iteration, we start from the same initial state. The function $h((\xi, t))$ represents a heuristic search pattern [166, 194], which aims to find a solution specifically for the states reachable from the initial state by following an optimal policy. This approach is particularly effective for problems with large state spaces since it avoids the need to solve for the entire state space, as required by the traditional value iteration algorithm.

It is worth noting that if $h((\xi, t))$ is an *admissible* heuristic function, meaning that its values are less than or equal to the true reward values, the algorithm will converge to the optimal solution. This result is formally stated below.

Theorem 2. *[22] If the goal is reachable from each initial state and the heuristic function is admissible, the RTDP iterations will eventually yield optimal objective value and optimal controller's policy on the set of states reachable from the initial states.*

2.6 Numerical Studies

In this section, we evaluate the effectiveness of our DRMDP model and RTDP algorithm in addressing the epidemic control problem.

We begin with Subsection 2.6.1, where we conduct a comprehensive performance comparison between our DRMDP model and traditional MDP-based models for addressing the

epidemic control problem. Specifically, in Subsection 2.6.1, we analyze the performance of two formulations of the DRMDP model: one utilizing the relaxed McCormick envelope (Corollary 1) and the other using the exact unary envelope (Corollary 2). This analysis helps us determine the optimal formulation for the DRMDP model before proceeding with the comparison. Moving on to Subsection 2.6.1, we present a comprehensive performance comparison between our DRMDP model and classic MDP as well as robust MDP models. We evaluate the models based on their effectiveness in controlling the number of infectives, the resulting epidemic control policies, and the total discounted health and economic loss. To assess the robustness of the models, we consider various scenarios, including accurate or inaccurate knowledge of transition probabilities.

Additionally, to assess the effectiveness and efficiency of RTDP when applied to the DRMDP model, we perform a comparative analysis in Subsection 2.6.2 between RTDP algorithm and the conventional dynamic programming (DP) approach using backward induction.

Lastly, in Subsection 2.6.3, we conduct a sensitivity analysis of the DRMDP model for the epidemic control problem. By systematically varying the values of various input parameters, we examine their influence on the model's performance. This analysis allows us to assess the model's sensitivity to changes in epidemic parameters and understand its performance across different scenarios.

All MDP-based models are solved using Gurobi. The RTDP and DP algorithms are implemented in Python. All experiments are conducted on machines equipped with 2.7 GHz Quad-Core Intel Core i7 processors.

2.6.1 Model Comparison

In this subsection, we present a comprehensive performance comparison between our proposed DRMDP model and traditional MDP-based models for addressing the epidemic control problem.

We refer to the decision-dependent DRMDP model formulated with the McCormick MIP approach in (2.12) as `DRMDP-McCormick`, and we denote the decision-dependent DRMDP model formulated with the unary MIP approach in (2.13) as `DRMDP-Unary`. In all experiments,

we set $k = 1000$ for (2.12) and (2.13). For comparison purposes, we consider a classic MDP model with known transition probabilities $\tilde{\boldsymbol{p}}^0_{a\xi}$ as in equation (2.4). Thus, we solve the Bellman equation $V^t(\xi) = \max_{a\in\mathcal{A}_\xi}\{\tilde{r}_{a\xi} + \lambda(\tilde{\boldsymbol{p}}^0_{a\xi})^\top \boldsymbol{V}^{t+1}\}$ at stage t for this baseline, denoted as `MDP`. Additionally, we consider a robust MDP model where we use the worst-case transition probabilities $\boldsymbol{p}'_{a\xi}$. These probabilities represent the highest probability of transitioning from exposed to infectious while staying within a distance of 0.5 from the nominal probability $\tilde{\boldsymbol{p}}^0_{a\xi}$ in (2.4). Consequently, we solve the Bellman equation $V^t(\xi) = \max_{a\in\mathcal{A}_\xi}\{\tilde{r}_{a\xi} + \lambda(\boldsymbol{p}'_{a\xi})^\top \boldsymbol{V}^{t+1}\}$ at stage t for this robust MDP baseline, which we denote as `Robust MDP`.

We compute optimal strategies for all models using the RTDP algorithm. For the heuristic function of RTDP, we utilize $h((\xi, T)) = 0, \forall\xi \in \tilde{\mathcal{S}}$ and $h((\xi, t)) = \max_{a\in\mathcal{A}_\xi} \tilde{r}_{a\xi}, \forall\xi \in \tilde{\mathcal{S}}, t = 1, \ldots, T-1$. In other words, $h((\xi, t))$ represents only the initial reward, which is less than or equal to the true reward (initial reward plus some non-negative future reward). Therefore, $h((\xi, t))$ is considered *admissible*, and Theorem 2 holds.

McCormick and Unary for DRMDP

To determine the optimal formulation for the DRMDP, we compare the computational efficiency and performance of `DRMDP-McCormick` and `DRMDP-Unary`.

We conduct experiments with the epidemic control problem, setting $M = 5$, $L = 5$, a time horizon of $T = 12$ (representing a 12-month horizon), a discretization level of $Y = 30$, an initial proportion of exposed individuals $p_E(1) = 0.1$, an initial proportion of susceptible individuals $p_S(1) = 0.6$, and an initial proportion of infectious individuals $p_I(1) = 0.3$. When simulating the spread of the epidemic, we consider two scenarios for the transition probabilities:

1. $\tilde{\boldsymbol{p}}^0_{a\xi}$ in (2.4). In this scenario, the model possesses precise knowledge of the transition probabilities.

2. $\boldsymbol{q}_{a\xi}$, which satisfies $||\boldsymbol{q}_{a\xi} - \tilde{\boldsymbol{p}}^0_{a\xi}||_1 \leq 0.5$. In this scenario, the transition probabilities are incorrectly specified, and the model lacks knowledge of the true transition probabilities.

The average computational time per iteration for `DRMDP-Unary` is $149.9s$, while for

`DRMDP-McCormick`, it is $75.1s$. However, the performance of `DRMDP-Unary` and `DRMDP-McCormick` is similar. Specifically, for transition probabilities $\tilde{\boldsymbol{p}}^0_{a\xi}$, the performance values are -3.77×10^7 and -3.79×10^7, respectively. For transition probabilities $\boldsymbol{q}_{a\xi}$, the performance values are -3.69×10^7 and -3.72×10^7, respectively. Since `DRMDP-McCormick` achieves similar performance with only half the runtime, we adopt `DRMDP-McCormick` for the subsequent experiments, denoting it as `DRMDP` for brevity.

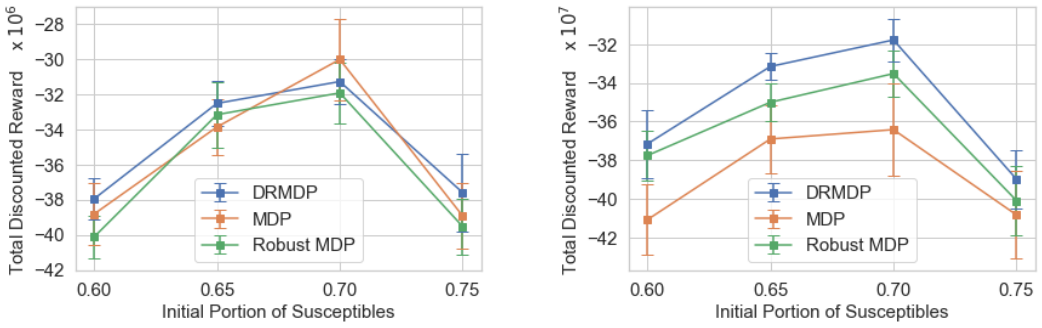
DRMDP, MDP and Robust MDP

We proceed by comparing the performance of our proposed `DRMDP` model to classic `MDP` and `Robust MDP` models.

We conduct experiments with the epidemic control problem, setting $M = 5$, $L = 5$, a time horizon of $T = 12$ (representing a 12-month horizon), and a discretization level of $Y = 100$. We fix the initial proportion of exposed individuals as $p_E(1) = 0.1$ and consider different initial proportions of susceptible and infectious individuals: $p_S(1) = 0.60, 0.65, 0.70, 0.75$ and $p_I(1) = 0.30, 0.25, 0.20, 0.15$, respectively. Similarly, we consider two different sets of transition probabilities, $\tilde{\boldsymbol{p}}^0_{a\xi}$ and $\boldsymbol{q}_{a\xi}$, for simulating the epidemic.

The results comparing `DRMDP`, `MDP`, and `Robust MDP` are presented in Figure 2.3. As shown in Figure 2.3a, when the transition probability used in simulation is $\tilde{\boldsymbol{p}}^0_{a\xi}$, the total discounted rewards of `DRMDP`, `MDP`, and `Robust MDP` are similar. However, when the transition probability used in simulation is $\boldsymbol{q}_{a\xi}$ (Figure 2.3b), the total discounted rewards of `DRMDP` are higher compared to `MDP` and `Robust MDP`. These findings demonstrate that `DRMDP` outperforms `MDP` and `Robust MDP` in adapting to environments when the true transition probabilities are unknown.

The optimal actions taken by `DRMDP`, `MDP`, and `Robust MDP` at each stage are summarized in Table 2.1, where the notation $[x, y]$ represents the action of choosing a vaccination level of x and a transmission-reduction level of y. Analyzing Table 2.1, we observe that `DRMDP` administers vaccinations to a greater number of susceptible individuals compared to `MDP`. Additionally, as the time period progresses, the need for implementing interventions decreases across all models.



(a) (b)

Figure 2.3: Total rewards versus initial proportion of susceptible individuals, averaged across 10 runs. The error bars show mean ± standard deviation. (a) the transition probability used in simulation is $\tilde{\boldsymbol{p}}^0_{a\xi}$. (b) the transition probability used in simulation is $\boldsymbol{q}_{a\xi}$.

Table 2.1: Action taken at each stage (averaged across 10 runs) when the transition probability used in simulation is $\boldsymbol{q}_{a\xi}$.

Stage	$t=1$	$t=2$	$t=3$	$t=4$	$t=5$	$t=6$	$t=7$	$t=8$	$t=9$	$t=10$	$t=11$
DRMDP, $p_S(1)=0.60$	[1,5]	[5,5]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
MDP, $p_S(1)=0.60$	[1,5]	[4.6,5]	[0.5,0.5]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
Robust MDP, $p_S(1)=0.60$	[1,5]	[5,5]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
DRMDP, $p_S(1)=0.65$	[1,5]	[3.8,5]	[3,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
MDP, $p_S(1)=0.65$	[1,5]	[3.6,5]	[2.5,2.5]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
Robust MDP, $p_S(1)=0.65$	[1,5]	[4.1,5]	[3,1]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
DRMDP, $p_S(1)=0.70$	[1,5]	[3,5]	[5,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
MDP, $p_S(1)=0.70$	[1,5]	[2.7,5]	[5,5]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
Robust MDP, $p_S(1)=0.70$	[1,5]	[3.2,5]	[5,1]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
DRMDP, $p_S(1)=0.75$	[1,5]	[2,5]	[5,0.5]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
MDP, $p_S(1)=0.75$	[1,5]	[2,5]	[4,4]	[1,1]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]
Robust MDP, $p_S(1)=0.75$	[1,5]	[2.4,5]	[5,1.5]	[0.7,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]	[0,0]

Furthermore, with increasing initial proportion of susceptible individuals $p_S(1)$, we note that vaccinations are required in more stages. For instance, in the case of `DRMDP`, when $p_S(1) = 0.60$, vaccination actions are taken during stages $t = 1$ to 2, while for $p_S(1) = 0.75$, vaccinations are required during stages $t = 1$ to 3. This observation indicates a higher demand for vaccinations when a larger proportion of the population is susceptible.

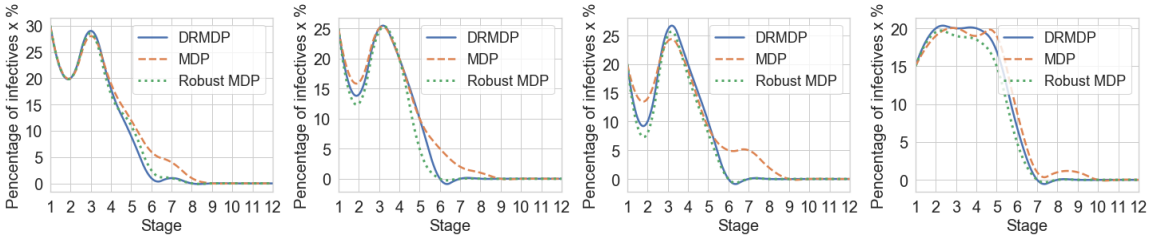
Moreover, the results demonstrate that `DRMDP` adopts a more aggressive vaccination strategy compared to `MDP`, implying that the distributionally robust approach favors allocating more resources towards vaccines to ensure a policy that is robust against uncertainty. On the other hand, `Robust MDP` implements an even stricter policy than `DRMDP`, resulting in lower total rewards due to the increased cost of intervention. This finding suggests that `DRMDP`, which considers the worst-case distribution $\mu_{a\xi}$, exhibits a less conservative behavior than `Robust MDP`.

The percentage of infectives, recovered individuals, and rewards for `DRMDP`, `MDP`, and `Robust MDP` at each stage, considering the true transition probability $\boldsymbol{q}_{a\xi}$, are depicted in Figure 2.4, 2.5, and 2.6, respectively. From these figures, several observations can be made.

Firstly, `DRMDP` exhibits higher stage-wise rewards and demonstrates greater effectiveness in controlling the number of infectives compared to `MDP`. Specifically, in Figure 2.4 and 2.5, both models perform similarly in the early stages ($t = 1$ to 4), but `DRMDP` achieves zero infectives and full recovery faster than `MDP` in the intermediate to late stages ($t = 5$ to 12). Similarly, Figure 2.6 showcases that `DRMDP` achieves a higher reward earlier than `MDP`.

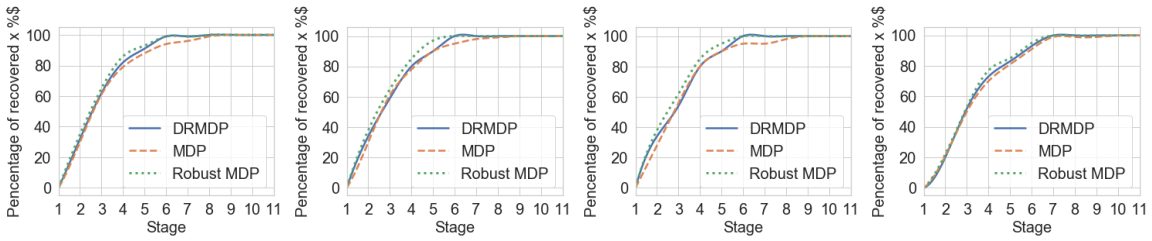
These results can be attributed to the fact that the optimal policy derived from the `MDP` model is based on the assumption that the true transition probabilities are $\tilde{\boldsymbol{p}}^0_{a\xi}$, while the policy obtained from the `DRMDP` model is robust to uncertainty in transition probabilities, which allows it to adapt and perform well even without knowing the true probabilities.

Comparing `DRMDP` with `Robust MDP`, we observe that `DRMDP` is slightly less effective in controlling the number of infectives but achieves a higher stage-wise reward overall. These findings align with our earlier conclusions that `DRMDP` exhibits a less conservative behavior compared to `Robust MDP`.



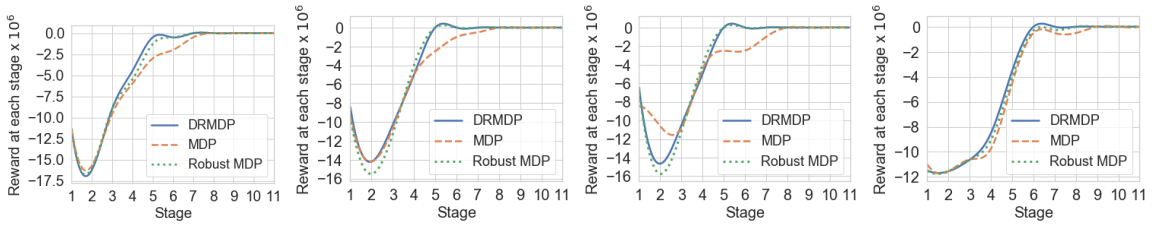
(a) $p_S(1) = 0.6$ (b) $p_S(1) = 0.65$ (c) $p_S(1) = 0.7$ (d) $p_S(1) = 0.75$

Figure 2.4: Percentage of infectives versus stage (averaged across 10 runs) when the transition probability used in simulation is $\boldsymbol{q}_{a\xi}$.



(a) $p_S(1) = 0.6$ (b) $p_S(1) = 0.65$ (c) $p_S(1) = 0.7$ (d) $p_S(1) = 0.75$

Figure 2.5: Percentage of recovered versus stage (averaged across 10 runs) when the transition probability used in simulation is $\boldsymbol{q}_{a\xi}$.



(a) $p_S(1) = 0.6$ (b) $p_S(1) = 0.65$ (c) $p_S(1) = 0.7$ (d) $p_S(1) = 0.75$

Figure 2.6: Stage-wise reward (averaged across 10 runs) when the transition probability used in simulation is $\boldsymbol{q}_{a\xi}$.

2.6.2 Algorithm Comparison

To comprehensively evaluate the effectiveness and efficiency of the RTDP algorithm when applied to the `DRMDP` model, we compare its performance with the conventional Dynamic Programming (DP) approach using backward induction. The DP method for solving the `DRMDP` model is described in Algorithm 2.

Algorithm 2: Conventional dynamic programming

Initialize $V((\xi, T)) = \tilde{q}_R(\xi)$ for $\xi \in \tilde{\mathcal{S}}$

for $t = T-1, \ldots, 1$ **do**

  for $\xi \in \tilde{\mathcal{S}} \cap \mathcal{S}$ **do**

    Solve MIP relaxation in (2.12) or (2.13) to obtain optimal $V^*((\xi, t))$ and a^*

    Update $V((\xi, t)) \leftarrow V^*((\xi, t))$

  end

  Update $V((\xi, t)) \leftarrow 0$ for $\xi \in \tilde{\mathcal{S}} \backslash \mathcal{S}$

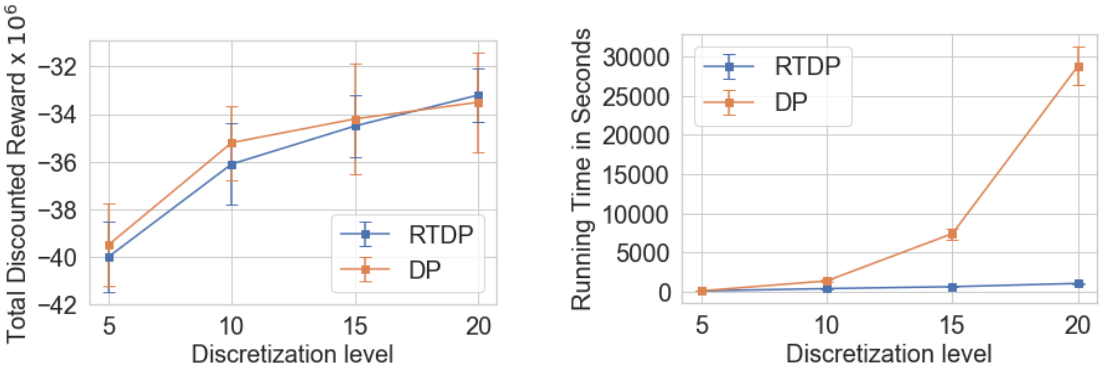
end

In this comparison, we consider different settings of the state space discretization level, denoted as Y, as defined in Section 2.3. This allows us to assess the impact of state space granularity on the performance of both algorithms. For the RTDP algorithm, we execute a total of 20 iterations when $Y = 5$, while for $Y = 10$, 15, and 20, we perform 50 iterations.

As shown in Figure 2.7a, DP yields slightly higher total rewards than RTDP for small discretization levels ($Y = 5, 10$). However, as the discretization level increases ($Y = 15, 20$), the total rewards of DP and RTDP become similar. Conversely, as depicted in Figure 2.7b, DP requires significantly more computational time than RTDP, especially for high discretization levels. This is due to the fact that DP needs to solve Bellman equations for all states, and the number of states grows polynomially with the discretization level.

2.6.3 Sensitivity Analysis

In this subsection, we analyze the impact of several input parameters on the performance of the `DRMDP` model for the epidemic control problem. Specifically, we investigate the influence



(a) Total rewards

(b) Computational time

Figure 2.7: Performance and runtime comparisons of RTDP and DP. Averaged across 3 runs.

of the following parameters:

1. Q: the vaccine price.

2. k_R: the cost of increasing the transmission reduction level by one level.

3. $\mu\beta$: Here, μ represents the contact rate without any transmission reduction method, and β denotes the probability of a susceptible individual becoming infected upon contact with an infectious individual. Since both μ and β collectively affect the number of susceptible individuals who become exposed, we evaluate their product as a combined variable in the sensitivity analysis.

4. W: the health loss plus treatment cost associated with a single infection.

5. α_0: the maximum possible fractional reduction in the contact rate achievable through transmission reduction interventions.

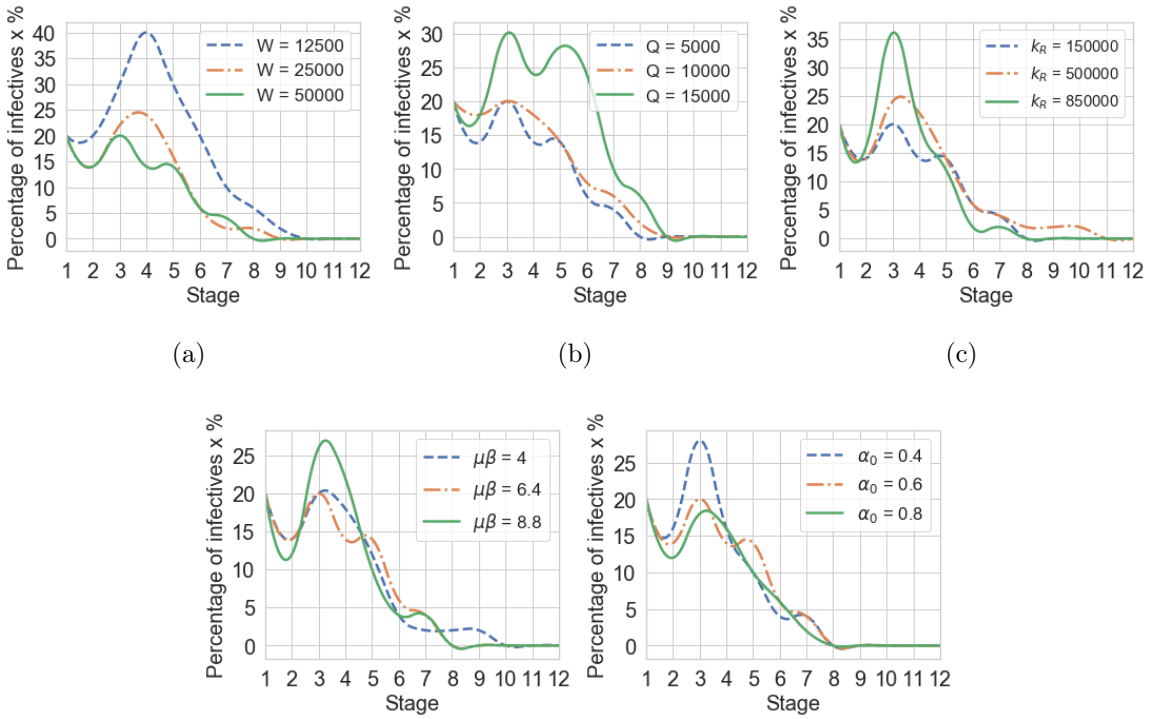


Figure 2.8: Sensitivity analysis of `DRMDP` (solved with RTDP). Percentage of infectives versus stage (averaged across 5 runs) when the transition probability used in simulation is $\boldsymbol{q}_{a\xi}$.

We kept the value of k fixed at 1000 in equation (2.12) and utilized the RTDP algorithm to calculate optimal strategies for the `DRMDP` model in all our sensitivity analysis experiments. The analysis was performed on the epidemic control problem, considering a time horizon of $T = 12$, an initial proportion of exposed individuals $p_E(1) = 0.1$, an initial proportion of susceptible individuals $p_S(1) = 0.7$, and an initial proportion of infectious individuals $p_I(1) = 0.2$.

Figure 2.8 illustrates the percentage of infectives at each stage for different input values. From the figure, we observe that the overall percentage of infectives increases as Q, k_R, or $\mu\beta$ increases, while it decreases as W or α_0 increases.

When the price of vaccines (Q) or transmission reduction interventions (k_R) increases,

the willingness to allocate resources for vaccinations or interventions decreases. Consequently, we expect to see a higher proportion of infected individuals. Additionally, higher values of μ or β lead to easier disease spread, resulting in more infections.

Conversely, an increased value of W indicates a higher penalty for infections, prompting policymakers to prioritize vaccination efforts to reduce the number of infections. Finally, a larger value of α_0 signifies the greater effectiveness of transmission reduction methods, leading to a reduced number of infected individuals.

2.7 Conclusion

In this study, we present a Distributionally Robust Markov Decision Process (DRMDP) framework for addressing the dynamic epidemic control problem. Our aim is to provide effective strategies to public health decision-makers that can reduce the proportion of infected individuals while considering cost-efficiency. The proposed DRMDP model is designed to handle the uncertainties inherent in scenarios where the true distribution of disease transmission parameters remains unknown.

To capture the impact of policy actions on the system dynamics, we incorporate an endogenous ambiguity set within our model. This inclusion ensures that the influence of policy-makers' decisions on the system dynamics is comprehensively considered. To solve the problem, we discretize the model and reformulate it as a mixed integer programming (MIP) using either McCormick or unary envelopes. Our computational experiments demonstrate that both approaches yield similar results.

Our findings indicate that the DRMDP outperforms both the classic MDP and robust MDP when evaluated against various metrics, including cost minimization and reduction of infected individuals. This superiority is particularly evident when dealing with scenarios where the true distribution of transition probabilities is unknown. In contrast, the performance of the MDP model deteriorates when the assumption of known transition probabilities is violated. The strength of the DRMDP model lies in its ability to handle uncertain transition probabilities without relying on this assumption, thereby delivering reliable results even in the absence of true probabilities.

Moreover, we highlight the efficiency and effectiveness of the RTDP algorithm in computing

optimal policies based on the DRMDP model. The RTDP algorithm achieves similar performance to traditional DP methods while requiring significantly less computational time.

To conclude our analysis, we conduct a sensitivity analysis that reveals the impact of increasing vaccine and transmission-reduction intervention prices, as well as contact and infection rates, on the proportion of infected individuals. Conversely, we observe that increasing health loss, treatment cost, and intervention effectiveness have the opposite effect.

As a direction for future research, we suggest exploring a continuous action space instead of the current discrete action space. Additionally, while this paper focuses on uncertainty in transition probabilities, future studies could expand their scope to encompass uncertainty in other model parameters.

Chapter 3

NONPARAMETRIC KULLBACK-LEIBLER CONSTRAINED POLICY OPTIMIZATION TOWARDS OPTIMAL PRICING OF DEMAND RESPONSE

The majority of model-free RL methods cannot guarantee the stability and optimality of the learned policy, which is undesirable in safety-critical systems. In this chapter, we propose an innovative nonparametric policy optimization approach with Kullback-Leibler divergence constraint. Our approach ensures the stability of the policy update through trust region constraints, and improves optimality by removing the restrictive parametric assumption on policy representation that the majority of the RL literature adopts. We derive a closed-form expression of optimal policy update for each iteration and develop an efficient on-policy actor-critic algorithm to address the proposed constrained policy optimization problem. The effectiveness of our approach is evaluated on a price-based demand response (DR) problem of the electricity market, with the goal of finding optimal pricing strategies to adjust electricity consumption in a timely and reliable manner. The experiments on two DR cases show the superior performance of our proposed nonparametric constrained policy optimization method compared with state-of-the-art RL algorithms.

3.1 Introduction

Demand response (DR), as an efficient way to reduce the peak load and improve the grid reliability, has been widely applied in recently years and recognized as one of the main forces to advance the smart grid technology. In order to incorporate DR resources into the wholesale energy market, a number of regional grid operators (such as NYISO, PJM, ISO-NE, and ERCOT) have established demand response programs for consumers to participate [91]. Federal Energy Regulatory Commission (FERC) also issued its order in 2020 to enable distributed energy resources, which include demand response, to participate in regional wholesale electricity markets [2].

In general, incentive-based programs and price-based programs are the two categories of DR systems. In incentive-based schemes, customers receive fixed or time-varying compensation in exchange for limiting their power usage during peak hours or system contingencies. In price-based schemes, customers are encouraged to shift their electricity consumption from high to low price periods in order to save money on their electricity bill. In the latter case, deriving an effective and efficient pricing strategy is critical in order to adjust electricity consumption in a reliable and timely manner.

Traditionally, a Stackelberg game model [165, 292] is often used to characterize the interactions between DR service provider (SP) and participating consumers (PCs), in which SP acts as the leader who sets the price strategy and PCs are the followers who adjust their electricity consumption. However, in practice, it is very challenging for both SP and PCs to set up each other's model and accurately estimate the model parameters. Also, due to a possible large number of PCs, there will be a significant number of models with different customer behavior patterns, which makes the model learning even harder. This challenge can be addressed by reinforcement learning (RL), since it is a model-free technique that does not require the identification of models for individual players.

In this vein, a variety of model-free RL methods including value-based methods such as Q-learning [131, 160, 204] and policy-based methods such as Advantage Actor Critic (A2C) [19], Trust Region Policy Optimization (TRPO) [149] have been exploited to address the dynamic pricing strategy of DR problems. However, the performances of traditional model-free RL methods are not satisfactory in practice for the following reasons: 1) different behavior patterns of PCs bring a high degree of environmental dynamics, which inevitably induces the instability of policy update and the difficulty for the convergence of final policy; 2) many RL methods may not reach policy optimality even after it stabilizes. In addition to bad initialization and inadequate exploration, the model restriction that is employed could exclude the algorithms from attaining a more optimal policy [249].

In this paper, we propose a novel trust region constrained policy optimization method that can effectively address the DR pricing strategy problem. Similar as TRPO, we impose a Kullback-Leibler (KL) divergence based constraint (trust region constraint) to restrict the size of the policy update, so that a good stability can be maintained. However, unlike

the traditional gradient based policy optimization methods such as A2C, TRPO, or PPO [223] that limit the policy representation to a particular parametric distribution class (e.g., Gaussian [218]), we release this restrictive distributional assumption by allowing all admissible policies in the trust region. Since in this case, the policy learnt is not confined to the scope of parametric functions, this certainly opens up the possibility of converging to a better final policy. In fact, we observed significant improvements in nearly all our test cases. In addition, the proposed method is more flexible than the majority of the state-of-the-art gradient based policy optimization methods - it can be applied to both discrete and continuous action cases. We successfully derive the reformulations of the proposed policy optimization problem and obtain the closed-form optimal policy update. We have investigated the effectiveness of the proposed approach with two representative test cases on DR. Both cases demonstrate that our approach outperforms state-of-the-art gradient based policy optimization methods.

3.2 MDP Formulation

We use a similar Markov Decision Process (MDP) model as [160] to describe the dynamic decision-making problem of price-based DR program, which is defined as a tuple $< T, \lambda, \mathcal{S}, \mathcal{A}, \boldsymbol{p}, \boldsymbol{r} >$, where T is the time horizon, λ is the discount factor, $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\boldsymbol{p}$ is the transition probability between states depending on the taken action, and $\boldsymbol{r}$ is the reward associated with the taken action. However, unlike [160], we allow continuous state and action for our MDP model.

- **Time Horizon**: We consider a finite time horizon with length $T = 24$, which represents 24 hours in a day. We use $t = 1, 2, \ldots, T$ to represent the discrete time stage.

- **Action and State**: The action of the dynamic pricing DR model is defined as $a_t = (\phi_{t,1}, \ldots, \phi_{t,N})$, where N is the total number of PCs and $\phi_{t,n}$ denotes the retail electricity price for the n-th PC at time t.

  The state of the dynamic pricing DR model is defined as

$$s_t = ((d_{t,1}, c_{t,1}), \ldots, (d_{t,N}, c_{t,N})),$$

where $d_{t,n}$ represents the base energy demand of the n-th PC at time t before knowing the retail price, and $c_{t,n}$ represents the actual energy consumption of the n-th PC at time t after knowing the retail price.

The base energy demand and the actual energy consumption of the n-th PC at time t are defined as:

$$d_{t,n} = d_{t,n}^{crit} + d_{t,n}^{curt}, \tag{3.1a}$$

$$c_{t,n} = c_{t,n}^{crit} + c_{t,n}^{curt}, \tag{3.1b}$$

where $d_{t,n}^{crit}$ and $c_{t,n}^{crit}$ denote the energy demand and energy consumption of the n-th PC at time t for critical load, and $d_{t,n}^{curt}$ and $c_{t,n}^{curt}$ denote the energy demand and energy consumption of the n-th PC at time t for curtailable load.

The critical load demands are always completely met, whereas the curtailable load consumption follows the price elasticity of demands, i.e.,

$$c_{t,n}^{crit} = d_{t,n}^{crit}, \tag{3.2a}$$

$$c_{t,n}^{curt} = d_{t,n}^{curt} \times (1 + \xi_t \times \frac{\phi_{t,n} - \pi_t}{\pi_t}), \tag{3.2b}$$

where $\xi_t < 0$ denotes the elasticity coefficient at time t and π_t represents the wholesale electricity price at time t.

- **Reward**: To characterize the reward, we first define the SP's profits: $P(t) = \sum_{n=1}^{N}(\phi_{t,n} - \pi_t) \times c_{t,n}$. Then, we define PCs' costs $C(t)$, which is defined as the sum of the electricity costs and the dissatisfaction costs for all PCs, i.e., $C(t) = \sum_{n=1}^{N}(\phi_{t,n} \times c_{t,n} + \delta_{t,n})$. Here $\delta_{t,n} = \frac{\alpha_n}{2}(d_{t,n}^{curt} - c_{t,n}^{curt})^2 + \beta_n(d_{t,n}^{curt} - c_{t,n}^{curt})$ denotes the dissatisfaction cost of the n-th PC at time t, where α_n is the customer preference parameter [293] and β_n is a predefined constant [291].

  The reward of the dynamic pricing DR model is then defined as a weighted combination of $P(t)$ and $-C(t)$:

$$r(s_t, a_t) = \rho \times P(t) - (1 - \rho) \times C(t). \tag{3.3}$$

Based on the reward, the total discounted return of the dynamic pricing DR model in a complete trajectory from time t onward can be represented as:

$$R_t = \sum_{k=0}^{T-t} \lambda^k r(s_{t+k}, a_{t+k}), \tag{3.4}$$

and the performance of a stochastic policy π can be represented as

$$\eta(\pi) = \mathbb{E}_{s_0,a_0,s_1\ldots}[\sum_{t=0}^{\infty} \lambda^t r(s_t, a_t)]. \tag{3.5}$$

As shown in [129], the expected return of a new policy π' can be expressed in terms of the advantage over the old policy π: $\eta(\pi') = \eta(\pi) + \mathbb{E}_{s\sim\rho^{\pi'},a\sim\pi'}[A^{\pi}(s,a)]$, where $A^{\pi}(s,a) = \mathbb{E}[R_t|s_t = s, a_t = a; \pi] - \mathbb{E}[R_t|s_t = s; \pi]$ represents the advantage function and ρ^{π} represents the unnormalized discounted visitation frequencies, i.e., $\rho^{\pi}(s) = \sum_{t=0}^{\infty} \gamma^t P(s_t = s)$.

Trust Region Policy Optimization (TRPO) [218] employs a trust region constraint to ensure that the new policy does not deviate significantly from the old policy. Furthermore, in TRPO, the policy π is parameterized as π_θ with the parameter vector θ. For notation brevity, we use θ to represent the policy π_θ. Then, the new policy θ' is updated in each iteration to maximize the expected value of the advantage function:

$$\begin{aligned} &\max_{\theta'} \quad \mathbb{E}_{s\sim\rho^{\theta},a\sim\theta'}[A^{\theta}(s,a)] \\ &\text{s.t.} \quad \mathbb{E}_{s\sim\rho^{\theta}}[d_{KL}(\theta',\theta)] \leq \delta, \end{aligned} \tag{3.6}$$

where δ is the threshold of the distance between the old policy θ and the new policy θ'; $d_{KL}(\theta',\theta)$ represents the KL divergence between the old and new policies, i.e., $d_{KL}(\theta',\theta) = \int_{a\in\mathcal{A}} \pi_{\theta'}(a|s) \log(\frac{\pi_{\theta'}(a|s)}{\pi_\theta(a|s)}) da$. Though parametrizing the policy may help ease computations, as indicated in [249], optimizing over parametric distributions will cause local movements in the action space and converge to a sub-optimal solution.

3.3 A Nonparametric Framework to Learn Distributional Policies

In this section, we first develop nonparametric constrained policy optimization framework, with the KL divergence to construct trust region. Then we derive a closed-form of the policy update and propose an efficient on-policy actor-critic algorithm to address the proposed problem.

3.3.1 Problem formulation

In our model, we release the restrictive assumption that a policy has to follow a parametric distribution class. Instead, we consider all admissible policies within the trust region, i.e,

$$\mathbb{E}_{s\sim\rho^{\pi}}[d_{KL}(\pi'(\cdot|s),\pi(\cdot|s))] \leq \delta. \tag{3.7}$$

That is, the policy update is nonparametric. As long as its expected distance from the old policies is no more than a threshold level δ, the policy will be considered as the candidate of the next policy update.

Since the optimization goal in each policy update is to obtain the maximal expected value of the advantage function, the proposed model focuses on identifying the optimistic policy that falls in the set depicted in (3.7). Therefore, our framework is shown as follows:

$$\begin{aligned} &\max_{\pi'\in\mathcal{D}} \quad \mathbb{E}_{s\sim\rho^{\pi},a\sim\pi'(\cdot|s)}[A^{\pi}(s,a)] \\ &\text{where} \quad \mathcal{D}=\{\pi'|\mathbb{E}_{s\sim\rho^{\pi}}[d_{KL}(\pi'(\cdot|s),\pi(\cdot|s))]\leq\delta\}. \end{aligned} \tag{3.8}$$

This framework is related to distributionally robust optimization (DRO) [203]. However, we note that our constrained policy optimization is conceptually different from DRO. Constrained policy optimization seeks the most optimistic policy that falls within a trust region, whereas DRO seeks to minimize some worst-case loss given by an adversarial distribution of unknown parameters within an ambiguity set.

Before describing our main result, we adopt a mild and conventional assumption that generally holds true in most practical cases:

Assumption 1. *Assume $A^{\pi}(s,a)$ is bounded, i.e., $\sup_{a\in\mathcal{A}}|A^{\pi}(s,a)|<\infty$, $\forall s\in\mathcal{S}$.*

3.3.2 Policy update

We present the optimal policy update of our method. In this formation, both the state space and the action space can be discrete or continuous. Theorems 3 and 4 show the reformulation of (3.8) and closed-form of optimal policy update. The detailed proofs of the two theorems can be found in Appendix.

Theorem 3. *If Assumption 1 holds, then the KL trust-region constrained optimization problem in* (3.8) *is equivalent to the following problem:*

$$\min_{\beta\geq 0}\{l_0(\beta) := \beta\delta + \mathbb{E}_{s\sim\rho^\pi}\beta\log\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}]\}. \tag{3.9}$$

Theorem 4. *If Assumption 1 holds and* β^* *is the global optimal solution to* (3.9)*, then the optimal policy solution to the KL trust-region constrained optimization problem in* (3.8) *is:*

$$\pi'^*(a|s) = \mathbb{F}(\pi) = \frac{e^{A^\pi(s,a)/\beta^*}\pi(a|s)}{\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta^*}]}, \quad \forall s\in\mathcal{S}, a\in\mathcal{A}. \tag{3.10}$$

We use gradient-based global optimization algorithms [143, 260, 297] to find the global optimal β^* in (3.9). The gradient of the objective in (3.9) is derived as below:

$$\begin{aligned}\frac{\partial l_0(\beta)}{\partial\beta} = \delta + \mathbb{E}_{s\sim\rho^\pi}\{&\log\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}] \\ &-\frac{\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}\times A^\pi(s,a)]}{\beta\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}]}\}.\end{aligned}$$

In implementation, we obtain the global optimal β^* using the Basin Hopping algorithm [260], which is a two-phase global optimization method that utilizes the gradient information to speed up the search in the local phase.

3.3.3 On-policy actor-critic algorithm

We provide a practical on-policy actor-critic algorithm as described in Algorithm 3, to address the proposed nonparametric constrained policy optimization problem. The trajectories sampled in the algorithm can either be complete or partial. If it is complete, G_t can be obtained by using the accumulated discounted rewards, i.e., $R_t = \sum_{k=0}^{T-t}\lambda^k r_{t+k}$. If it is partial, G_t can be estimated by using the multi-step temporal difference (TD) methods [61]: $\hat{R}_{t:t+n} = \sum_{k=0}^{n-1}\lambda^k r_{t+k} + \lambda^n V(s_{t+n})$. To estimate the advantage $\hat{A}_t^{\pi_k}$, we can either use the Monte Carlo approach, i.e., $\hat{A}_t^{\pi_k} = G_t - V_{\psi_k}(s_t)$ or Generalized Advantage Estimation (GAE) [220].

Algorithm 3: On-policy actor-critic algorithm

Input: number of iterations K, learning rate α

Initialize policy π_0 and value network V_{ψ_0} with random parameter ψ_0

for $k = 0, 1, 2 \dots K$ **do**

Collect trajectory set $\mathcal{D}_k$ on policy π_k

For each timestep t in each trajectory, compute total returns G_t and estimate advantages $\hat{A}_t^{\pi_k}$

Update value: $\psi_{k+1} \leftarrow \psi_k - \alpha \nabla_{\psi_k} \sum (R_t - V_{\psi_k}(s_t))^2$

Update policy: $\pi_{k+1} \leftarrow \mathbb{F}(\pi_k)$ via (3.10) with $\hat{A}_t^{\pi_k}$

end

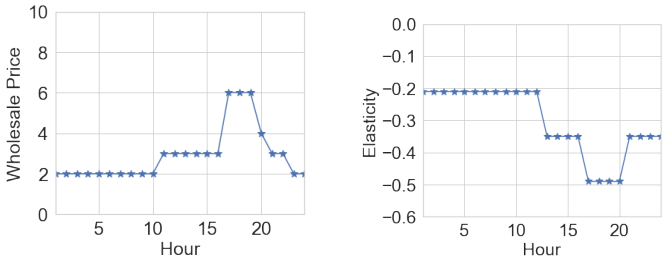
3.4 Experiments

In this section, we test the performance of our approach by comparing with multiple state-of-the-art RL approaches, for both continuous and discrete action spaces. Q-learning, A2C, TRPO, PPO, and DDPG [151] are among the benchmark methods we choose. The reason that we compare our method with A2C is because that it is also an on-policy actor-critic method that utilizes the advantage information. We also choose PPO as the benchmark since it is a variant of TRPO: Instead of restricting the KL divergence between the old and the new policies, it penalizes on the KL divergence. DDPG is chosen because previous work [233] demonstrates its effectiveness in solving DR problems. We test Q-learning only for the discrete action case and DDPG only for continuous action case, due to the nature of these algorithms: One only works for discrete actions while the other primarily supports continuous actions. The environment is implemented with OpenAI Gym [40] and experiments are conducted using Python code, 2.7GHz quad-core intel core i7 processor and 16GB RAM hardware.

3.4.1 Experimental setup

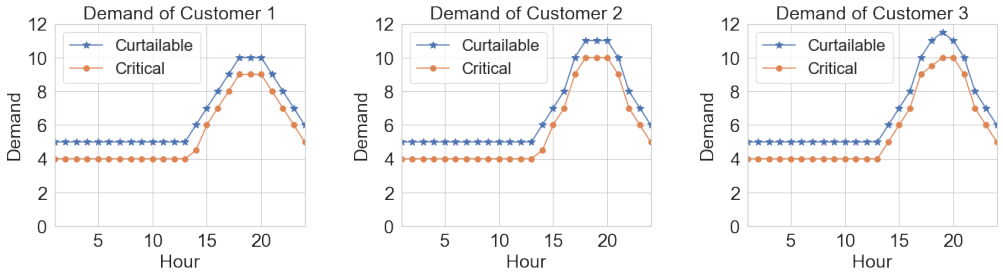
First, we test a simple DR case with three PCs for illustration purpose. In each episode with 24 hours, the wholesale price π_t, elasticity coefficient ξ_t, critical and curtailable demands $d_{t,n}^{crit}$

and $d_{t,n}^{curt}$ are depicted in Fig. 3.1 and 3.2. Wholesale price is set to be higher, and elasticity to be lower during peak hours.



(a) Wholesale price (b) Elasticity

Figure 3.1: Wholesale price and elasticity



(a) Customer 1 (b) Customer 2 (c) Customer 3

Figure 3.2: Customer demands within an episode

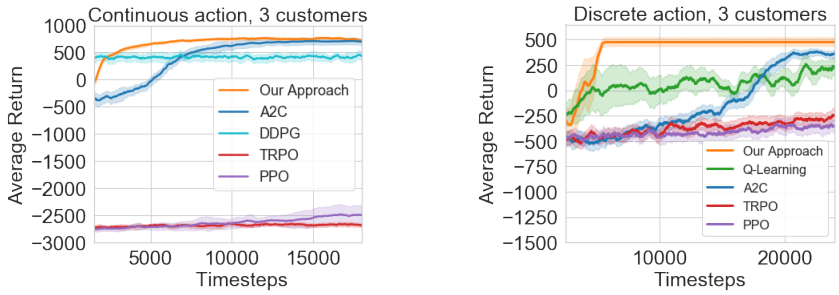
3.4.2 Performance and final policy

In this subsection, we present the performance and final policy of our approach versus baseline algorithms on the electricity market models. For the continuous action model, any retail prices ranging from 0 to 12 are allowed. For the discrete action model, we adopt a discrete action space where the retail prices can only take discrete values among $0, 0.5, 1, \ldots, 11.5, 12$.

The performance comparisons are provided in Fig. 3.3. As shown in Fig. 3.3, in both continuous and discrete action model, our approach converges faster and reaches a better

final performance than baseline algorithms. This suggests that the use of nonparametric policy representation indeed finds a more optimal final policy.

We provide the final pricing strategy of our approach in Fig. 3.4, from which we can see that the retail prices are higher during peak hours when wholesale price is high. We also analyze how this pricing strategy affects the usage in Fig. 3.5, which shows that the load reduction follows a similar trend as the unit profit. Also, the load reduction is higher during peak hours, indicating that our pricing strategy effectively controls peak-hour usages.

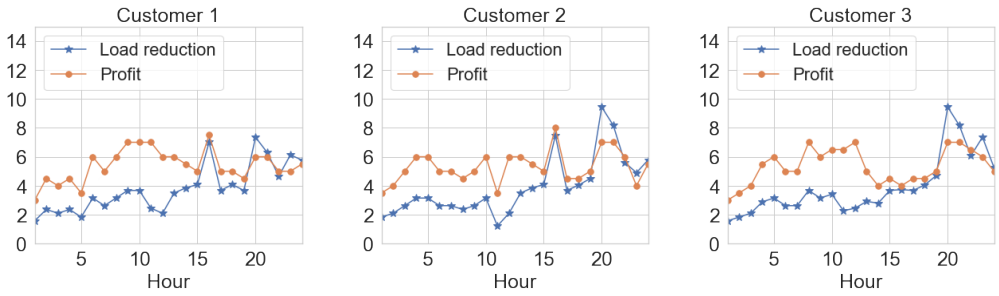


(a) Continuous action model (b) Discrete action model

Figure 3.3: Episode rewards during the training process, averaged across 3 runs with a random initialization. The shaded area depicts the mean ± the standard deviation.

(a) Customer 1 (b) Customer 2 (c) Customer 3

Figure 3.4: Final retail prices for each customer within an episode, continuous action model.



(a) Customer 1 (b) Customer 2 (c) Customer 3

Figure 3.5: Final profit (retail minus wholesale price) and load reduction within an episode, continuous action model.

3.4.3 Extension to large-scale action space

We further extend our experiments to a larger scale continuous action model with 30 customers. We consider continuous action because it is more general and closer to practical use case. We simulate customer demands by adding small variance to Fig. 3.2. The performance comparison in Fig. 3.6 shows similar findings as small-scale case: Our approach converges faster and has a higher performance compared to baseline algorithms.

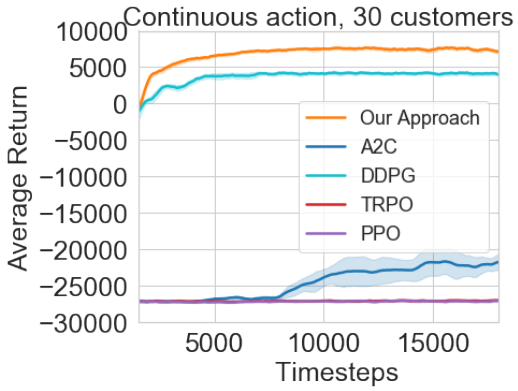


Figure 3.6: Episode rewards during the training process, continuous action model, averaged across 3 runs with a random initialization. The shaded area depicts the mean ± the standard deviation.

3.5 Conclusion

In this paper, we present a novel nonparametric constrained policy optimization RL algorithm, which addresses the sub-optimality issue of traditional gradient based policy optimization methods such like TRPO, PPO, DDPG and A2C. Our approach improves TRPO and PPO with a better final performance, while maintaining the stable learning property.

To our knowledge, our approach is the first policy optimization method that learns nonparametric policies in DR settings. Experiments on two DR cases show the superior performance of our proposed method compared with state-of-the-art RL algorithms.

Chapter 4

WASSERSTEIN POLICY OPTIMIZATION: A NONPARAMETRIC, PROVABLY CONVERGENCE CONSTRAINED POLICY OPTIMIZATION APPROACH

Trust-region methods based on Kullback-Leibler divergence are pervasively used to stabilize policy optimization in model-free reinforcement learning. In this chapter, we exploit more flexible metrics and examine two natural extensions of policy optimization with Wasserstein and Sinkhorn trust regions, namely *Wasserstein policy optimization (WPO)* and *Sinkhorn policy optimization (SPO)*. Instead of restricting the policy to a parametric distribution class, we directly optimize the policy distribution and derive their close-form policy updates based on the Lagrangian duality. Theoretically, we show that WPO guarantees a monotonic performance improvement, and SPO provably converges to WPO as the entropic regularizer diminishes. Moreover, we prove that with a decaying Lagrangian multiplier to the trust region constraint, both methods converge to global optimality. Experiments across tabular domains, robotic locomotion, and continuous control tasks further demonstrate the performance improvement of both approaches, more robustness of WPO to sample insufficiency, and faster convergence of SPO, over state-of-art policy gradient methods.

4.1 Introduction

Policy-based reinforcement learning (RL) approaches have received remarkable success in many domains, including video games [168, 268, 277], robotics [97, 146], and continuous control tasks [72, 107, 221]. One prominent example is policy gradient method [97, 151, 168, 199, 227, 243, 276]. The core idea is to represent the policy with a probability distribution $\pi_\theta(a|s) = P[a|s;\theta]$, such that the action a in state s is chosen stochastically following the policy π_θ controlled by parameter θ. Determining the right step size to update the policy is crucial for maintaining the stability of policy gradient methods: too conservative choice of stepsizes result in slow convergence, while too large stepsizes may lead to catastrophically

bad updates.

To control the size of policy updates, Kullback-Leibler (KL) divergence is commonly adopted to measure the difference between two policies. For example, the seminal work on trust region policy optimization (TRPO) by [219] introduced KL divergence based constraints (trust region constraints) to restrict the size of the policy update; see also [5, 198]. [130] and [222] introduced a KL-based penalty term to the objective to prevent excessive policy shift.

Though KL-based policy optimization has achieved promising results, it remains interesting whether using other metrics to gauge the similarity between policies could bring additional benefits. Recently, a few work [177, 189, 206, 300] has explored the Wasserstein metric to restrict the deviation between consecutive policies. Compared with KL divergence, the Wasserstein metric has several desirable properties. Firstly, it is a true symmetric distance measure. Secondly, it allows flexible user-defined costs between actions and is less sensitive to ill-posed likelihood ratios. Thirdly but most importantly, the Wasserstein metric takes into account the geometry of the metric space [190] and allows distributions to have different or even non-overlapping supports.

Motivating Example: Below we provide an example of a grid world (see Figure 4.1) that illustrates the advantages of using the Wasserstein metric over the KL divergence to construct trust regions and policy updates. The grid world consists of 5 regular grids and 2 goal grids, and there are three possible actions: left, right, and pickup. The player always starts from the middle grid, and making a left or right move results in a reward of -1. Picking up yields a reward of -3 at regular grids, $+5$ at the blue goal grid, and $+10$ at the red goal grid. An episode terminates either at the maximum length of 10 or immediately after picking up. We define the geometric distance between left and right actions to be 1, and 4 between other actions.

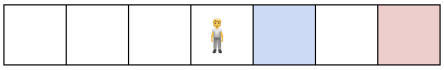


Figure 4.1: Motivating grid world example

(a) Policy shift of close action

(b) Policy shift of far action

Figure 4.2: Wasserstein utilizes geometric feature of action space

Figure 4.2 shows the Wasserstein distance and KL divergence for different policy shifts of this grid world example. We can see that Wasserstein metric utilizes the geometric distance between actions to distinguish the shift of policy distribution to a close action (policy distribution 1 $\rightarrow$ 2 in Figure 4.2a) from the shift to a far action (policy distribution 1 $\rightarrow$ 3 in Figure 4.2b), while KL divergence does not. Figure 4.3 demonstrates the constrained policy updates based on Wasserstein distance and KL divergence respectively with a fixed trust region size 1. We can see that Wasserstein-based policy update finds the optimal policy faster than KL-based policy update. This is because KL distance is larger than Waserstein when considering policy shifts of close actions (see Figure 4.2a). Therefore, Wasserstein policy update is able to shift action (from left to right) in multiple states, while KL update is only allowed to shift action in a single state. Besides, KL policy update keeps using a suboptimal short-sighted solution between the 2nd and 4th iteration, which further slows down the convergence.

However, the challenge of applying the Wasserstein metric for policy optimization is also evident: evaluating the Wasserstein distance requires solving an optimal transport problem, which could be computationally expensive. To avoid this computation hurdle, existing work resorts to different techniques to *approximate the policy update* under Wasserstein regularization. For example, [206] solved the resulting RL problem using Fokker-Planck equations; [300] introduced particle approximation method to estimate the Wasserstein gradient flow. Recently, [177] instead considered the second-order Taylor expansion of Wasserstein distance based on Wasserstein information matrix to characterize the local

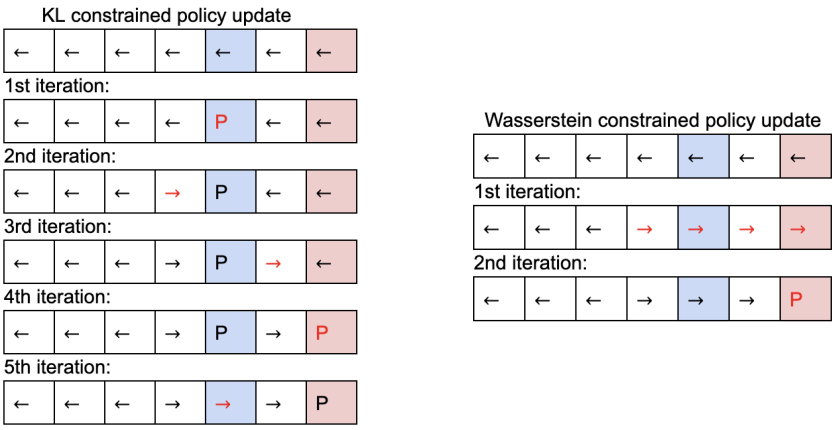


Figure 4.3: Demonstration of policy updates under different trust regions

behavioral structure of policies. [189] tackled behavior-guided policy optimization with smooth Wasserstein regularization by solving an approximate dual reformulation defined on reproducing kernel Hilbert spaces. Aside from such approximation, some of these work also limits the policy representation to a particular parametric distribution class, As indicated in [248], since parametric distributions are not convex in the distribution space, optimizing over such distributions results in local movements in the action space and thus leads to convergence to a sub-optimal solution. Until now, the theoretical performance of policy optimization under the Wasserstein metric remains elusive in light of these approximation errors.

In this paper, we study policy optimization with trust regions based on Wasserstein distance and Sinkhorn divergence. The latter is a smooth variant of Waserstein distance by imposing an entropic regularization to the optimal transport problem [58]. We call them, *Wasserstein Policy Optimization (WPO)* and *Sinkhorn Policy Optimization (SPO)*, respectively. Instead of confining the distribution of policy to a particular distribution class, we work on the space of policy distribution directly, and consider all admissible policies that are within the trust regions with the goal of avoiding approximation errors. Unlike existing work, we focus on *exact characterization* of the policy updates. We would like to emphasize that our methodology and theoretical analysis in Section 4.3, 4.4, and 4.5

primarily concentrate on a discrete action space. However, we also present an extension of our method to accommodate a continuous action space, detailed in Section 4.7.5. We highlight our contributions as follows:

1. **Algorithms:** We develop closed-form expressions of the policy updates for both WPO and SPO based on the corresponding optimal Lagrangian multipliers of the trust region constraints. To the best of our knowledge, this is the first explicit closed-form updates for policy optimization based on Wasserstein and Sinkhorn trust regions. In particular, the optimal Lagrangian multiplier of SPO admits a simple form and can be computed efficiently. A practical on-policy actor-critic algorithm is proposed based on the derived expressions of policy updates and advantage value function estimation.

2. **Theory:** We theoretically show that WPO guarantees a *monotonic performance improvement* through the iterations, *even with non-optimal Lagrangian multipliers*. We also prove that SPO converges to WPO as the entropic regularizer diminishes. Moreover, we prove that with a decaying schedule of the multiplier, SPO and WPO converge to *global optimality*, and with a constant multiplier, both methods converge *linearly* up to a neighborhood of the optimal value. To our best knowledge, this appears to be the first convergence rate analysis of policy optimization based on Wasserstein-type metrics.

3. **Experiments:** We provide comprehensive evaluation on the efficiency of WPO and SPO under several types of testing environments including tabular domains, robotic locomotion tasks, and further extend it to continuous control tasks. Compared to state-of-art policy gradients approaches that use KL divergence such as TRPO and PPO and those use Wasserstein metric such as Wasserstein Natural Policy Gradient (WNPG) [177] and Behavior Guided Policy Gradients (BGPG) [189], our methods achieve better sample efficiency, faster convergence, and improved final performance. Numerical study indicates that by properly choosing the weight of entropic regularizer, SPO achieves a better trade-off between convergence and final performance than WPO.

Related work: Wasserstein-like metrics have been explored in a number of works in the context reinforcement learning. [81] first introduced bisimulation metrics based on

Wasserstein distance to quantify behavioral similarity between states for the purpose of state aggregation. Such bisimulation metrics were recently utilized for representation learning of RL; see e.g., [8, 43]. In addition, a few recent work has also exploited Wasserstein distance for imitation learning (see e.g., [59, 278]) and unsupervised RL (see e.g., [106]). Our work is closely related to several previous studies, including [177, 189, 206, 300], which also utilize Wasserstein distance to measure the proximity of policies. However, unlike the aforementioned studies that solely employ Wasserstein distance as an explicit penalty function, we additionally utilize it as a trust region constraint. Moreover, we consider nonparametric policies and derive explicit policy update forms, whereas these studies update parametric policies using policy gradients. Furthermore, we demonstrate monotonic performance improvement and global convergence with our policy update, which is not provided in these previous works. Regarding the use of Sinkhorn divergence in RL, [189] is the only related work to our best knowledge, where the entropy regularization is used to mitigate the computational burden of computing Wasserstein metric. However, no explicit form of policy update is provided in this work, while we derive an explicit Sinkhorn policy update and demonstrate its advantage in convergence speed. Additionally, we use Wasserstein distance to directly measure the proximity of nonparametric policies in the distribution space, while [177, 189] measure the similarity of parametric policies in the behavioral space.

Wasserstein-like metrics are also pervasively studied in distributionally robust optimization (DRO); see e.g., [36, 78, 85, 309]. We also point out that a recent concurrent work by [262] studied DRO using the Sinkhorn distance. Our duality formulations are largely inspired from existing work in DRO. However, we note that constrained policy optimization is conceptually different from DRO. Constrained policy optimization focuses on finding the optimistic policy that falls in a trust region, whereas DRO (e.g., the KL DRO) aims to optimize some worst-case loss given by the adversarial distribution of unknown parameters within some ambiguity set.

4.2 Background and Notations

Markov Decision Process (MDP): We consider an infinite-horizon discounted MDP, defined by the tuple $(\mathcal{S}, \mathcal{A}, P, r, \upsilon, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the transition probability, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function,

$\upsilon : \mathcal{S} \to \mathbb{R}$ is the distribution of the initial state s_0, and $\gamma \in (0, 1)$ is the discount factor. We define the return of timestep t as the accumulated discounted reward from t, $R_t = \sum_{k=0}^{\infty} \gamma^k r(s_{t+k}, a_{t+k})$, and the value function as $V^\pi(s) = \mathbb{E}[R_t | s_t = s; \pi]$. The performance of a stochastic policy π is defined as $J(\pi) = \mathbb{E}_{s_0, a_0, s_1 \ldots}[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]$ where $a_t \sim \pi(a_t|s_t)$, $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$. As shown in [129], the expected return of a new policy π' can be expressed in terms of the advantage over the old policy π: $J(\pi') = J(\pi) + \mathbb{E}_{s \sim \rho_\upsilon^{\pi'}, a \sim \pi'}[A^\pi(s, a)]$, where $A^\pi(s, a) = \mathbb{E}[R_t | s_t = s, a_t = a; \pi] - \mathbb{E}[R_t | s_t = s; \pi]$ represents the advantage function and ρ_υ^π represents the unnormalized discounted visitation frequencies with initial state distribution υ, i.e., $\rho_\upsilon^\pi(s) = \mathbb{E}_{s_0 \sim \upsilon}[\sum_{t=0}^{\infty} \gamma^t P(s_t = s | s_0)]$.

Trust Region Policy Optimization (TRPO): In TRPO [219], the policy π is parameterized as π_θ with parameter vector θ. For notation brevity, we use θ to represent the policy π_θ. Then, the new policy θ' is found in each iteration to maximize the expected improvement $J(\pi') - J(\pi)$, or equivalently, the expected value of the advantage function:

$$\begin{aligned} \max_{\theta'} \quad & \mathbb{E}_{s \sim \rho_\upsilon^\theta, a \sim \theta'}[A^\theta(s, a)] \\ \text{s.t.} \quad & \mathbb{E}_{s \sim \rho_\upsilon^\theta}[d_{\mathrm{KL}}(\theta', \theta)] \leq \delta, \end{aligned} \tag{4.1}$$

where d_{KL} represents the KL divergence and δ is the threshold of the distance between new and old policies.

Wasserstein Distance: Given two probability distributions of policies π and π' on the discrete action space $\mathcal{A} = \{a_1, a_2, \ldots, a_N\}$, the Wasserstein distance between the policies is defined as:

$$d_{\mathrm{W}}(\pi', \pi) = \inf_{Q \in \Pi(\pi', \pi)} \langle Q, D \rangle, \tag{4.2}$$

where $\langle \cdot, \cdot \rangle$ denotes the Frobenius inner product. The infimum is taken over all joint distributions Q with marginals π' and π, and D is the cost matrix with $D_{ij} = d(a_i, a_j)$, where $d(a_i, a_j)$ is defined as the distance between actions a_i and a_j. Its largest entry in magnitude is denoted by $\|D\|_\infty$. In implementation, our choice of distance d is task-dependent and is reported in Table C.1 in Appendix C.1.

Sinkhorn Divergence: Sinkhorn divergence [58] provides a smooth approximation of the Wasserstein distance by adding an entropic regularizer. The Sinkhorn divergence is defined as:

$$d_{\mathrm{S}}(\pi', \pi|\lambda) = \inf_{Q \in \Pi(\pi', \pi)} \left\{ \langle Q, D \rangle - \frac{1}{\lambda} h(Q) \right\}, \tag{4.3}$$

where $h(Q) = -\sum_{i=1}^{N} \sum_{j=1}^{N} Q_{ij} \log Q_{ij}$ represents the entropy term, and $\lambda > 0$ is a regularization parameter. The intuition of adding the entropic regularization is: since most elements of the optimal joint distribution Q will be 0 with a high probability, by trading the sparsity with entropy, a smoother and denser coupling between distributions can be achieved [55, 56]. Therefore, when the weight of the entropic regularization decreases (i.e., λ increases), the sparsity of the divergence increases, and the Sinkhorn divergence converges to the Wasserstein metric, i.e., $\lim_{\lambda \to \infty} d_{\mathrm{S}}(\pi', \pi|\lambda) = d_{\mathrm{W}}(\pi', \pi)$. More critically, Sinkhorn divergence is useful to mitigate the computational burden of computing Wasserstein distance. In fact, the efficiency improvement that Sinkhorn divergence and the related algorithms brought paves the way to utilize Wasserstein-like metrics in many machine learning domains, including online learning [47], model selection [128, 207], generative modeling [87, 193, 200], dimensionality reduction [118, 153, 263].

4.3 Wasserstein Policy Optimization

Motivated by TRPO, here we consider a trust region based on the Wasserstein metric. Moreover, we lift the restrictive assumption that a policy has to follow a parametric distribution class by allowing all admissible policies. Then, the new policy π' is found in each iteration to maximize the estimated expected value of the advantage function. Therefore, the *Wasserstein Policy Optimization* (WPO) framework is shown as follows:

$$\begin{aligned} &\max_{\pi' \in \mathcal{D}} \quad \mathbb{E}_{s \sim \rho_v^{\pi}, a \sim \pi'(\cdot|s)}[A^{\pi}(s, a)] \\ &\text{where} \quad \mathcal{D} = \{\pi' | \mathbb{E}_{s \sim \rho_v^{\pi}}[d_{\mathrm{W}}(\pi'(\cdot|s), \pi(\cdot|s))] \leq \delta\}, \end{aligned} \tag{4.4}$$

where the Wasserstein distance $d_{\mathrm{W}}(\cdot, \cdot)$ is defined in (4.2).

In most practical cases, the reward r is bounded and correspondingly, the accumulated discounted reward R_t is bounded. So without loss of generality, we make the following assumption:

Assumption 1. *Assume $A^\pi(s,a)$ is bounded, i.e., $\sup_{a\in\mathcal{A},s\in\mathcal{S}}|A^\pi(s,a)|\leq A^{max}$ for some $A^{max}>0$.*

With Wasserstein metric based trust region constraint, we are able to derive the closed-form of the policy update shown in Theorem 5. The main idea is to form the Lagrangian dual of the constrained optimization problem presented above, which is inspired by the way to obtain the extremal distribution in Wasserstein DRO literature, see e.g., [36, 140, 309]. The detailed proof can be found in Appendix C.2.

Theorem 5. ***(Closed-form policy update)*** *Let $\kappa_s^\pi(\beta,j)=argmax_{k=1\ldots N}\{A^\pi(s,a_k)-\beta D_{kj}\}$, where D denotes the cost matrix. If Assumption 1 holds, then an optimal solution to (4.4) is:*

$$\pi^*(a_i|s)=\sum\nolimits_{j=1}^{N}\pi(a_j|s)f_s^*(i,j), \tag{4.5}$$

where $f_s^*(i,j)=1$ if $i=\kappa_s^\pi(\beta^*,j)$ and $f_s^*(i,j)=0$ otherwise, and β^* is an optimal Lagrangian multiplier corresponds to the following dual formulation:

$$\min_{\beta\geq 0}\ F(\beta)=\min_{\beta\geq 0}\{\beta\delta+\mathbb{E}_{s\sim\rho_v^\pi}\sum\nolimits_{j=1}^{N}\pi(a_j|s)\max_{i=1\ldots N}(A^\pi(s,a_i)-\beta D_{ij})\}. \tag{4.6}$$

Moreover, we have $\beta^*\leq\bar{\beta}$, where $\bar{\beta}:=\max_{s\in\mathcal{S},k,j=1\ldots N,k\neq j}(D_{kj})^{-1}(A^\pi(s,a_k)-A^\pi(s,a_j))$.

Remark 1. *For ease of notation and simplicity, we assume the uniqueness of $\kappa_s^\pi(\beta,j)$ in order to form the simple expression of f_s^* in Theorem 5. When it is not unique, a necessary condition for the optimality of π^* in (4.5) is $\sum_{i\in\mathcal{K}_s^\pi(\beta,j)}f_s^*(i,j)=1$, and $f_s^*(i,j)=0$ for $i\notin\mathcal{K}_s^\pi(\beta,j)$, where $\mathcal{K}_s^\pi(\beta,j)=argmax_{k=1\ldots N}A^\pi(s,a_k)-\beta D_{kj}$. The weight $f_s^*(i,j)$ for $i\in\mathcal{K}_s^\pi(\beta,j)$ could be determined through linear programming (see details in (C.4) in Appendix C.2).*

The exact policy update for WPO in (4.5) requires computing the optimal Lagrangian multiplier β^* by solving the one-dimensional subproblem (4.6). A closed-form of β^* is not easy to obtain in general, except for special cases of the distance $d(x,y)$ or cost matrix D. In Appendix C.3, we provide the closed-form of β^* for the case when $d(x,y)=0$ if $x=y$ and 1 otherwise.

WPO Policy Update: Based on Theorem 5, we introduce the following WPO policy

updating rule:

$$\pi_{k+1}(a_i|s) = \mathbb{F}^{\text{WPO}}(\pi_k) := \sum_{j=1}^{N} \pi_k(a_j|s) f_s^k(i,j), \tag{WPO}$$

where $f_s^k(i,j) = 1$ if $i = \kappa_s^{\pi_k}(\beta_k, j)$ and 0 otherwise. Note that different from (4.5), we allow β_k to be chosen arbitrarily and time dependently. We show that such policy update always leads to a monotonic improvement of the performance even when β_k is not the optimal Lagrangian multiplier. In particular, we propose two strategies to update multiplier β_k:

(i) Approximation of optimal β_k: To improve the convergence, we can approximately solve the optimal Lagrangian multiplier based on Sinkhorn divergence. More details in Section 4.4.

(ii) Time-dependent β_k: To improve the computational efficiency, we can simply treat β_k as a time-dependent parameter, e.g., we can set β_k as a diminishing sequence. In this setting, (WPO) produces the solution to the following penalty version of problem (4.4) (with $d = d_{\text{W}}$):

$$\max_{\pi_{k+1}} \mathbb{E}_{s\sim\rho_v^{\pi_k}, a\sim\pi_{k+1}(\cdot|s)}[A^{\pi_k}(s,a)] - \beta_k \mathbb{E}_{s\sim\rho_v^{\pi_k}}[d(\pi_{k+1}(\cdot|s), \pi_k(\cdot|s))]. \tag{4.7}$$

4.4 Sinkhorn Policy Optimization

In this section, we introduce Sinkhorn policy optimization (SPO) by constructing trust region with Sinkhorn divergence. In the following theorem, we derive the optimal policy update in each step when using Sinkhorn divergence based trust region. Detailed proofs are provided in Appendix C.4.

Theorem 6. *If Assumption 1 holds, then the optimal solution to (4.4) with Sinkhorn divergence is:*

$$\pi_\lambda^*(a_i|s) = \sum_{j=1}^{N} \pi(a_j|s) f_{s,\lambda}^*(i,j), \tag{4.8}$$

where D denotes the cost matrix, $f_{s,\lambda}^*(i,j) = \frac{\exp(\frac{\lambda}{\beta_\lambda^*} A^\pi(s,a_i) - \lambda D_{ij})}{\sum_{k=1}^N \exp(\frac{\lambda}{\beta_\lambda^*} A^\pi(s,a_k) - \lambda D_{kj})}$ and β_λ^* is an optimal

solution to the following dual formulation:

$$\min_{\beta\geq 0} F_\lambda(\beta) = \min_{\beta\geq 0}\Big\{\beta\delta - \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)(\frac{\beta}{\lambda}+\frac{\beta}{\lambda}\ln(\pi(a_j|s)) - \frac{\beta}{\lambda}\ln[\sum_{i=1}^{N}\exp(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij})]) \\ \mathbb{E}_{s\sim\rho_v^\pi}\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\beta}{\lambda}\frac{\exp(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij})\cdot\pi(a_j|s)}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj})}\Big\}. \tag{4.9}$$

Moreover, we have $\beta_\lambda^* \leq \frac{2A^{max}}{\delta}$.

In contrast to the Wasserstein dual formulation (4.6), the objective in the Sinkhorn dual formulation (4.9) is differentiable in β and admits closed-form gradients (shown in Appendix C.6). With this gradient information, we can use gradient-based global optimization algorithms [143, 260, 297] to find a global optimal solution β_λ^* to (4.9).

Next, we show that if the entropic regularization parameter λ is large enough, then the optimal solution β_λ^* is a close approximation to the β^* of Wasserstein dual formulation. Proof is provided in Appendix C.7.

Theorem 7. *Define* $\beta_{UB} = \max\{\frac{2A^{max}}{\delta}, \bar{\beta}\}$. *We have:*

1. $F_\lambda(\beta)$ *converges to* $F(\beta)$ *uniformly on* $[0, \beta_{UB}]$: $\lim_{\lambda\to\infty}\sup_{0\leq\beta\leq\beta_{UB}}\Big|F_\lambda(\beta)-F(\beta)\Big| \leq \lim_{\lambda\to\infty}\frac{\beta_{UB}}{\lambda}N\ln N = 0$.

2. $\lim_{\lambda\to\infty} argmin_{0\leq\beta\leq\beta_{UB}} F_\lambda(\beta) \subseteq argmin_{0\leq\beta\leq\beta_{UB}} F(\beta)$.

Although it is difficult to obtain the exact value of the optimal solution β^* to the Wasserstein dual formulation (4.6), the above theorem suggests that we can approximate β^* via β_λ^* by setting up a relative large λ. In practice, we can also adopt a smooth homotopy approach by setting an increasing sequence λ_k for each iteration and letting $\lambda_k \to \infty$.

SPO Policy Update: Based on Theorem 6, we introduce the following SPO policy updating rule:

$$\pi_{k+1}(a_i|s) = \mathbb{F}^{\text{SPO}}(\pi_k) = \sum_{j=1}^{N}\pi_k(a_j|s)f^k_{s,\lambda_k}(i,j). \tag{SPO}$$

Here $f^k_{s,\lambda_k}(i,j) = \frac{\exp(\frac{\lambda_k}{\beta_k}A^{\pi_k}(s,a_i)-\lambda_k D_{ij})}{\sum_{l=1}^{N}\exp(\frac{\lambda_k}{\beta_k}A^{\pi_k}(s,a_l)-\lambda_k D_{lj})}$, $\lambda_k \geq 0$ and $\beta_k \geq 0$ are some control parameters. The parameter β_k can be either computed via solving the one-dimensional subproblem (4.9) or simply set as a diminishing sequence. The proper setup of λ_k can effectively adjust the trade-off between convergence speed and final performance. More details are provided in the ablation study in Section 4.7.

4.5 Theoretical Analysis

We first show that SPO policy update converges to WPO policy update as the regularization parameter increases (i.e., $\lambda \to \infty$). The detailed proof is provided in Appendix C.8.

As $\lambda_k \to \infty$, SPO update converges to WPO update: $\lim_{\lambda_k \to \infty} \mathbb{F}^{\mathrm{SPO}}(\pi_k) \in \mathbb{F}^{\mathrm{WPO}}(\pi_k)$. We then provide a theoretical justification that WPO policy update (and SPO with $\lambda \to \infty$) are always guaranteed to improve the true performance J monotonically if we have access to the true advantage function. If the advantage function can only be evaluated inexactly with limited samples, then an extra estimation error (measured by the largest absolute entry $\|\cdot\|_\infty$) will occur. Proof can be found in Appendix C.9.

Theorem 8. ***(Performance improvement)*** *For any initial state distribution υ and any $\beta_k \geq 0$, if $||\hat{A}^\pi - A^\pi||_\infty \leq \epsilon$ for some $\epsilon > 0$, let $\hat{\mathcal{K}}^{\pi_k}_s(\beta_k, j) = argmax_{i=1,\ldots,N}\{\hat{A}^{\pi_k}(s,a_i) - \beta_k D_{ij}\}$, WPO policy update (and SPO with $\lambda \to \infty$) guarantee the following performance improvement bound when the inaccurate advantage function $\hat{A}^\pi$ is used,*

$$J(\pi_{k+1}) \geq J(\pi_k) + \beta_k \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{j=1}^{N} \pi_k(a_j|s) \sum_{i\in\hat{\mathcal{K}}^{\pi_k}_s(\beta_k,j)} f^k_s(i,j) D_{ij} - \frac{2\epsilon}{1-\gamma}. \tag{4.10}$$

The value of ϵ, which quantifies the approximation error of the advantage function, is dependent on various factors such as the advantage estimation algorithm used and the number of samples [221]. It is worth noting that the improvement bound of NPG/TRPO [46]includes the same additional term $-\frac{2\epsilon}{1-\gamma}$, which indicates that our methods offer comparable theoretical performance guarantees to KL based updates. In the following, we show that with a decreasing schedule of the multiplier β_k, both WPO and SPO policy updates have their values $J(\pi_k)$

converging to the optimal $J^\star = \max_\pi J(\pi)$ on the tabular domain. To start, for k-th iteration, we consider (WPO) and (SPO) (with arbitrary $\lambda > 0$) whose updates π_{k+1} are optimal solutions to (4.7) with d being d_{W} and d_{S} respectively.

Assumption 2. *The state space and the action space are both finite, the reward function r is non-negative, and the initial distribution covers all state.*

Note that once state and action spaces are both finite, the reward can be assumed non-negative without loss of generality, as we can always add $\max_{s,a} |r(s,a)|$ to the reward function without changing the optimal policy and the order of the policies. Defining the optimal value function $V^\star(s) = \max_\pi \mathbb{E}[R_t | s_t = s]$, we have the following theorem, whose proof is in Appendix C.10 and is inspired by [29].

Theorem 9. ***(Global convergence)*** *Under Assumption 2, we have for any $\beta_k \geq 0$, (WPO) satisfies that*

$$\|V^\star - V^{\pi_{k+1}}\|_\infty \leq \gamma \|V^\star - V^{\pi_k}\|_\infty + \beta_k \|D\|_\infty, \tag{4.11}$$

and (SPO) satisfies that

$$\|V^\star - V^{\pi_{k+1}}\|_\infty \leq \gamma \|V^\star - V^{\pi_k}\|_\infty + 2\frac{\beta_k}{1-\gamma}\left(\|D\|_\infty + 2\frac{\log N}{\lambda}\right). \tag{4.12}$$

If $\lim_{k\to\infty} \beta_k = 0$, we further have $\lim_{k\to\infty} J(\pi_k) = J^\star$.

Remark 2. *Note the convergence is* geometric. *If we keep β_k as a constant, then $0 \leq J^\star - J(\pi^T) \leq \|V^\star - V^{\pi^T}\|_\infty \leq \gamma^T \|V^\star - V^{\pi_0}\|_\infty + \frac{\beta B}{1-\gamma}$, where $B = \|D\|_\infty$ for (WPO) and $B = 2\frac{\|D\|_\infty + 2\frac{\log N}{\lambda}}{1-\gamma}$ for (SPO). To achieve an ϵ optimality gap, we only need to take $\beta = \frac{(1-\gamma)\epsilon}{2B}$ and let $T \geq \frac{\log(\epsilon/2)}{\gamma}$.*

Remark 3. *The study of global non-asymptotic convergence of nonconvex policy optimization algorithms has been an active research topic. Recent theoretical work has mostly centered on PG and natural policy gradient (NPG) [130] - a close relative of TRPO; see e.g., [7, 46, 142]. To our best knowledge, a few work has discussed the global convergence of TRPO. [181] and [86] established the connection of TRPO to Mirror Descent, but did not provide any non-asymptotic rate; [224] showed that adaptive TRPO with decaying stepsize achieved $O(1/\sqrt{T})$*

convergence rate for unregularized MDPs in the tabular setting (finite state and finite action). Our result seems to be the first non-asymptotic analysis of policy optimization based on Wasserstein and Sinkhorn divergence. It remains interesting to extend the convergence theory of TRPO/WPO/SPO to function approximation regime following recent advance [7]. However, this is beyond the scope of our current work, as we focus on explicit closed-form update of WPO/SPO, which can be a viable alternative to TRPO in practice.

4.6 A Practical Algorithm

In practice, the advantage value functions are often estimated from sampled trajectories. In this section, we provide a practical on-policy actor-critic algorithm, described in Algorithm 4, that combines WPO/SPO with advantage function estimation.

At each iteration, the first step is to collect trajectories, which can be either complete or partial. If the trajectory is complete, the total return can be directly expressed as the accumulated discounted rewards $R_t = \sum_{k=0}^{T-t-1} \gamma^k r_{t+k}$. If the trajectory is partial, it can be estimated by applying the multi-step temporal difference (TD) methods [62]: $\hat{R}_{t:t+n} = \sum_{k=0}^{n-1} \gamma^k r_{t+k} + \gamma^n V(s_{t+n})$. Then for the advantage estimation, we can use Monte Carlo advantage estimation, i.e., $\hat{A}_t^{\pi_k} = R_t - V_{\psi_k}(s_t)$ or Generalized Advantage Estimation (GAE) [221], which provides a more explicit control over the bias-variance trade-off. In the value update step, we use a neural net to represent the value function, where ψ is the parameter that specifies the value net $s \to V(s)$. Then, we can update ψ by using gradient descent, which significantly reduces the computational burden of computing advantage directly. The computational complexity of the algorithm is discussed in Appendix C.11.

4.7 Experiments

In this section, we evaluate the proposed WPO and SPO approaches presented in Algorithm 4. We compare the performance of our methods with benchmarks including TRPO [219], PPO [222], A2C [168]; and with BGPG [189], WNPG [177] for continuous control. The code of our WPO/SPO can be found here[1]. We adopt the implementations of TRPO, PPO and

[1] `https://github.com/efficientwpo/EfficientWPO`

Algorithm 4: On-policy WPO/SPO algorithm

Input: number of iterations K, learning rate α

Initialize policy π_0 and value network V_{ψ_0} with random parameter ψ_0

for $k = 0, 1, 2 \dots K$ **do**

Collect trajectory set $\mathcal{D}_k$ on policy π_k

For each timestep t in each trajectory, compute total returns G_t and estimate advantages $\hat{A}_t^{\pi_k}$

Update value:

$\psi_{k+1} \leftarrow \psi_k - \alpha \nabla_{\psi_k} \sum (G_t - V_{\psi_k}(s_t))^2$

Update policy:

$\pi_{k+1} \leftarrow \mathbb{F}(\pi_k)$ via WPO/ SPO with $\hat{A}_t^{\pi_k}$

end

A2C from OpenAI Baselines [67] for MuJuCo tasks and Stable Baselines [111] for other tasks. For BGPG, we adopt the same implementation[2] as [189].

Our experiments include (1) ablation study that focuses on sensitivity analysis of WPO and SPO; (2) tabular domain tasks with discrete state and action including the Taxi, Chain, and Cliff Walking environments; (3) locomotion tasks with continuous state and discrete action including the CartPole, Acrobot environments; (4) comparison of KL and Wasserstein trust regions under tabular domain and locomotion tasks; and (5) extension to continuous control tasks with continuous action including HalfCheetah, Hopper, Walker, and Ant environments from MuJuCo. See Table C.2 in Appendix C.1 for a summary of performance. The setting of hyperparameters and network sizes of our algorithms and additional results are provided in Appendix C.1.

4.7.1 Ablation Study

In this experiment, we first examine the sensitivity of WPO in terms of different strategies of β_k. We test four settings of β value for WPO policy update: (1) Setting 1: Computing

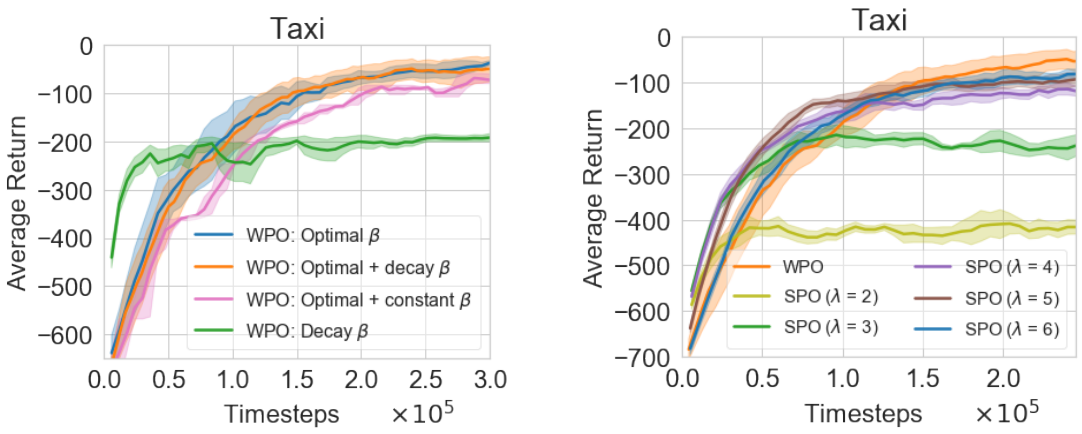
[2] `https://github.com/behaviorguidedRL/BGRL`

optimal β value for all policy update; (2) Setting 2: Computing optimal β value for first 20% of policy updates and decaying β for the remaining; (3) Setting 3: Computing optimal β value for first 20% of policy updates and fix β as its last updated value for the remaining; (4) Setting 4: Decaying β for all policy updates (e.g., $\beta_k = \Theta(1/\log k)$). In particular, Setting 2 is rooted in the observation that β^* decays slowly in the later stage of the experiments carried out in the paper. Small perturbations are added to the approximate values to avoid any stagnation in updating. Taxi task [68] from tabular domain is selected for this experiment.

The performance comparisons and average run times are shown in Figure 4.4 and Table 4.1 respectively. Figure 4.4a and Table 4.1 clearly indicate a tradeoff between computation efficiency and accuracy in terms of different choices of β value. Setting 2 is the most effective way to balance the tradeoff between performance and run time. For the rest of experiments, we adopt this setting for both WPO and SPO (see Appendix C.1.2 for how Setting 2 is tuned for each task). Figure 4.4b shows that as λ increases, the convergence of SPO becomes slower but the final performance of SPO improves and becomes closer to that of WPO, which verifies the convergence property of Sinkhorn to Wasserstein distance shown in Theorem 7. Therefore, the choice of λ can effectively adjust the trade-off between convergence and final performance. Similar results are observed when using time-varying λ on Taxi, Chain and CartPole tasks, presented in Figure C.1 in Appendix C.1.

Table 4.1: Runtime for different β settings, average across 5 runs with random initialization

Runtime	Taxi (s)	CartPole (s)
Setting 1 (optimal β)	1224.3 $\pm$ 105.7	129.7 $\pm$ 15.2
Setting 2 (optimal-then-decay)	648.4 $\pm$ 55.7	63.2 $\pm$ 8.3
Setting 3 (optimal-then-fix)	630.2 $\pm$ 67.4	67.1 $\pm$ 9.7
Setting 4 (decaying β)	522.7 $\pm$ 49.5	44.3 $\pm$ 6.2



(a) Different settings of β

(b) Different choices of λ

Figure 4.4: Episode rewards for Taxi with different β and λ settings, averaged across 5 runs with random initialization. The shaded area depicts the mean $\pm$ the standard deviation.

4.7.2 Tabular Domains

We evaluate WPO and SPO on tabular domain tasks and test the exploration ability of the algorithms on several environments including Taxi, Chain, and Cliff Walking. We use a table of size $|\mathcal{S}| \times |\mathcal{A}|$ to represent the policy $\pi(a|s)$. For the value function, we use a neural net to smoothly update the values. The performance of WPO and SPO are compared to the performance of TRPO, PPO and A2C under the same neural net structure. Results on Taxi, Cliff and Chain are reported in Figure 4.5.

As shown in Figure 4.5, the performances of WPO, SPO and TRPO are manifestly better than A2C and PPO. Among the trust region based methods, WPO and SPO outperform TRPO in Taxi and Cliff Walking, whereas in Chain, the performances of these three methods are comparable. In all of the test cases, SPO converges faster than WPO but to a lower optimum. As further shown in Table 4.2, for the Taxi environment, WPO has a higher successful drop-off rate and a lower task completion time while the original TRPO reaches the time limit with a drop-off rate 0, suggesting that WPO finds a better policy than the original TRPO. In Figure 4.7, we also compare the performance of WPO under Wasserstein

and KL divergences given different number of samples N_A used to estimate the advantage function, and the result suggests that using Wasserstein metric is more robust than KL divergence under inaccurate advantage values.

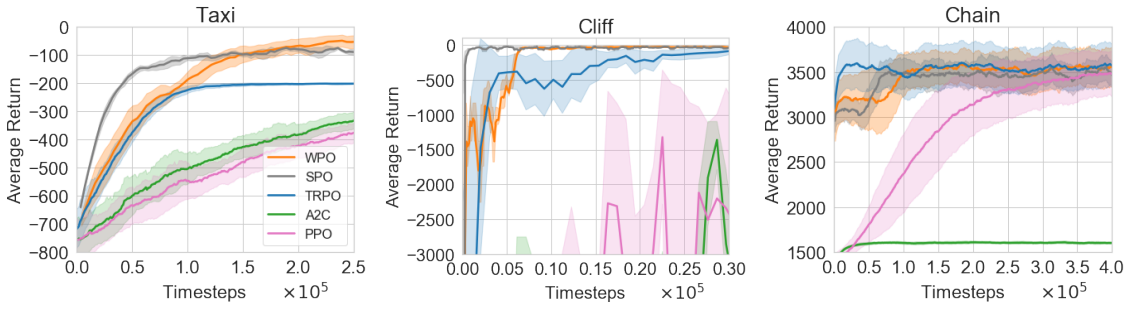


Figure 4.5: Episode rewards during training for tabular domain tasks, averaged across 5 runs with random initialization. The shaded area depicts the mean ± the standard deviation.

Table 4.2: Trained agents performance on Taxi

	Success (+20)	Fail (-10)	Steps (-1)	Return
WPO	0.753	0.232	70.891	-58.151
TRPO	0	0	200	-200

4.7.3 Robotic Locomotion Tasks

We now integrate deep neural network architecture into WPO and SPO and evaluate their performance on several locomotion tasks (with continuous state and discrete action), including CartPole [23] and Acrobot [88]. We use two separate neural nets to represent the policy and the value. The policy neural net receives state s as an input and outputs the categorical distribution of $\pi(a|s)$. A random subset of states $\mathcal{S}_k \in \mathcal{S}$ is sampled at each iteration to perform policy updates.

Figure 4.6 shows that WPO and SPO outperform TRPO, PPO and A2C in most tasks in terms of final performance, except in Acrobot where PPO performs the best. In most cases, SPO converges faster but WPO has a better final performance. To train 10^5 timesteps in the discrete locomotion tasks, the training wall-clock time is 63.2 $\pm$8.2s for WPO, 64.7 $\pm$7.8s

for SPO, 59.4 $\pm 10.2s$ for TRPO and 69.9 $\pm 10.5s$ for PPO. Therefore, WPO has a similar computational efficiency as TRPO and PPO.

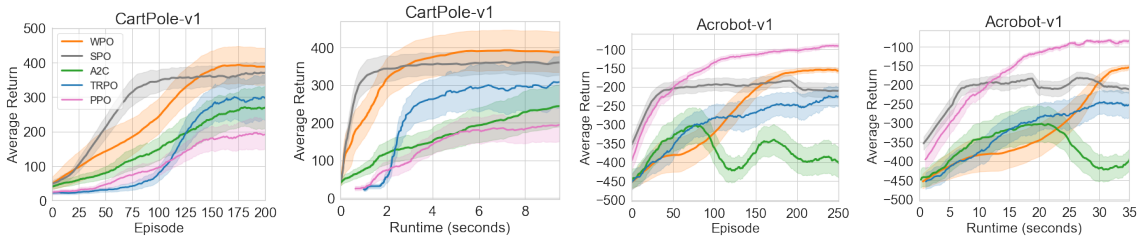
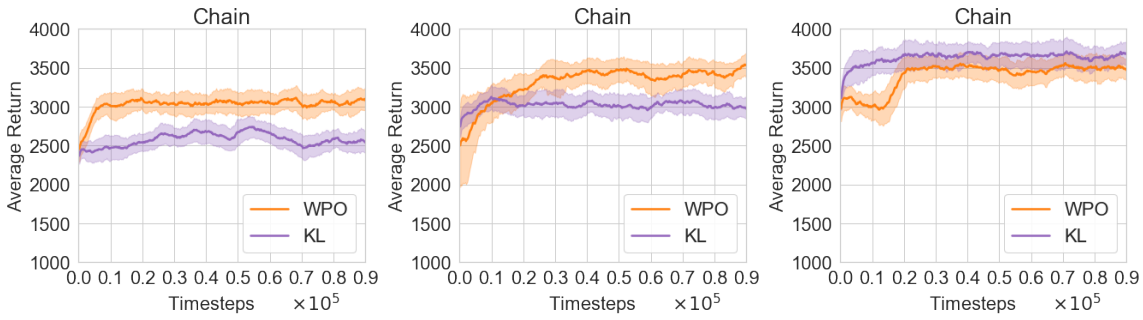


Figure 4.6: Episode rewards during the training process for the locomotion tasks, averaged across 5 runs with random initialization. The shaded area depicts the mean ± the standard deviation.

4.7.4 Comparison of Wasserstein and KL Trust Regions

We show that compared with the KL divergence, the utilization of Wasserstein metric can cope with the inaccurate advantage estimations caused by the lack of samples. Let N_A denote the number of samples used to estimate the advantage function. We evaluate the performance of WPO framework (4.4) with Wasserstein and KL constraints (as derived in [198]). We consider the Chain task and different N_A. As shown in Figure 4.7, when N_A is 1000, KL performs slightly better than WPO. However, when N_A decreases to 100 or 250, WPO outperforms KL. These results indicate that WPO is more robust than KL under inaccurate advantage values. This finding is consistent with our observations on the policy update formulations of Wasserstein and KL. For the Wasserstein update in (4.5), policy will be updated only when the advantage difference between two actions is significant, i.e., $A^\pi(s, a_j) - \beta D_{ij} \geq A^\pi(s, a_i)$. However, for the KL update in [198], policy will be updated as long as the current advantage function has a single non-zero value. Therefore, KL update is more sensitive; while Wasserstein update is more robust and more tolerant to advantage inaccuracies. Similar results are obtained for the locomotion tasks (Figure C.2 in Appendix C.1). The runtime of Wasserstein and KL updates are reported in Table C.3 in Appendix C.1.



(a) $N_A = 100$ (b) $N_A = 250$ (c) $N_A = 1000$

Figure 4.7: Episode rewards during training for the Chain task, where advantage value function is estimated under different number of samples, averaged across 5 runs with random initialization. The shaded area depicts the mean ± the standard deviation.

4.7.5 Extension to Continuous Control

To extend to environments with continuous action, we use Implicit Quantile Networks (IQN) [275] actor that can represent an arbitrary complex non-parametric policy. Let $F_s^{-1}(p)$ represent the quantile function associated with policy $\pi(\cdot|s)$. The IQN actor takes state s and probability $p \in [0, 1]$ as input, and outputs the corresponding quantile value $a = F_s^{-1}(p)$. IQN actor can be trained to approach pre-defined target policy distributions through quantile regression [248, 275].

Define the action support for state s in k-th iteration as $I^{\pi_k}(s) = \{a' : A^{\pi_k}(s, a') > \min_{a \in I^{\pi_{k-1}}(s)} A^{\pi_k}(s, a)\}$. Then, the WPO/SPO target policy distribution to guide IQN update in the k-th iteration is:

$$P_{I^{\pi_k}(s)}(a'|s) = \sum\nolimits_{a \in I^{\pi_{k-1}}(s)} \pi_k(a|s) f_s(a', a), \tag{4.13}$$

where for WPO update $f_s(a', a) = 1$ if $a' = \text{argmax}_{a' \in I^{\pi_k}(s)} \{A^{\pi_k}(s, a') - \beta_k d(a', a)\}$ and $f_s(a', a) = 0$ otherwise; for SPO update, $f_s(a', a) = \frac{\exp(\frac{\lambda_k}{\beta_k} A^{\pi_k}(s,a') - \lambda_k d(a',a))}{\sum_{a' \in I^{\pi_k}(s)} \exp(\frac{\lambda_k}{\beta_k} A^{\pi_k}(s,a') - \lambda_k d(a',a))}$. In implementation, we sample a batch of states $\mathcal{S}_k \in \mathcal{S}$ at each iteration to perform policy updates, and for each $s \in \mathcal{S}_k$, we sample $|\mathcal{A}_k|$ actions to approximate the support $I^{\pi_k}(s)$ and the target policy distribution $P_{I^{\pi_k}(s)}(\cdot|s)$.

We additionally compare WPO and SPO with BGPG [189] and WNPG [177] that are specially designed to address the continuous control with Wasserstein metric, for several MuJuCo tasks including HalfCheetah, Hopper, Walker, and Ant. Figure 4.8 shows that WPO and SPO have consistently better performances than other benchmarks. Similar results are obtained for the challenging Humanoid task, presented in Figure C.3 in Appendix C.1. We also provide the runtime of each algorithm in Table C.4 in Appendix C.1.

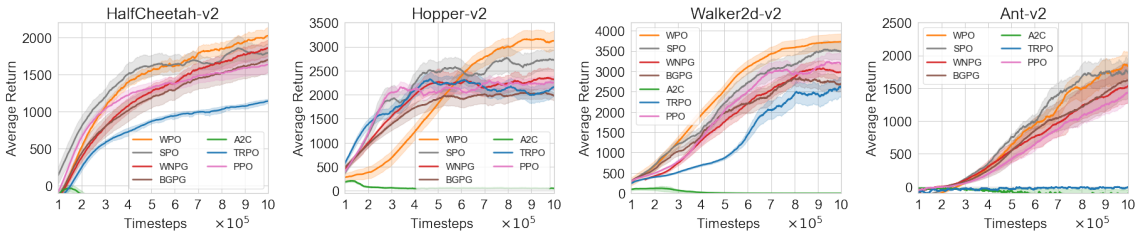


Figure 4.8: Episode rewards during training for MuJuCo continuous control tasks, averaged across 10 runs with random initialization. The shaded area depicts the mean ± the standard deviation.

4.8 Conclusion

In this paper, we present two policy optimization frameworks, WPO and SPO, which can exactly characterize the policy updates instead of confining their distributions to particular distribution class or requiring any approximation. Our methods outperform TRPO and PPO with better sample efficiency, faster convergence, and improved final performance. Our numerical results show that the Wasserstein metric is more robust to the ambiguity of advantage functions, compared with the KL divergence. Our strategy for adjusting β value for WPO can reduce the computational time and boost the convergence without noticeable performance degradation. SPO improves the convergence speed of WPO by properly choosing the weight of the entropic regularizer. Performance improvement and global convergence for WPO are discussed. For future work, it remains interesting to extend the idea to PPO and natural policy gradients, which penalize the policy update instead of imposing trust region constraint, and extend it to off-policy frameworks.

Chapter 5

EXPERT-GUIDED WASSERSTEIN POLICY OPTIMIZATION FOR WHOLE-BUILDING HVAC CONTROL

In general, model-free RL methods face the challenge of sample inefficiency, requiring a substantial amount of data to refine their policies. The WPO framework, as a model-free RL approach introduced in Chapter 4, is no exception to this limitation. In this chapter, we propose an enhancement to address the sample inefficiency of WPO by integrating human guidance through the application of Generative Adversarial Imitation Learning (GAIL) [113]. This results in the creation of the Expert-Guided WPO (EGWPO), combining the strengths of both WPO and GAIL. By merging expert knowledge with reinforcement signals, EGWPO aims to augment the learning efficiency of the original WPO. Additionally, by leveraging the nonparametric policy representation inherent in WPO, EGWPO seeks to mitigate the sub-optimality issue associated with the original GAIL. To evaluate the effectiveness of our approach, we apply it to a challenging whole-building HVAC control problem. Acquiring samples from real building environments is resource-intensive, emphasizing the need for a highly sample-efficient approach. We derive a closed-form expression for optimal policy updates and develop an efficient on-policy actor-critic algorithm to perform these updates. Through experiments conducted on a 5-zone building model for HVAC control, we showcase the superior performance of EGWPO compared to state-of-the-art RL algorithms.

5.1 Introduction

Buildings account for a significant portion of global energy consumption, with the buildings sector accounting for approximately 36% of global energy usage and 37% of global CO2 emissions in 2021 [1]. The heating, ventilation, and air conditioning (HVAC) system, which regulates building's temperature, humidity, and air quality to maintain a comfortable and healthy indoor environment, becomes the largest contributor to energy usage of buildings,

accounting for approximately 50%. Therefore, it is essential to develop an energy-efficient HVAC control strategy that can effectively reduce energy consumption while still maintaining comfortable indoor conditions.

Optimizing HVAC control strategies is crucial for reducing energy consumption and achieving sustainable building design. Currently, many HVAC systems rely on feedback controllers like the proportional-integral-derivative (PID) [145] or traditional rule-based controller (RBC) [49, 213]. PID approaches adjust system operation based on the error between the desired set point and the measured system output, while RBC approaches rely on a predetermined set of rules and decision-making logic to govern system control. Although these methods are straightforward and easy to implement, they lack the ability to incorporate predictive information on future thermal dynamics, often resulting in suboptimal performance.

In recent years, model predictive control (MPC) has emerged as a more advanced model-based approach for HVAC control [51, 76, 205, 289]. By optimizing HVAC objectives over a prediction horizon, MPC can dynamically adjust the HVAC system to minimize energy consumption while maintaining indoor thermal comfort. However, the implementation of MPC-related methods in practice is not without its challenges. One of the significant challenges of MPC is the requirement for an accurate thermal dynamic model, which can be difficult to obtain due to uncertainties in the building's construction, occupancy, and usage patterns. Another challenge is the computational burden of solving a new optimization problem every time the system state changes, which can limit the real-time applicability of MPC [141].

To address the above issues, model-free deep reinforcement learning (RL) has emerged as a promising tool in HVAC control. Unlike traditional model-based approaches, RL does not require an explicit dynamics model. This makes it more flexible and adaptable, especially when an accurate dynamics model is not available. Additionally, RL supports online decision making, enabling an optimal control strategy to be generated instantly in any new system state. Furthermore, it is able to scale up to large spaces due to the control policy being represented by a neural network.

Deep Q-Network (DQN) [9, 70, 172, 255, 271] was among the first DRL methods employed

in HVAC control, but it is only compatible with discrete control actions. To extend its applicability to continuous action, many policy-based RL methods, including actor-critic [265, 301, 304, 305], Proximal Policy Optimization (PPO) [13, 302], Trust Region Policy Optimization (TRPO) [33] and Deep Deterministic Policy Gradient (DDPG) [71, 84, 172, 301], have been applied to HVAC control.

However, when applying model-free RL methods to HVAC control, several limitations can hinder their performance. One prominent limitation is the constraint imposed on the policy representation, wherein it is restricted to a specific distribution class such as Dirac delta [70, 71, 84, 255, 271], Gaussian [13, 33], and others. This constraint can often lead to the convergence to suboptimal policies, as the exploration process may exclude the optimal policy from consideration. The implications of suboptimal policies in HVAC control can be detrimental. Suboptimal policies can result in significant energy waste, temperature variations, discomfort, and even equipment failures, posing risks to the overall efficiency and safety of building systems. These issues are particularly problematic as HVAC control systems play a vital role in maintaining optimal indoor thermal conditions, ensuring occupant comfort, and minimizing energy consumption. Another limitation of these RL methods is their sample inefficiency, which hampers their practical applicability in HVAC control. Achieving a satisfactory policy with these methods often requires a substantial amount of training data, making the process time-consuming and resource-intensive. However, collecting real-world data for training and evaluating HVAC control algorithms is a challenging task. It involves significant costs, consumes time, and may not encompass a wide range of scenarios, limiting the generalizability of the learned policies.

To overcome these limitations, we adopt Wasserstein Policy Optimization (WPO) [235] approach. WPO is a policy optimization framework with a non-parametric policy representation that allows for a wider range of admissible policies compared to traditional model-free RL methods. By releasing the restrictive parametric policy assumption and considering all admissible policies within the trust region, WPO offers the potential to discover policies that are more energy-efficient and comfortable for occupants.

To further enhance the sample efficiency of WPO, we propose the incorporation of expert demonstrations. These demonstrations typically consist of a series of expert actions or control

sequences along with contextual information such as temperature, humidity, occupancy, and time of day. Expert demonstrations serve as a reference for desirable control behaviors under various operating conditions. Prior studies have demonstrated the effectiveness of expert demonstrations in expediting the learning and exploration process of HVAC control [50, 125]. Our proposed approach integrates expert demonstrations with reinforcement signals, utilizing the imitation learning technique Generative Adversarial Imitation Learning (GAIL) [113]. By leveraging the knowledge and experience of human experts, our approach has the potential to accelerate the learning process of WPO and enhance the overall effectiveness of HVAC control.

- Theory: We introduce Expert-Guided Wasserstein Policy Optimization (EGWPO), a framework that combines the imitation learning component GAIL with the non-parametric WPO policy optimization framework. In Section 5.3, we formulate the EGWPO optimization problem, propose an efficient policy update strategy, and design the corresponding practical RL algorithm. The new EGWPO algorithm is able to learn simultaneously from human demonstrations and environment feedback.

- Experiments: In Section 5.5, we conduct HVAC control experiments on a simulated building environment with EnergyPlus [57]. Our numerical study demonstrates that WPO outperforms both baseline RL algorithms and rule-based control strategies. Moreover, EGWPO further enhances the convergence and final performance of WPO. Additionally, we show that EGWPO maintains a more stable and comfortable indoor temperature compared to other methods.

5.2 Markov Decision Process (MDP) Model of HVAC Control

This section presents our approach to HVAC control as a Markov Decision Process (MDP) with an infinite horizon, defined by the tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the transition probability, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $\gamma \in (0, 1)$ is the discount factor. In our numerical studies, we use EnergyPlus [57] to simulate the whole building. In the following, we specify each key components of this MDP.

- **State:** For each simulation timestep t, the state $s_t \in \mathcal{S}$ comprises 19 continuous state variables (observations) from EnergyPlus. These variables include outdoor air temperature, outdoor air humidity, wind speed, solar radiation rate, among others.

- **Action:** For each simulation timestep t, the action $a_t \in \mathcal{A}$ consists of two continuous control action variables that dictate the heating and cooling setpoints. The heating setpoint's value range is set to $[15, 22.5]$, while the cooling setpoint's value range is set to $[22.5, 30]$.

- **Transition Probabilities:** The transition probabilities govern the behavior of the building environment. Unlike model-based approaches like MPC, model-free RL doesn't rely on a detailed understanding of the building dynamics. Instead, it learns by trial and error using feedback from the environment. The dynamics is specified by EnergyPlus, and is unknown to the RL agent.

- **Rewards:** Our reward function is a composite of two factors: the cost of electrical energy and the penalty for temperature violations. For each simulation timestep t, the reward r_t is defined as:

$$\begin{aligned} r_t = &- \lambda \cdot c_E \cdot \text{cost}(a_{t-1}, s_{t-1}) \\ &- (1-\lambda) \cdot c_T \cdot \sum_{i=1}^{N} ([T_t^i - \overline{T}^i]_+ + [\underline{T}^i - T_t^i]_+), \end{aligned}$$

  where $\text{cost}(a_{t-1}, s_{t-1})$ denotes the electrical energy cost of performing control action a_{t-1} in state s_{t-1}, T_t^i denotes the temperature in zone i at timestep t, $\underline{T}^i$ and $\overline{T}^i$ denote the lower and upper bounds of comfort temperature in zone i, λ denotes the weight of energy cost, c_E and c_T denote the scaling constant of energy cost and temperature-violation penalty respectively.

The policy $\pi(\cdot|s)$ is a probability distribution $\mathcal{S} \times \mathcal{A} \to [0, 1]$, which specifies the probability of taking an action a in a state s. We define the return of timestep t as the accumulated discounted reward from t, $R_t = \sum_{k=0}^{\infty} \gamma^k r(s_{t+k}, a_{t+k})$, and the value function as $V^\pi(s) = \mathbb{E}[R_t | s_t = s; \pi]$. The performance of a stochastic policy π is defined as

$J(\pi) = \mathbb{E}_{s_0,a_0,s_1...}[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]$ where $a_t \sim \pi(a_t|s_t)$, $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$. The objective of the HVAC control is to find an optimal policy π^* that minimizes the total accumulated cost, or, maximizes $J(\pi)$. As shown in [129], the expected return of a new policy π' can be expressed in terms of the advantage over the old policy π: $J(\pi') = J(\pi) + \mathbb{E}_{s\sim\rho_\upsilon^{\pi'}, a\sim\pi'}[A^\pi(s, a)]$, where $A^\pi(s, a) = \mathbb{E}[R_t|s_t = s, a_t = a; \pi] - \mathbb{E}[R_t|s_t = s; \pi]$ represents the advantage function and ρ_υ^π represents the unnormalized discounted visitation frequencies with initial state distribution υ, i.e., $\rho_\upsilon^\pi(s) = \mathbb{E}_{s_0\sim\upsilon}[\sum_{t=0}^{\infty} \gamma^t P(s_t = s|s_0)]$.

5.3 EGWPO: An Expert-Guided Wasserstein Policy Optimization Framework

In this section, we introduce the Expert-Guided Wasserstein Policy Optimization (EGWPO) framework, a novel approach that combines Generative Adversarial Imitation Learning (GAIL), an imitation learning component, with Wasserstein Policy Optimization (WPO), a nonparametric policy optimization reinforcement learning (RL) method. By integrating these components, the EGWPO framework enables simultaneous learning from both human demonstrations and environment feedback.

To provide a comprehensive understanding of the EGWPO framework, we first present an overview of the original WPO and GAIL frameworks, offering the necessary background information for the subsequent discussions.

5.3.1 Wasserstein Policy Optimization

Wasserstein Policy Optimization (WPO) [235] is a nonparametric RL method for constrained policy optimization. It tackles the suboptimality problem commonly encountered in traditional gradient-based policy optimization methods such as TRPO [219], PPO [222], DDPG [151] and actor-critic [168]. In WPO, the new policy π' is found in each iteration to maximize the expected improvement of $J(\pi') - J(\pi)$, or equivalently, the expected value of the advantage function. The optimization problem of WPO is formulated as follows:

$$\begin{aligned} &\max_{\pi'\in\mathcal{P}} \quad \mathbb{E}_{s\sim\rho_\upsilon^\pi, a\sim\pi'}[A^\pi(s, a)] \\ &\text{where} \quad \mathcal{P} = \{\pi'|\mathbb{E}_{s\sim\rho_\upsilon^\pi}[d_{\mathrm{W}}(\pi(\cdot|s), \pi'(\cdot|s))] \leq \delta\}, \end{aligned} \tag{5.1}$$

where d_W is the Wasserstein distance $d_{\mathrm{W}}(\pi, \pi') = \inf_{Q \in \Pi(\pi, \pi')} \int d(a, a') dQ(a, a')$. The infimum is taken over all joint distributions Q with marginals π and π', and $d(a, a')$ is defined as the distance between actions a and a'. The core concept behind WPO is to operate directly on the policy distribution space instead of restricting policies to a specific distribution class. This approach allows us to consider all admissible policies that fall within trust regions, thereby minimizing approximation errors. The use of the Wasserstein metric enables the incorporation of flexible user-defined costs between actions and takes into account the geometry of the metric space, allowing distributions to have distinct or non-overlapping supports.

5.3.2 Generative Adversarial Imitation Learning

Generative Adversarial Imitation Learning (GAIL) [113] is a model-free imitation learning (IL) [119] method that utilizes the concept of generative adversarial network (GAN) [93] to generate realistic outputs. GAN has been successful in generating a wide range of outputs, from images to voice outputs. In GAIL, the agent trains a generator to mimic the expert behavior and a discriminator to differentiate the generated policy from the expert trajectories. The generator is considered to have succeeded in producing an accurate representation of the expert's behavior when the discriminator can no longer distinguish between the generated data and the true expert trajectories. By accomplishing this, the agent can identify the optimal policy that matches the generator's output. The optimization problem of GAIL is shown below:

$$\min_{\theta'} \max_{\omega} \mathbb{E}_{s \sim \rho_v^{\pi}, a \sim \pi_{\theta'}} [\log D_\omega(s, a)] + \mathbb{E}_{\pi_E} [\log(1 - D_\omega(s, a))], \tag{5.2}$$

where D_ω is the discriminator network parameterized by ω, π'_θ represents the policy to be learned that is parametrized by θ, and π_E represents the expert policy. The inner maximization seeks to identify the optimal negative log loss of a binary classification problem, which involves distinguishing between state-action pairs of π'_θ and π_E. On the other hand, the outer minimization involves determining the best policy π' that can minimize the loss.

During training, the GAIL agent learns to perform the task solely from expert demonstrations without the ability to request additional information from the expert or receiving

any reinforcement signals from the environment. In addition, GAIL restricts the policy representation the policy π'_θ to a specific parametric distribution class. As noted in [248], optimizing over such distributions can lead to convergence to a sub-optimal solution due to the non-convex nature of these distributions in the distribution space.

5.3.3 Our Approach: Expert-Guided Wasserstein Policy Optimization Framework

We propose a novel framework called Expert-Guided Wasserstein Policy Optimization (EGWPO) that combines reinforcement learning and imitation learning. This framework allows us to leverage expert demonstrations, as observed in GAIL, while preserving the desired nonparametric properties of the WPO policy. By simultaneously learning from environmental and expert feedback, EGWPO improves upon both WPO and GAIL by harnessing the advantages of both approaches. The integration of expert knowledge in EGWPO enhances the learning effectiveness of the original WPO. Additionally, by utilizing the nonparametric policy representation from WPO, EGWPO mitigates the sub-optimality issue typically associated with the original GAIL algorithm. This innovative combination of reinforcement learning and imitation learning in EGWPO offers a powerful framework that achieves improved performance and addresses the limitations of the individual methods.

The original WPO objective in (5.1) solely relies on environmental rewards. To incorporate expert demonstrations, we extend it by integrating the objective of GAIL. We begin by revising the objective of GAIL to accommodate the use of a nonparametric policy π' instead of a parametric policy π'_θ:

$$\max_{\pi'} \min_{\omega} -\mathbb{E}_{s\sim\rho^{\pi}_{\upsilon},a\sim\pi'}[\log D_\omega(s,a)] - \mathbb{E}_{\pi_E}[\log(1-D_\omega(s,a))], \tag{5.3}$$

Next, we linearly combine the nonparametric GAIL objective in (5.3) with the original WPO objective to formulate the optimization problem for our proposed framework, EGWPO, as follows:

$$\begin{aligned} &\zeta \times (5.1) + (1-\zeta) \times (5.3) \\ &= \max_{\pi'\in\mathcal{P}} \min_{\omega} \mathbb{E}_{s\sim\rho^{\pi}_{\upsilon},a\sim\pi'}[\zeta A^{\pi}(s,a) - (1-\zeta)\log D_\omega(s,a)] \\ &\quad - (1-\zeta)\mathbb{E}_{\pi_E}[\log(1-D_\omega(s,a))]. \end{aligned} \tag{5.4}$$

In the optimization problem of EGWPO, we take into consideration both environmental and expert feedbacks, with ζ serving as a parameter that determines the balance between learning from the environment and human demonstrations. Notably, ζ can be time-dependent, allowing for adaptive weighting of WPO and IL during the learning process. When constructing the trust region (i.e., the ambiguity set for policy) for EGWPO, we utilize the Wasserstein metric, denoted as:

$$\mathcal{P} = \{\pi' | \mathbb{E}_{s\sim\rho_v^\pi}[d_{\mathrm{W}}(\pi(\cdot|s), \pi'(\cdot|s))] \leq \delta\}$$

We choose the Wasserstein metric over the Kullback-Leibler divergence because it accounts for the geometry of the metric space and allows distributions to have different or non-overlapping supports, which is particularly advantageous in our case where the action support changes at each iteration due to the sampled action space in (5.7). This makes the Wasserstein metric more suitable for effectively handling the dynamics of our sampled action space in the optimization process.

We derive the optimal closed-form policy update for the EGWPO problem in (5.4), as presented in Theorem 10:

Theorem 10. ***(Closed-form policy update)*** *Let* $k_s^\pi(\beta, a) = argmax_{a_k\in\mathcal{A}} \zeta A^\pi(s, a_k) - (1-\zeta)\log D_\omega(s, a_k) - \beta d(a, a_k)$. *Assume that* $A^\pi(s, a)$ *and* $D_\omega(s, a)$ *are bounded, then an optimal solution to the EGWPO problem* (5.4) *is:*

$$\pi^*(a'|s) = \int_{a\in\mathcal{A}} \pi(a|s) f_s^*(a, a') da, \tag{5.5}$$

where $f_s^*(a, a') = 1$ *if* $a' = k_s^\pi(\beta^*, a)$ *and* $f_s^*(a, a') = 0$ *otherwise, and* β^* *is an optimal Lagrangian multiplier corresponding to the following dual formulation:*

$$\begin{aligned}\min_{\beta\geq 0}\{\beta\delta + \mathbb{E}_{s\sim\rho_v^\pi} \int_{a\in\mathcal{A}} \pi(a|s) \max_{a'\in\mathcal{A}} (\zeta A^\pi(s, a') \\ - (1-\zeta)\log D_\omega(s, a') - \beta d(a, a'))\}.\end{aligned} \tag{5.6}$$

The proof is provided in Appendix D.1. This concise form of policy update facilitates seamless integration with the practical policy optimization algorithm discussed in Section 5.4, making it straightforward to implement in practice.

5.4 A Practical Algorithm

In this section, we introduce methods aimed at improving the efficiency of EGWPO policy updates. These techniques encompass the integration of a time-dependent parameter β and the utilization of support space sampling. Additionally, considering that advantage value functions are commonly estimated from sampled trajectories in practical scenarios, we propose a practical on-policy actor-critic algorithm, outlined in Algorithm 5. This algorithm combines EGWPO policy updates with the estimation of advantage functions.

5.4.1 Efficient EGWPO Policy Update

To enhance computational efficiency, we avoid solving the dual formulation in (5.6) to determine β. Instead, we introduce a time-dependent sequence β_k, where β_k serves as a time-varying penalty on the trust region constraint in (5.1). As the policy is refined through learning, the violation of the trust region constraint diminishes. Therefore, β_k can be chosen as a decreasing sequence, such as $\frac{1}{k}$. This approach substantially reduces computational costs while yielding a solution to the penalty-based version of the EGWPO problem:

$$\begin{aligned}
&\max_{\pi_{k+1}} \min_{\omega_k} \\
&\mathbb{E}_{s\sim\rho_v^{\pi_k}, a\sim\pi_{k+1}}[\zeta_k A^{\pi_k}(s,a) - (1-\zeta_k)\log D_{\omega_k}(s,a)] \\
&\qquad - (1-\zeta_k)\mathbb{E}_{\pi_E}[\log(1 - D_{\omega_k}(s,a))] \\
&\qquad - \beta_k \mathbb{E}_{s\sim\rho_v^{\pi_k}}[d_{\mathrm{W}}(\pi_k(\cdot|s), \pi_{k+1}(\cdot|s))].
\end{aligned}$$

Rather than integrating over the entire action space $\mathcal{A}$ for the policy update in (5.5), we opt to sample a subset of $\mathcal{A}$. Specifically, we define the action support for state s in the k-th iteration as:

$$\begin{aligned}
I_k(s) = \{a' : &\zeta_k A^{\pi_k}(s,a') - (1-\zeta_k)\log D_{\omega_k}(s,a') \\
> &\min_{a\in I_{k-1}(s)} \zeta_k A^{\pi_k}(s,a) - (1-\zeta_k)\log D_{\omega_k}(s,a)\}.
\end{aligned} \tag{5.7}$$

The policy update for efficient EGWPO in the k-th iteration is then constructed based on the sampled action support $I_k(s)$ as follows:

$$\pi_{k+1}(a'|s) = \mathbb{F}(\pi_k) = \sum\nolimits_{a\in I_{k-1}(s)} \pi_k(a|s) f_s^k(a,a') da, \tag{5.8}$$

where $f_s^k(a, a') = 1$ if $a' = \text{argmax}_{a' \in I_k(s)}\{\zeta_k A^{\pi_k}(s, a') - (1 - \zeta_k) \log D_{\omega_k}(s, a') - \beta_k d(a, a')\}$ and $f_s^k(a, a') = 0$ otherwise.

During implementation, we sample $|\mathcal{A}_k|$ actions to approximate the support $I_k(s)$ as well as the target policy distribution $\pi_{k+1}(\cdot|s)$. Additionally, we sample a batch of states $\mathcal{S}_k \subseteq \mathcal{S}$ at each iteration to perform policy updates. This combination of action and state sampling enables efficient computation of policy updates in the EGWPO algorithm.

5.4.2 Practical EGWPO Algorithm

Based on the efficient EGWPO policy update shown in (5.8), we propose the following practical EGWPO algorithm:

Algorithm 5: On-policy Expert-Guided Wasserstein Policy Optimization algorithm (EGWPO)

Input: number of iterations K, learning rate α

Initialize policy π_0 and value network V_{ψ_0} with random parameter ψ_0, discriminator network D_{ω_0} with random parameter ω_0

for $k = 0, 1, 2 \ldots K$ **do**

Collect trajectory set $\mathcal{D}_k$ on policy π_k

For each timestep t in each trajectory, compute total returns G_t and estimate advantages $\hat{A}_t^{\pi_k}$

Update value:

$\psi_{k+1} \leftarrow \psi_k - \alpha \nabla_{\psi_k} \sum (G_t - V_{\psi_k}(s_t))^2$

Update discriminator network:

$\omega_{k+1} \leftarrow \omega_k + \alpha \nabla_{\omega_k} \{\mathbb{E}_{\pi_k}[\log D_{\omega_k}(s, a)] + \mathbb{E}_{\pi_E}[\log(1 - D_{\omega_k}(s, a))]\}$

Update policy:

$\pi_{k+1} \leftarrow \mathbb{F}(\pi_k)$ via EGWPO (5.8)

end

In this practical algorithm, we follow the standard on-policy actor-critic framework, where both the value network and the policy network are updated simultaneously in each iteration

using freshly generated trajectories. In addition to the value and policy networks, we also train a discriminator network that aids in the imitation learning process, leveraging human feedback. The discriminator network parameter ω is updated through gradient ascent, with the gradient computed from the EGWPO objective, given by $(1-\zeta)\{\mathbb{E}_{s\sim\rho_v^{\pi},a\sim\pi'}[\log D_\omega(s,a)]+\mathbb{E}_{\pi_E}[\log(1-D_\omega(s,a))]\}$.

The trajectories sampled in the algorithm can be either complete or partial. For complete trajectories, G_t can be obtained using the accumulated discounted rewards, i.e., $R_t = \sum_{k=0}^{T-t}\lambda^k r_{t+k}$. For partial trajectories, G_t can be estimated using multi-step temporal difference (TD) methods [61], as $\hat{R}_{t:t+n} = \sum_{k=0}^{n-1}\lambda^k r_{t+k} + \lambda^n V(s_{t+n})$, where n is the number of steps. To estimate the advantage $\hat{A}_t^{\pi_k}$, we can use either the Monte Carlo approach, i.e., $\hat{A}_t^{\pi_k} = G_t - V_{\psi_k}(s_t)$, or the Generalized Advantage Estimation (GAE) method [220].

5.5 Numerical Studies

In this section, we focus on the HVAC control of a classic 5-zone building model and evaluate the effectiveness of our proposed EGWPO algorithm as outlined in Algorithm 5. We compare the performance of EGWPO with several other methods:

1. Rule-based method: This approach employs fixed cooling and heating setpoints based on comfort temperature bounds.

2. Baseline RL algorithms: We include Trust Region Policy Optimization (TRPO), Proximal Policy Optimization (PPO), and Advantage Actor Critic (A2C) [168]. A2C is an actor-critic RL algorithm based on the advantage function.

3. Individual components of EGWPO: We also evaluate the performance of Wasserstein Policy Optimization (WPO) and Generative Adversarial Imitation Learning (GAIL) separately to assess if the integrated EGWPO algorithm surpasses their individual performance.

5.5.1 Environment Setup

In our experiment, we utilized the widely used 5-zone building model from EnergyPlus, as depicted in Figure 5.1, to simulate a single-floor rectangular building measuring 463.6m^2 (30.5m $\times$ 15.2m), which includes 4 exterior zones and 1 interior zone. To enable interactions between EnergyPlus and deep reinforcement learning algorithms, including WPO and baseline RL methods, we employed Sinergym [127], which allows EnergyPlus to interface with OpenAI Gym [39] compatible environments. Figure 5.2 presents a visual representation of how Sinergym connects EnergyPlus with deep reinforcement learning [127]. All experiments were performed on a 2.7 GHz Quad-Core Intel Core i7 processor.

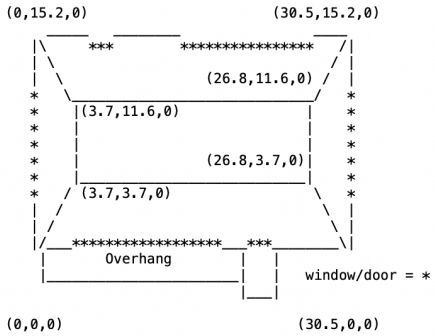


Figure 5.1: 5-zone building

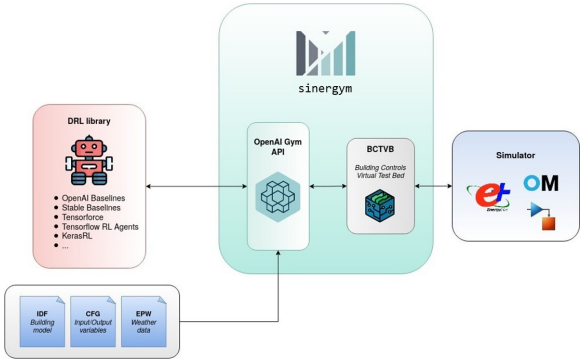


Figure 5.2: Sinergym interface

5.5.2 Performance Comparison

We performed a thorough assessment of the performance of Expert-Guided Wasserstein Policy Optimization (EGWPO) in the field of HVAC control. To generate expert demonstrations, Proximal Policy Optimization (PPO) agents were trained over 500 episodes. The comparative performance results are presented in Figure 5.3. The figure clearly demonstrates that, with the exception of A2C, all RL algorithms outperform the rule-based method. Notably, WPO exhibits rapid convergence and achieves a superior optimal solution compared to the baseline RL methods. These findings strongly suggest that employing a nonparametric WPO policy representation can expedite convergence and enhance optimality in the domain of HVAC control.

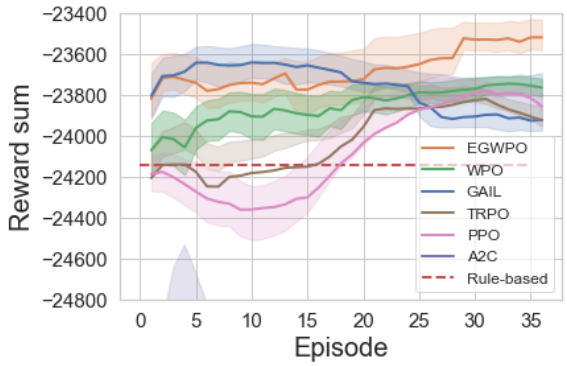
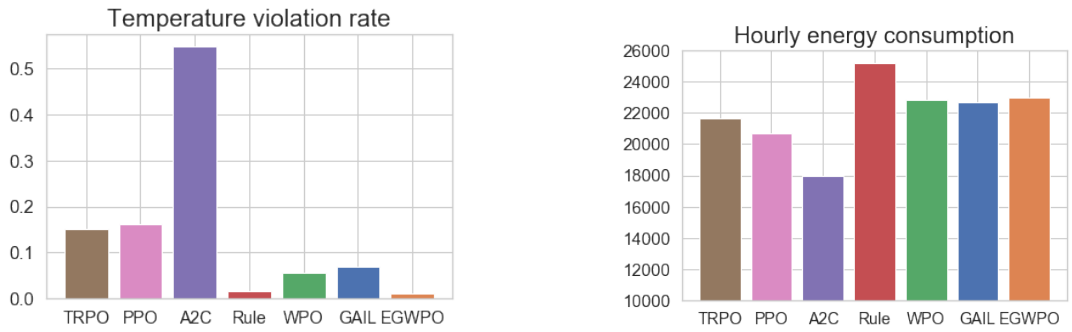


Figure 5.3: Episode rewards during the training process, averaged across 5 runs with a random initialization. The shaded area depicts the mean ± the standard deviation.

We conducted a detailed performance analysis of EGWPO, WPO and GAIL. The results, also illustrated in Figure 5.3, demonstrate that EGWPO and GAIL initially outperform WPO. However, towards the end of the training process, GAIL's performance deteriorates due to overfitting. Importantly, the integrated algorithm, EGWPO, achieves better overall performance than its individual components, WPO and GAIL. Moreover, we observed that EGWPO reached the same reward sum approximately 20 episodes (equivalent to 7×10^5 samples) earlier than WPO, indicating its greater sample efficiency in achieving desirable performance.



(a) Temperature violation rate

(b) Hourly energy consumption

Figure 5.4: Statistics on temperature violation rate and hourly energy consumption

To gain deeper insights into the performance of each algorithm, we conducted a comprehensive analysis of temperature violation rate and hourly energy consumption, as depicted in Figure 5.4a and 5.4b, respectively. Additionally, we present detailed temperature trends in Figure 5.5 and 5.6 to provide a more comprehensive evaluation.

Upon analyzing the results presented in Figure 5.4a and 5.4b, a trade-off between temperature violation and energy consumption becomes evident. It is observed that higher energy usage is required to maintain a stable and comfortable indoor temperature. Among the tested algorithms, A2C stands out with the lowest energy consumption, approximately $18,000Wh$ per hour. However, it also exhibits the highest temperature violation rate, exceeding 50%. Conversely, the rule-based control strategy shows the most stable temperature trends but consumes a significant amount of energy, around $25,000Wh$ per hour. WPO, GAIL, and EGWPO achieve lower energy consumption, ranging from $22,000$ to $23,000Wh$ per hour, representing an approximate 10% reduction compared to the rule-based strategy. Notably, EGWPO outperforms the other algorithms, maintaining a comfortable and stable temperature with the lowest temperature violation rate of 1.048%, even surpassing the rule-based control strategy in this aspect.

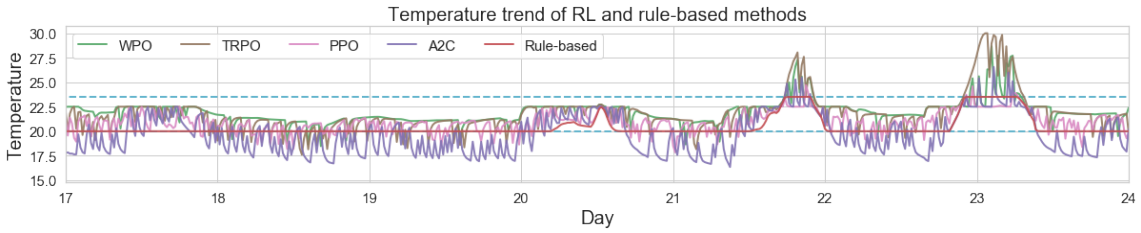


Figure 5.5: Temperature trend of RL and rule baselines

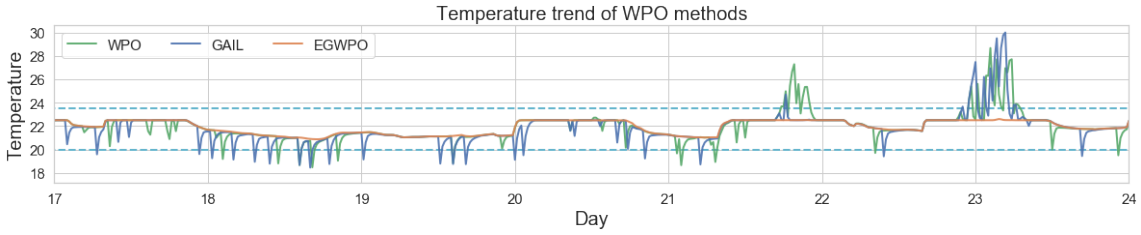


Figure 5.6: Temperature trend of WPO, GAIL and EGWPO

5.6 Conclusion

In this paper, we introduce EGWPO, a novel Expert-Guided Wasserstein Policy Optimization framework that combines the advantages of GAIL's expert demonstrations with the nonparametric properties of WPO policies. Additionally, we propose a practical and computationally efficient algorithm by incorporating a time-dependent parameter β and leveraging support sampling. To evaluate the effectiveness of our approach, we conduct experiments on HVAC control using a classic 5-zone building model.

The experimental results demonstrate that our proposed EGWPO algorithm outperforms baseline RL algorithms, GAIL and WPO, achieving superior final performance in terms of HVAC control. Moreover, EGWPO exhibits enhanced sample efficiency compared to the original WPO, requiring fewer samples to achieve desirable performance levels. These findings highlight the efficacy of our Expert-Guided Wasserstein Policy Optimization framework, showcasing its potential for improving the performance and sample efficiency of HVAC control systems.

Chapter 6

FUTURE WORK

Our future research directions can be categorized into two main areas: the application of KLPO, WPO, and SPO to large-scale systems, and the extension of EGWPO for HVAC control in real-world smart homes. These directions are detailed below:

The innovative deep RL methodologies introduced in our previous research, namely KLPO (Chapter 3), WPO, and SPO (Chapter 4), have shown remarkable robustness and superior performance compared to the traditional RL. These methodologies offer substantial potential for application across diverse domains, especially in addressing challenges characterized by large-scale complexities. Looking ahead, we anticipate employing the WPO approach to address various challenges within large-scale energy systems. For example, it shows great promise for enhancing energy efficiency in large-scale building energy management systems (BEMS). Additionally, it can efficiently determine optimal dynamic energy dispatch strategies within large-scale integrated energy systems (IES). Our future research will focus on these applications, aiming to comprehensively demonstrate the effectiveness of our proposed RL approach.

The effectiveness of the EGWPO, as proposed in Chapter 5, has been successfully demonstrated in its application to the simulated EnergyPlus building system [127]. This involved guidance from a simulated PPO expert [223]. Looking ahead, our research endeavors will extend to deploying EGWPO for HVAC control in real-world smart homes located in Texas, where high summer temperatures present unique challenges. In our upcoming implementations, we will not only rely on simulated experts but also integrate guidance from actual human operators. This multi-faceted approach ensures the robust and adaptive application of EGWPO, enhancing its performance and applicability in real-world scenarios.

BIBLIOGRAPHY

[1] 2022 global status report for buildings and construction. https://www.unep.org/resources/publication/2022-global-status-report-buildings-and-construction.

[2] FERC Order No. 2222: A new day for distributed energy resources. *Federal Energy Regulatory Commission, Tech. Rep*, 2021.

[3] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning*, page 1, 2004.

[4] Abbas Abdolmaleki, Jost Tobias Springenberg, Jonas Degrave, Steven Bohez, Yuval Tassa, Dan Belov, Nicolas Heess, and Martin Riedmiller. Relative entropy regularized policy iteration. *ArXiv Preprint*, page arXiv:1812.02256, 2018.

[5] Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. *ArXiv Preprint*, page arXiv:1806.06920, 2018.

[6] Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *ArXiv Preprint*, page arXiv:1908.00261, 2019.

[7] Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *J. Mach. Learn. Res.*, 22(98):1–76, 2021.

[8] Rishabh Agarwal, Marlos C Machado, Pablo Samuel Castro, and Marc G Bellemare. Contrastive behavioral similarity embeddings for generalization in reinforcement learning. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[9] Aya A. Amer, Khaled Shaban, and Ahmed M. Massoud. DRL-HEMS: Deep reinforcement learning agent for demand response in home energy management systems considering customers and operators perspectives. *IEEE Transactions on Smart Grid*, 14(1):239–250, 2023.

[10] Pavlos Athanasios Apostolopoulos, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. Demand response management in smart grid networks: A two-stage game-theoretic learning-based approach. *Mobile Networks and Applications*, pages 1–14, 2018.

[11] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *ArXiv Preprint*, page arXiv:1701.07875, 2017.

[12] Dilip Arumugam, Jun Ki Lee, Sophie Saskin, and Michael L Littman. Deep reinforcement learning from policy-dependent human feedback. *ArXiv Preprint*, page arXiv:1902.04257, 2019.

[13] Donald Azuatalam, Wee-Lih Lee, Frits de Nijs, and Ariel Liebman. Reinforcement learning for whole-building hvac control and demand response. *Energy and AI*, 2:100020, 2020.

[14] Sadra Babaei, Ruiwei Jiang, and Chaoyue Zhao. Distributionally robust distribution network configuration under random contingency. *IEEE Transactions on Power Systems*, 35(5):3332–3341, 2020.

[15] Sadra Babaei, Chaoyue Zhao, and Lei Fan. A data-driven model of virtual power plants in day-ahead unit commitment. *IEEE Transactions on Power Systems*, 34(6):5125–5135, 2019.

[16] Ali Bagheri, Jianhui Wang, and Chaoyue Zhao. Data-driven stochastic transmission expansion planning. *IEEE Transactions on Power Systems*, 32(5):3461–3470, 2016.

[17] Ali Bagheri and Chaoyue Zhao. Distributionally robust reliability assessment for transmission system hardening plan under *nk* security criterion. *IEEE Transactions on Reliability*, 68(2):653–662, 2019.

[18] Ali Bagheri, Chaoyue Zhao, Feng Qiu, and Jianhui Wang. Resilient transmission hardening planning in a high renewable penetration era. *IEEE Transactions on Power Systems*, 34(2):873–882, 2018.

[19] Shahab Bahrami, Vincent WS Wong, and Jianwei Huang. An online learning algorithm for demand response in smart grid. *IEEE Transactions on Smart Grid*, 9(5):4712–4725, 2017.

[20] David Barber and Christopher M Bishop. Ensemble learning in bayesian neural networks. *NATO ASI Series. Series F: Computer and Systems Sciences*, 168:215–238, 1998.

[21] Gabriel Barth-Maron, Matthew W Hoffman, David Budden, Will Dabney, Dan Horgan, Dhruva Tb, Alistair Muldal, Nicolas Heess, and Timothy Lillicrap. Distributed distributional deterministic policy gradients. *ArXiv Preprint*, page arXiv:1804.08617, 2018.

[22] Andrew G. Barto, Steven J. Bradtke, and Satinder P. Singh. Learning to act using real-time dynamic programming. *Artificial Intelligence*, 72(1):81 – 138, 1995.

[23] Andrew G. Barto, Richard S. Sutton, and Charles W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5):834–846, 1983.

[24] Beste Basciftci, Shabbir Ahmed, and Siqian Shen. Distributionally robust facility location problem under decision-dependent stochastic demand. *Arxiv Preprint ArXiv:1912.05577*, 2020.

[25] Richard Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[26] Aharon Ben-Tal, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

[27] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust Optimization*. Princeton University Press, 2009.

[28] Aharon Ben-Tal and Arkadi Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.

[29] Jalaj Bhandari and Daniel Russo. On the linear convergence of policy gradient methods for finite mdps. In *International Conference on Artificial Intelligence and Statistics*, pages 2386–2394. PMLR, 2021.

[30] Ankit Bhardwaj, Han Ching Ou, Haipeng Chen, Shahin Jabbari, Milind Tambe, Rahul Panicker, and Alpan Raval. Robust lock-down optimization for COVID-19 policy guidance. In *AAAI Fall Symposium*, 2020.

[31] Jinbo Bi and Tong Zhang. Support vector classification with input data uncertainty. In *Advances in Neural Information Processing Systems*, volume 17, pages 161–168, 2004.

[32] Jinbo Bi and Tong Zhang. Support vector classification with input data uncertainty. In *Advances in Neural Information Processing Systems*, pages 161–168, 2005.

[33] Marco Biemann, Fabian Scheller, Xiufeng Liu, and Lizhen Huang. Experimental evaluation of model-free reinforcement learning algorithms for continuous hvac control. *Applied Energy*, 298:117164, 2021.

[34] John R. Birge and François Louveaux. *Introduction to Stochastic Programming*. Springer, 2011.

[35] Christopher M Bishop et al. *Neural networks for pattern recognition*. Oxford university press, 1995.

[36] Jose Blanchet and Karthyek Murthy. Quantifying distributional model risk via optimal transport. *Mathematics of Operations Research*, 44(2):565–600, 2019.

[37] Bruce Blumberg, Marc Downie, Yuri Ivanov, Matt Berlin, Michael Patrick Johnson, and Bill Tomlinson. Integrated learning for interactive synthetic characters. In *Annual Conference on Computer Graphics and Interactive Techniques*, pages 417–426, 2002.

[38] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pages 1613–1622, 2015.

[39] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[40] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schnpageser, John Schulman, Jie Tang, and Wojciech Zaremba. OpenAI gym. *ArXiv Preprint*, page arXiv:1606.01540, 2016.

[41] Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *ArXiv Preprint*, page arXiv:1204.5721, 2012.

[42] Russel E. Caflisch. Monte Carlo and quasi-Monte Carlo methods. *Acta Numerica*, 7:1–49, 1998.

[43] Pablo Samuel Castro. Scalable methods for computing state similarity in deterministic Markov decision processes. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, volume 34, pages 10069–10076, 2020.

[44] Thomas Cederborg, Ishaan Grover, Charles L Isbell Jr, and Andrea Lockerd Thomaz. Policy shaping with human teachers. In *International Joint Conferences on Artificial Intelligence*, pages 3366–3372, 2015.

[45] Shicong Cen, Chen Cheng, Yuxin Chen, Yuting Wei, and Yuejie Chi. Fast global convergence of natural policy gradient methods with entropy regularization. *ArXiv Preprint*, page arXiv:2007.06558, 2020.

[46] Shicong Cen, Chen Cheng, Yuxin Chen, Yuting Wei, and Yuejie Chi. Fast global convergence of natural policy gradient methods with entropy regularization. *Operations Research*, 2021.

[47] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

[48] Minwoo Chae and Stephen Walker. Wasserstein upper bounds of the total variation for smooth densities. *Statistics & Probability Letters*, 163:108771, 2020.

[49] Subhadip Chakraborty, Gaurav Modi, and Bhim Singh. A cost optimized-reliable-resilient-realtime-rule based energy management scheme for a SPV-BES based microgrid for smart building applications. *IEEE Transactions on Smart Grid*, pages 1–1, 2022.

[50] Bingqing Chen, Zicheng Cai, and Mario Bergés. Gnu-RL: A precocial reinforcement learning solution for building HVAC control using a differentiable MPC policy. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, BuildSys '19, page 316–325, 2019.

[51] Chen Chen, Jianhui Wang, Yeonsook Heo, and Shalinee Kishore. MPC-based appliance scheduling for residential building energy management controller. *IEEE Transactions on Smart Grid*, 4(3):1401–1410, 2013.

[52] Chunyu Chen, Mingjian Cui, Fangxing Fran Li, Shengfei Yin, and Xinan Wang. Model-free emergency frequency control based on reinforcement learning. *IEEE Transactions on Industrial Informatics*, 2020.

[53] Zhi Chen, Pengqian Yu, and William B. Haskell. Distributionally robust optimization for sequential decision-making. *Optimization*, 68(12):2397–2426, 2019.

[54] Po-Wei Chou, Daniel Maturana, and Sebastian Scherer. Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution. In *Proceedings of the 34th International Conference on Machine Learning*, pages 834–843, 2017.

[55] Nicolas Courty, Rémi Flamary, and Devis Tuia. Domain adaptation with regularized optimal transport. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 274–289. Springer, 2014.

[56] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1853–1865, 2016.

[57] Drury Crawley, Curtis Pedersen, Linda Lawrie, and Frederick Winkelmann. Energyplus: Energy simulation program. *Ashrae Journal*, 42:49–56, 2000.

[58] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, pages 2292–2300, 2013.

[59] Robert Dadashi, Léonard Hussenot, Matthieu Geist, and Olivier Pietquin. Primal Wasserstein imitation learning. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[60] Scott Davies. Multidimensional triangulation and interpolation for reinforcement learning. In *Advances in Neural Information Processing Systems*, page 1005–1011, 1996.

[61] Kristopher De Asis, J Hernandez-Garcia, G Holland, and Richard Sutton. Multi-step reinforcement learning: A unifying algorithm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, page 2902–2909, 2018.

[62] Kristopher De Asis, J. Fernando Hernandez-Garcia, G. Zacharias Holland, and Richard S. Sutton. Multi-step reinforcement learning: A unifying algorithm. *ArXiv Preprint*, page arXiv:1703.01327, 2017.

[63] Richard Dearden, Nir Friedman, and Stuart Russell. Bayesian Q-learning. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence and Tenth Innovative Applications of Artificial Intelligence Conference*, pages 761–768, 1998.

[64] Thomas Degris, Martha White, and Richard S Sutton. Off-policy actor-critic. *ArXiv Preprint*, page arXiv:1205.4839, 2012.

[65] Erick Delage and Yinyu Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612, 2010.

[66] John S Denker and Yann LeCun. Transforming neural-net output levels to probability distributions. In *International Conference on Neural Information Processing Systems*, pages 853–859, 1990.

[67] Prafulla Dhariwal, Christopher Hesse, Oleg Klimov, Alex Nichol, Matthias Plappert, Alec Radford, John Schulman, Szymon Sidor, Yuhuai Wu, and Peter Zhokhov. OpenAI baselines. `https://github.com/openai/baselines`, 2017.

[68] Thomas G. Dietterich. The MAXQ method for hierarchical reinforcement learning. In *Proceedings of the 15th International Conference on Machine Learning*, pages 118–126, 1998.

[69] Tao Ding, Cheng Li, Chaoyue Zhao, and Min Wang. Total supply capability considering distribution network reconfiguration under n- k transformer contingency and the decomposition method. *IET Generation, Transmission & Distribution*, 11(5):1212–1222, 2016.

[70] Xianzhong Ding, Wan Du, and Alberto Cerpa. Octopus: Deep reinforcement learning for holistic smart building dontrol. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, BuildSys '19, page 326–335, 2019.

[71] Yan Du, Helia Zandi, Olivera Kotevska, Kuldeep Kurte, Jeffery Munk, Kadir Amasyali, Evan Mckee, and Fangxing Li. Intelligent multi-zone residential hvac control strategy based on deep reinforcement learning. *Applied Energy*, 281:116117, 2021.

[72] Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 1329–1338, 2016.

[73] Gabriel Dulac-Arnold, Richard Evans, Hado van Hasselt, Peter Sunehag, Timothy Lillicrap, Jonathan Hunt, Timothy Mann, Theophane Weber, Thomas Degris, and Ben Coppin. Deep reinforcement learning in large discrete action spaces. *ArXiv Preprint*, page arXiv:1512.07679, 2015.

[74] Gabriel Dulac-Arnold, Daniel Mankowitz, and Todd Hester. Challenges of real-world reinforcement learning. *ArXiv Preprint*, page arXiv:1904.12901, 2019.

[75] Laurent El Ghaoui and Hervé Lebret. Robust solutions to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications*, 18(4):1035–1064, 1997.

[76] Jens Engel, Thomas Schmitt, Tobias Rodemann, and Jürgen Adamy. Hierarchical economic model predictive control approach for a building energy management system With scenario-driven EV charging. *IEEE Transactions on Smart Grid*, 13(4):3082–3093, 2022.

[77] Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1–2):115–166, 2018.

[78] Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166, 2018.

[79] Lei Fan, Chaoyue Zhao, Guangyuan Zhang, and Qiuhua Huang. Flexibility management in economic dispatch with dynamic automatic generation control. *ArXiv Preprint*, page arXiv:2006.03890, 2020.

[80] Nuno Fernandes. Economic effects of coronavirus outbreak (COVID-19) on the world economy. *Available at SSRN*, 2020.

[81] Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite Markov decision processes. In *Uncertainty in Artificial Intelligence*, volume 4, pages 162–169, 2004.

[82] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1582–1591, 2018.

[83] Holly Gaff and Elsa Schaefer. Optimal control applied to vaccination and treatment strategies for various epidemiological models. *Mathematical Biosciences & Engineering*, 6(3):469–492, 2009.

[84] Guanyu Gao, Jie Li, and Yonggang Wen. Deepcomfort: Energy-efficient thermal comfort control in buildings via reinforcement learning. *IEEE Internet of Things Journal*, 7(9):8472–8484, 2020.

[85] Rui Gao and Anton J Kleywegt. Distributionally robust stochastic optimization with Wasserstein distance. *arXiv preprint arXiv:1604.02199*, 2016.

[86] Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *International Conference on Machine Learning*, pages 2160–2169. PMLR, 2019.

[87] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with Sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pages 1608–1617, 2018.

[88] Alborz Geramifard, Christoph Dann, Robert H. Klein, William Dabney, and Jonathan P. How. RLPy: A value-function-based reinforcement learning framework for education and research. *Journal of Machine Learning Research*, 16(46):1573–1578, 2015.

[89] Alison L. Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International Statistical Review*, 70(3):419–435, 2002.

[90] Giulia Giordano, Franco Blanchini, Raffaele Bruno, Patrizio Colaneri, Alessandro Filippo, Angela Matteo, and Marta Colaneri. Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy. *Nature Medicine*, 26:855–860, 2020.

[91] Richard Glick, James Danly, Allison Clements, Mark C. Christie, and Willie L. Phillips. 2021 assessment of demand response and advanced metering. *Federal Energy Regulatory Commission, Tech. Rep*, 2021.

[92] Joel Goh and Melvyn Sim. Distributionally robust optimization and its tractable approximations. *Operations Research*, 58(4-part-1):902–917, 2010.

[93] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[94] Alex Graves. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*, pages 2348–2356, 2011.

[95] Shane Griffith, Kaushik Subramanian, Jonathan Scholz, Charles L Isbell, and Andrea L Thomaz. Policy shaping: Integrating human feedback with reinforcement learning. In *Advances in Neural Information Processing Systems*, page 2625–2633, 2013.

[96] Veronika Grimm, Friederike Mengel, and Martin Schmidt. Extensions of the SEIR model for the analysis of tailored social distancing and tracing approaches to cope with COVID-19. *Scientific Reports*, 11(4214), 2021.

[97] Gregory Z. Grudic, Vijay Kumar, and Lyle H. Ungar. Using policy gradient reinforcement learning on autonomous robot controllers. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 406–411, 2003.

[98] Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *Proceedings of the 2017 IEEE International Conference on Robotics and Automation*, pages 3389–3396, 2017.

[99] Shixiang Gu, Timothy Lillicrap, Zoubin Ghahramani, Richard E Turner, and Sergey Levine. Q-prop: Sample-efficient policy gradient with an off-policy critic. *ArXiv Preprint*, page arXiv:1611.02247, 2016.

[100] Akshay Gupte, Shabbir Ahmed, Myun Seok Cheon, and Santanu Dey. Solving mixed integer bilinear problems using MILP formulations. *SIAM Journal on Optimization*, 23(2):721–744, 2013.

[101] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In

Proceedings of the 35th International Conference on Machine Learning, pages 1861–1870, 2018.

[102] Chuanjia Han, Bo Yang, Tao Bao, Tao Yu, and Xiaoshun Zhang. Bacteria foraging reinforcement learning for risk-based economic dispatch via knowledge transfer. *Energies*, 10(5):638, 2017.

[103] Grani A. Hanasusanto, Vladimir Roitch, Daniel Kuhn, and Wolfram Wiesemann. Ambiguous joint chance constraints under mean and dispersion information. *Operations Research*, 65(3):751–767, 2017.

[104] Hado van Hasselt. Double Q-learning. In *Advances in Neural Information Processing Systems*, pages 2613–2621, 2010.

[105] Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double Q-learning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2094–2100. AAAI Press, 2016.

[106] Shuncheng He, Yuhang Jiang, Hongchang Zhang, Jianzhun Shao, and Xiangyang Ji. Wasserstein Unsupervised Reinforcement Learning. In *Proceedings of the Thirty-Sixth AAAI Conference on Artificial Intelligence*. AAAI Press, 2022.

[107] Nicolas Heess, Gregory Wayne, David Silver, Timothy Lillicrap, Tom Erez, and Yuval Tassa. Learning continuous control policies by stochastic value gradients. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[108] Johannes Heinrich and David Silver. Deep reinforcement learning from self-play in imperfect-information games. *ArXiv Preprint*, page arXiv:1603.01121, 2016.

[109] José Miguel Hernández-Lobato and Ryan Adams. Probabilistic backpropagation for scalable learning of bayesian neural networks. In *International Conference on Machine Learning*, pages 1861–1869, 2015.

[110] Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving

linear systems. *Journal of Research of the National Bureau of Standards*, 49:409–436, 1952.

[111] Ashley Hill, Antonin Raffin, Maximilian Ernestus, Adam Gleave, Anssi Kanervisto, Rene Traore, Prafulla Dhariwal, Christopher Hesse, Oleg Klimov, Alex Nichol, Matthias Plappert, Alec Radford, John Schulman, Szymon Sidor, and Yuhuai Wu. Stable baselines. `https://github.com/hill-a/stable-baselines`, 2018.

[112] Geoffrey E Hinton and Drew Van Camp. Keeping the neural networks simple by minimizing the description length of the weights. In *Annual Conference on Computational Learning Theory*, pages 5–13, 1993.

[113] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

[114] Charles C. Holt, Franco Modigliani, John F. Muth, and Herbert A. Simon. *Planning Production, Inventories, and Work Force*. Prentice Hall, 1960.

[115] Can Hou, Jiaxin Chen, Yaqing Zhou, Lei Hua, Jinxia Yuan, Shu He, Yi Guo, Sheng Zhang, Qiaowei Jia, Chenhui Zhao, Jing Zhang, Guangxu Xu, and Enzhi Jia. The effectiveness of quarantine of Wuhan city against the Corona Virus Disease 2019 (COVID-19): A well-mixed SEIR model analysis. *Journal of Medical Virology*, 92:841–848, 2020.

[116] Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *ArXiv Preprint*, page arXiv:1605.09674, 2016.

[117] Zhaolin Hu and L. Jeff Hong. Kullback-Leibler divergence constrained distributionally robust optimization. *Optimization Online preprint*, 2012.

[118] Minhui Huang, Shiqian Ma, and Lifeng Lai. A Riemannian block coordinate descent method for computing the projection robust Wasserstein distance. In *Proceedings of the 38th International Conference on Machine Learning*, pages 4446–4455, 2021.

[119] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.

[120] IMF. World economic outlook Oct 2021. *International Monetary Fund*, 2021.

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

[122] EA Jasmin, TP Imthias Ahamed, and VP Jagathiraj. A reinforcement learning algorithm to economic dispatch considering transmission losses. In *TENCON 2008-2008 IEEE Region 10 Conference*, pages 1–6. IEEE, 2008.

[123] EA Jasmin, TP Imthias Ahamed, and VP Jagathiraj. A reinforcement learning approach to economic dispatch using neural networks. In *Fifteenth National Power Systems Conference (NPSC)*, pages 84–89, 2008.

[124] Meryem Jefferies, Bisma Rauff, Harunor Rashid, Thao Lam, and Shafquat Rafiq. Update on global epidemiology of viral hepatitis and preventive strategies. *World Journal of Clinical Cases*, 6(13):589–599, 2018.

[125] Ruoxi Jia, Ming Jin, Kaiyu Sun, Tianzhen Hong, and Costas Spanos. Advanced building control via deep reinforcement learning. *Energy Procedia*, 158:6158–6163, 2019.

[126] Changxu Jiang, Zhigang Li, JH Zheng, QH Wu, and Xiaoya Shang. Two-level area-load modelling for OPF of power system using reinforcement learning. *IET Generation, Transmission & Distribution*, 13(18):4141–4149, 2019.

[127] Javier Jiménez-Raboso, Alejandro Campoy-Nieves, Antonio Manjavacas-Lucas, Juan Gómez-Romero, and Miguel Molina-Solana. Sinergym: A building simulation and control framework for training reinforcement learning agents. In *Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, BuildSys '21, page 319–323, 2021.

[128] Anatoli Juditsky, Philippe Rigollet, and Alexandre B Tsybakov. Learning by mirror averaging. *The Annals of Statistics*, 36(5):2183–2206, 2008.

[129] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the 19th International Conference on Machine Learning*, pages 267–274, 2002.

[130] Sham M Kakade. A natural policy gradient. In *Advances in Neural Information Processing Systems*, volume 14, 2001.

[131] Byung-Gook Kim, Yu Zhang, Mihaela Van Der Schaar, and Jang-Won Lee. Dynamic pricing and energy consumption scheduling with reinforcement learning. *IEEE Transactions on smart grid*, 7(5):2187–2198, 2015.

[132] Stacey Knobler, Adel Mahmoud, Stanley Lemon, Alison Mack, Laura Sivitz, and Katherine Oberholtzer. *Learning from SARS: Preparing for the Next Disease Outbreak.* The National Academies Press, 2004.

[133] W Bradley Knox and Peter Stone. Interactively shaping agents via human reinforcement: The tamer framework. In *International Conference on Knowledge Capture*, pages 9–16, 2009.

[134] W Bradley Knox and Peter Stone. Combining manual feedback with subsequent mdp reward signals for reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 5–12. Citeseer, 2010.

[135] W Bradley Knox and Peter Stone. Augmenting reinforcement learning with human feedback. In *ICML 2011 Workshop on New Developments in Imitation Learning*, volume 855, page 3, 2011.

[136] W Bradley Knox and Peter Stone. Reinforcement learning from simultaneous human and mdp reward. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 475–482, 2012.

[137] W Bradley Knox and Peter Stone. Learning non-myopically from human-generated reward. In *International Conference on Intelligent User Interfaces*, pages 191–202, 2013.

[138] W Bradley Knox and Peter Stone. Framing reinforcement learning from human reward: Reward positivity, temporal discounting, episodicity, and performance. *Artificial Intelligence*, 225:24–50, 2015.

[139] Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*, pages 1008–1014. Citeseer, 2000.

[140] Daniel Kuhn, Peyman Mohajerin Esfahani, Viet Anh Nguyen, and Soroosh Shafieezadeh-Abadeh. Wasserstein distributionally robust optimization: Theory and applications in machine learning. In *Operations Research & Management Science in the Age of Analytics*, pages 130–166. INFORMS, 2019.

[141] Roger Kwadzogah, Mengchu Zhou, and Sisi Li. Model predictive control for hvac systems—a review. In *2013 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 442–447. IEEE, 2013.

[142] Guanghui Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *Mathematical programming*, pages 1–48, 2022.

[143] Robert Leary. Global optimization on funneling landscapes. *Journal of Global Optimization*, 18:367–383, 2000.

[144] Phenyo E. Lekone and Bärbel F. Finkenstädt. Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy. *Biometrics*, 62(4):1170–1177, 2006.

[145] Geoff Levermore. *Building Energy Management Systems: An Application to Heating, Natural Ventilation, Lighting and Occupant Satisfaction.* Taylor and Francis, 1992.

[146] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.

[147] Fangyuan Li, Jiahu Qin, and Wei Xing Zheng. Distributed Q-learning-based online optimization algorithm for unit commitment and dispatch in smart grid. *IEEE transactions on cybernetics*, 50(9):4146–4156, 2019.

[148] Guangliang Li, Randy Gomez, Keisuke Nakamura, and Bo He. Human-centered reinforcement learning: A survey. *IEEE Transactions on Human-Machine Systems*, 49(4):337–349, 2019.

[149] Hepeng Li, Zhiqiang Wan, and Haibo He. Real-time residential demand response. *IEEE Transactions on Smart Grid*, 11(5):4144–4154, 2020.

[150] Jiawen Li and Tao Yu. Deep reinforcement learning based multi-objective integrated automatic generation control for multiple continuous power disturbances. *IEEE Access*, 8:156839–156850, 2020.

[151] 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. In *Proceedings of the 4th International Conference on Learning Representations*, 2016.

[152] Lin Lin, Xin Guan, Yu Peng, Ning Wang, Sabita Maharjan, and Tomoaki Ohtsuki. Deep reinforcement learning for economic dispatch of virtual power plant in internet of energy. *IEEE Internet of Things Journal*, 7(7):6288–6301, 2020.

[153] Tianyi Lin, Chenyou Fan, Nhat Ho, Marco Cuturi, and Michael I Jordan. Projection robust Wasserstein distance and Riemannian optimization. *ArXiv Preprint*, page arXiv:2006.07458, 2020.

[154] Haotian Liu and Wenchuan Wu. Two-stage deep reinforcement learning for inverter-based volt-var control in active distribution networks. *IEEE Transactions on Smart Grid*, 2020.

[155] Weirong Liu, Peng Zhuang, Hao Liang, Jun Peng, and Zhiwu Huang. Distributed economic dispatch in microgrids based on cooperative reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2192–2203, 2018.

[156] Robert Loftin, James MacGlashan, Bei Peng, Matthew Taylor, Michael Littman, Jeff Huang, and David Roberts. A strategy-aware technique for learning behaviors from discrete human feedback. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, volume 28, 2014.

[157] Robert Loftin, Bei Peng, James MacGlashan, Michael L Littman, Matthew E Taylor, Jeff Huang, and David L Roberts. Learning behaviors via human-delivered discrete feedback: modeling implicit feedback strategies to speed up learning. *Autonomous Agents and Multi-agent Systems*, 30(1):30–59, 2016.

[158] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *ArXiv Preprint*, page arXiv:1706.02275, 2017.

[159] Renzhi Lu and Seung Ho Hong. Incentive-based demand response for smart grid with reinforcement learning and deep neural network. *Applied energy*, 236:937–949, 2019.

[160] Renzhi Lu, Seung Ho Hong, and Xiongfeng Zhang. A dynamic pricing demand response algorithm for smart grid: reinforcement learning approach. *Applied Energy*, 220:220–230, 2018.

[161] Renzhi Lu, Seung Ho Hong, and Xiongfeng Zhang. A dynamic pricing demand response algorithm for smart grid: Reinforcement learning approach. *Applied Energy*, 220:220–230, 2018.

[162] Fengqiao Luo and Sanjay Mehrotra. Distributionally robust optimization with decision dependent ambiguity sets. In *Optimization Letters*, 2020.

[163] Leonardo López and Xavier Rodó. A modified SEIR model to predict the COVID-19 outbreak in Spain and Italy: Simulating control scenarios and multi-scale epidemics. *Results in Physics*, 21:103746, 2021.

[164] James MacGlashan, Mark K Ho, Robert Loftin, Bei Peng, Guan Wang, David L Roberts, Matthew E Taylor, and Michael L Littman. Interactive learning from policy-dependent human feedback. In *International Conference on Machine Learning*, pages 2285–2294, 2017.

[165] Sabita Maharjan, Quanyan Zhu, Yan Zhang, Stein Gjessing, and Tamer Basar. Dependable demand response management in the smart grid: A stackelberg game approach. *IEEE Transactions on Smart Grid*, 4(1):120–132, 2013.

[166] Mausam and Andrey Kolobov. *Planning with Markov Decision Processes: An AI Perspective*. Morgan Claypool, 2012.

[167] Garth P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I — Convex underestimating problems. *Mathematical Programming*, 10:147–175, 1976.

[168] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 1928–1937, 2016.

[169] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 1928–1937, 2016.

[170] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with deep reinforcement learning. *ArXiv Preprint*, page arXiv:1312.5602, 2013.

[171] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King,

Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[172] Elena Mocanu, Decebal Constantin Mocanu, Phuong H. Nguyen, Antonio Liotta, Michael E. Webber, Madeleine Gibescu, and J. G. Slootweg. On-line building energy optimization using deep reinforcement learning. *IEEE Transactions on Smart Grid*, 10(4):3698–3708, 2019.

[173] Thomas M. Moerland, Joost Broekens, Aske Plaat, and Catholijn M. Jonker. Model-based reinforcement learning: A survey. *Foundations and Trends in Machine Learning*, 16(1):1–118, 2023.

[174] Andrew William Moore. Efficient memory-based learning for robot control. Technical report, University of Cambridge Computer Laboratory, 1990.

[175] Douglas William Moore. Simplicial mesh generation with applications. *PhD Thesis, Cornell University, Department of Computer Science*, 1992.

[176] Ted Moskovitz, Michael Arbel, Ferenc Huszar, and Arthur Gretton. Efficient Wasserstein natural gradients for reinforcement learning. *ArXiv Preprint*, page arXiv:2010.05380, 2020.

[177] Ted Moskovitz, Michael Arbel, Ferenc Huszar, and Arthur Gretton. Efficient Wasserstein natural gradients for reinforcement learning. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[178] Rémi Munos and Andrew Moore. Variable resolution discretization in optimal control. *Machine Learning*, 49:291–323, 2002.

[179] Hideaki Nakao, Ruiwei Jiang, and Siqian Shen. Distributionally robust partially observable Markov decision process with moment-based ambiguity. *SIAM Journal on Optimization*, 31(1):461–488, 2021.

[180] Nandan Kumar Navin and Rajneesh Sharma. A fuzzy reinforcement learning approach

to thermal unit commitment problem. *Neural Computing and Applications*, 31(3):737–750, 2019.

[181] Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

[182] Andrew Y Ng, Stuart J Russell, et al. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning*, volume 1, page 2, 2000.

[183] Viet Anh Nguyen, Soroosh Shafieezadeh Abadeh, Man-Chung Yue, Daniel Kuhn, and Wolfram Wiesemann. Optimistic distributionally robust optimization for nonparametric likelihood approximation. In *Advances in Neural Information Processing Systems*, pages 15846–15856, 2019.

[184] Arnab Nilim and Laurent El Ghaoui. Robustness in Markov decision problems with uncertain transition matrices. In *Advances in Neural Information Processing Systems*, pages 839–846, 2004.

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

[186] Ali Nouri. *Efficient Model-based Exploration in Continuous State-space Environments*. Rutgers The State University of New Jersey-New Brunswick, 2011.

[187] Nilay Noyan, Gabor Rudolf, and Miguel Lejeune. Distributionally robust optimization under decision-dependent ambiguity set with an application to machine scheduling. *Optimization Online Preprint*, 2020.

[188] Takayuki Osogami. Robustness and risk-sensitivity in Markov decision processes. In *Advances in Neural Information Processing Systems*, pages 233–241, 2012.

[189] Aldo Pacchiano, Jack Parker-Holder, Yunhao Tang, Krzysztof Choromanski, Anna Choromanska, and Michael Jordan. Learning to score behaviors for guided policy optimization. In *Proceedings of the 37th International Conference on Machine Learning*, pages 7445–7454, 2020.

[190] Victor M. Panaretos and Yoav Zemel. Statistical aspects of wasserstein distances. *Annual Review of Statistics and Its Application*, 6(1):405–431, 2019.

[191] Matteo Papini, Andrea Battistello, and Marcello Restelli. Balancing learning speed and stability in policy gradient via adaptive exploration. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pages 1188–1199, 2020.

[192] Hoda Parvin, Piyush Goel, and Natarajan Gautam. An analytic framework to develop policies for testing, prevention, and treatment of two-stage contagious diseases. *Annals of Operations Research*, 196(1):707–735, 2012.

[193] 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, 2019.

[194] Judea Pearl. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley Longman Publishing Co., Inc., 1984.

[195] Joao Pedro Pedroso. Hybrid enumeration strategies for mixed integer programming. *Technical Report Series: DCC-2004-8, Universidade do Porto*, 2004.

[196] Bei Peng, James MacGlashan, Robert Loftin, Michael L Littman, David L Roberts, and Matthew E Taylor. A need for speed: Adapting agent action speed to improve task learning from non-expert humans. In *International Joint Conference on Autonomous Agents and Multiagent Systems*, 2016.

[197] Liangrong Peng, Wuyue Yang, Dongyan Zhang, Changjing Zhuge, and Liu Hong. Epidemic analysis of COVID-19 in China by dynamical modeling. *Arxiv Preprint ArXiv:2002.06563*, 2020.

[198] Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *ArXiv Preprint*, page arXiv:1910.00177, 2019.

[199] Jan Peters and Stefan Schaal. Policy gradient methods for robotics. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2219–2225, 2006.

[200] Henning Petzka, Asja Fischer, and Denis Lukovnicov. On the regularization of Wasserstein GANs. *ArXiv Preprint*, page arXiv:1709.08894, 2017.

[201] Patrick M Pilarski, Michael R Dawson, Thomas Degris, Farbod Fahimi, Jason P Carey, and Richard S Sutton. Online human training of a myoelectric prosthesis controller via actor-critic reinforcement learning. In *IEEE international Conference on Rehabilitation Robotics*, pages 1–7. IEEE, 2011.

[202] Pascal Poupart, Craig Boutilier, Relu Patrascu, and Dale Schuurmans. Piecewise linear value function approximation for factored MDPs. *Proceedings of the National Conference on Artificial Intelligence*, 05 2002.

[203] Hamed Rahimian and Sanjay Mehrotra. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

[204] T Remani, EA Jasmin, and TP Imthias Ahamed. Residential load scheduling with renewable generation in the smart grid: A reinforcement learning approach. *IEEE Systems Journal*, 13(3):3283–3294, 2018.

[205] Ehsan Rezaei, Hanane Dagdougui, and Mahdi Rezaei. Distributed stochastic model predictive control for peak load limiting in networked microgrids with building thermal dynamics. *IEEE Transactions on Smart Grid*, 13(3):2038–2049, 2022.

[206] Pierre H. Richemond and Brendan Maginnis. On Wasserstein reinforcement learning and the Fokker-Planck equation. *ArXiv Preprint*, page arXiv:1712.07185, 2017.

[207] Philippe Rigollet and Alexandre Tsybakov. Exponential screening and optimal rates of sparse estimation. *The Annals of Statistics*, 39(2):731–771, 2011.

[208] R. Tyrrell Rockafellar and Roger J. B. Wets. *Variational Analysis*. Springer, 1998.

[209] Johannes O. Royset. Approximations and solution estimates in optimization. *Mathematical Programming*, 170:479–506, 2018.

[210] Johannes O. Royset and Roger J.-B. Wets. Variational theory for optimization under stochastic ambiguity. *SIAM Journal on Optimization*, 27(2):1118–1149, 2017.

[211] Stuart Russell. Learning agents for uncertain environments. In *Annual Conference on Computational Learning Theory*, pages 101–103, 1998.

[212] Régis Sabbadin and Anne-France Viet. A tractable leader-follower MDP model for animal disease management. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, page 1320–1326. AAAI Press, 2013.

[213] Jyri Salpakari and Peter Lund. Optimal and rule-based control strategies for energy flexibility in buildings with PV. *Applied Energy*, 161:425–436, 2016.

[214] William Saunders, Girish Sastry, Andreas Stuhlmueller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. *ArXiv Preprint*, page arXiv:1707.05173, 2017.

[215] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *ArXiv Preprint*, page arXiv:1511.05952, 2015.

[216] Jürgen Schmidhuber. Curious model-building control systems. In *International Joint Conference on Neural Networks*, pages 1458–1463, 1991.

[217] Fabio Schoen. Two-phase methods for global optimization. *Handbook of Global Optimization*, pages 151–177, 2002.

[218] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

[219] John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1889–1897, 2015.

[220] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

[221] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *Proceedings of the 4th International Conference on Learning Representations*, 2016.

[222] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *ArXiv Preprint*, page arXiv:1707.06347, 2017.

[223] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[224] Lior Shani, Yonathan Efroni, and Shie Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5668–5675, 2020.

[225] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory, Second Edition.* Society for Industrial and Applied Mathematics, 2014.

[226] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[227] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *Proceedings of the 31st International Conference on Machine Learning*, pages 387–395, 2014.

[228] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin

Riedmiller. Deterministic policy gradient algorithms. In *International Conference on Machine Learning*, pages 387–395, 2014.

[229] Vijay Pratap Singh, Nand Kishor, and Paulson Samuel. Distributed multi-agent system-based load frequency control for multi-area power system in smart grid. *IEEE Transactions on Industrial Electronics*, 64(6):5151–5160, 2017.

[230] Maurice Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, 1958.

[231] Henrik Sjödin, Annelies Wilder-Smith, Sarah Osman, Zia Farooq, and Joacim Rocklöv. Only strict quarantine measures can curb the coronavirus disease (COVID-19) outbreak in Italy, 2020. *Eurosurveillance*, 25(13), 2020.

[232] Edward Jay Sondik. The optimal control of partially observable Markov decision processes. *PhD Thesis, Stanford University*, 1971.

[233] Chunhe Song, Guangjie Han, and Peng Zeng. Cloud computing based demand response management using deep reinforcement learning. *IEEE Transactions on Cloud Computing*, 10(1):72–81, 2022.

[234] H. Francis Song, Abbas Abdolmaleki, Jost Tobias Springenberg, Aidan Clark, Hubert Soyer, Jack W. Rae, Seb Noury, Arun Ahuja, Siqi Liu, Dhruva Tirumala, Nicolas Heess, Dan Belov, Martin Riedmiller, and Matthew M. Botvinick. V-MPO: On-Policy maximum a posteriori policy optimization for discrete and continuous Control. In *Proceedings of the 8th International Conference on Learning Representations*, 2020.

[235] Jun Song, Niao He, Lijun Ding, and Chaoyue Zhao. Provably convergent policy optimization via metric-aware trust region methods. *Transactions on Machine Learning Research*, 2023.

[236] Jun Song and Chaoyue Zhao. Optimistic distributionally robust policy optimization. *arXiv preprint arXiv:2006.07815*, 2020.

[237] Jun Song and Chaoyue Zhao. Optimistic distributionally robust policy optimization. *ArXiv Preprint*, page arXiv:2006.07815, 2020.

[238] Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

[239] Jan Storck, Sepp Hochreiter, and Jürgen Schmidhuber. Reinforcement driven information acquisition in non-deterministic environments. In *International Conference on Artificial Neural Networks*, volume 2, pages 159–164. Citeseer, 1995.

[240] Student. The probable error of a mean. *Biometrika*, 6(1):1–25, 1908.

[241] Yi Sun, Faustino Gomez, and Jürgen Schmidhuber. Planning to be surprised: Optimal bayesian exploration in dynamic environments. In *International Conference on Artificial General Intelligence*, pages 41–51. Springer, 2011.

[242] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[243] Richard S. Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, pages 1057–1063, 1999.

[244] Aviv Tamar, Yinlam Chow, Mohammad Ghavamzadeh, and Shie Mannor. Policy gradient for coherent risk measures. *ArXiv Preprint*, page arXiv:1502.03919, 2015.

[245] Biao Tang, Fan Xia, Sanyi Tang, Nicola Bragazzi, Qian Li, Xiaodan Sun, Juhua Liang, Yanni Xiao, and Jianhong Wu. The effectiveness of quarantine and isolation determine the trend of the COVID-19 epidemics in the final phase of the current outbreak in China. *International Journal of Infectious Diseases*, 95:288–293, 2020.

[246] Yunhao Tang and Shipra Agrawal. Discretizing continuous action space for on-policy optimization. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, volume 34. AAAI Press, 2020.

[247] Gerald Tesauro. Temporal difference learning and td-gammon. *Communications of the ACM*, 38(3):58–68, 1995.

[248] Chen Tessler, Guy Tennenholtz, and Shie Mannor. Distributional policy optimization: An alternative approach for continuous control. In *Advances in Neural Information Processing Systems*, pages 1350–1360, 2019.

[249] Chen Tessler, Guy Tennenholtz, and Shie Mannor. Distributional policy optimization: An alternative approach for continuous control. *Advances in Neural Information Processing Systems*, 32, 2019.

[250] Andrea L Thomaz and Cynthia Breazeal. Teachable robots: Understanding human teaching behavior to build more effective robot learners. *Artificial Intelligence*, 172(6-7):716–737, 2008.

[251] Andrea Lockerd Thomaz, Cynthia Breazeal, et al. Reinforcement learning with human teachers: Evidence of feedback and guidance with implications for learning performance. In *Association for the Advancement of Artificial Intelligence*, volume 6, pages 1000–1005. Boston, MA, 2006.

[252] Naftali Tishby, Esther Levin, and Sara A Solla. Consistent inference of probabilities in layered networks: Predictions and generalization. In *International Joint Conference on Neural Networks*, volume 2, pages 403–409, 1989.

[253] Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.

[254] John N Tsitsiklis and Benjamin Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5):674–690, 1997.

[255] William Valladares, Marco Galindo, Jorge Gutiérrez, Wu-Chieh Wu, Kuo-Kai Liao, Jen-Chung Liao, Kuang-Chin Lu, and Chi-Chuan Wang. Energy optimization associated

with thermal comfort and indoor air control via a deep reinforcement learning algorithm. *Building and Environment*, 155:105–117, 2019.

[256] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double Q-learning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, volume 30, 2016.

[257] Ngo Anh Vien and Wolfgang Ertel. Reinforcement learning combined with human feedback in continuous state and action spaces. In *IEEE International Conference on Development and Learning and Epigenetic Robotics*, pages 1–6. IEEE, 2012.

[258] Anne-France Viet, Stéphane Krebs, Olivier Rat-Aspert, Laurent Jeanpierre, Catherine Belloc, and Pauline Ezanno. A modelling framework based on MDP to coordinate farmers' disease control decisions at a regional scale. *PLOS ONE*, 13(6):1–20, 2018.

[259] José R. Vázquez-Canteli and Zoltán Nagy. Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Applied Energy*, 235:1072–1089, 2019.

[260] David Wales and Jonathan Doye. Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *The Journal of Physical Chemistry A*, 101(28):5111–5116, 1998.

[261] Huaizhi Wang, Zhenxing Lei, Xian Zhang, Jianchun Peng, and Hui Jiang. Multiobjective reinforcement learning-based intelligent approach for optimization of activation rules in automatic generation control. *IEEE Access*, 7:17480–17492, 2019.

[262] Jie Wang, Rui Gao, and Yao Xie. Sinkhorn distributionally robust optimization. *ArXiv Preprint*, page arXiv:2109.11926, 2021.

[263] Jie Wang, Rui Gao, and Yao Xie. Two-sample test using projected Wasserstein distance. In *Proceedings of IEEE International Symposium on Information Theory*, volume 21, 2021.

[264] Shengyi Wang, Jiajun Duan, Di Shi, Chunlei Xu, Haifeng Li, Ruisheng Diao, and Zhiwei Wang. A data-driven multi-agent autonomous voltage control framework using

deep reinforcement learning. *IEEE Transactions on Power Systems*, 35(6):4644–4654, 2020.

[265] Yuan Wang, Kirubakaran Velswamy, and Biao Huang. A long-short term memory recurrent neural network based reinforcement learning controller for office heating ventilation and air conditioning systems. *Processes*, 5(3), 2017.

[266] Yuhui Wang, Hao He, Xiaoyang Tan, and Yaozhong Gan. Trust region-guided proximal policy optimization. *ArXiv Preprint*, page arXiv:1901.10314, 2019.

[267] Ziyu Wang, Victor Bapst, Nicolas Heess, Volodymyr Mnih, Remi Munos, Koray Kavukcuoglu, and Nando de Freitas. Sample efficient actor-critic with experience replay. *ArXiv Preprint*, page arXiv:1611.01224, 2016.

[268] Ziyu Wang, Victor Bapst, Nicolas Heess, Volodymyr Mnih, Remi Munos, Koray Kavukcuoglu, and Nando de Freitas. Sample efficient actor-critic with experience replay. In *Proceedings of the 5th International Conference on Learning Representations*, 2017.

[269] 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, 2016.

[270] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, 1992.

[271] Tianshu Wei, Yanzhi Wang, and Qi Zhu. Deep reinforcement learning for building hvac control. In *2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pages 1–6, 2017.

[272] Stefan Weinzierl. Introduction to Monte Carlo methods. *Arxiv Preprint ArXiv:hep-ph/0006269*, 2000.

[273] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *International Conference on Machine Learning*, pages 681–688. Citeseer, 2011.

[274] Chelsea C. White and Hany K. Eldeib. Markov decision processes with imprecise transition probabilities. *Operations Research*, 42(4):739–749, 1992.

[275] David Silver Will Dabney, Georg Ostrovski and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International Conference on Machine Learning*, 2018.

[276] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

[277] Yuhuai Wu, Elman Mansimov, Shun Liao, Roger Grosse, and Jimmy Ba. Scalable trust-region method for deep reinforcement learning using Kronecker-factored approximation. In *Advances in Neural Information Processing Systems*, page 5285–5294, 2017.

[278] Huang Xiao, Michael Herman, Joerg Wagner, Sebastian Ziesche, Jalal Etesami, and Thai Hong Linh. Wasserstein adversarial imitation learning. *ArXiv Preprint*, page arXiv:1906.08113, 2019.

[279] Hanchen Xu, Alejandro D Domínguez-García, and Peter W Sauer. Optimal tap setting of voltage regulation transformers using batch reinforcement learning. *IEEE Transactions on Power Systems*, 35(3):1990–2001, 2019.

[280] Huan Xu and Shie Mannor. Distributionally robust Markov decision processes. In *Advances in Neural Information Processing Systems*, pages 2505–2513, 2010.

[281] Liu Ya, Zhang Deliang, and Wang Xuanyuan. A peak regulation ancillary service optimal dispatch method of virtual power plant based on reinforcement learning. In *2019 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)*, pages 4356–4361. IEEE, 2019.

[282] Reza Yaesoubi and Ted Cohen. Dynamic health policies for controlling the spread of emerging infections: Influenza as an example. *PLOS ONE*, 6(9):1–11, 2011.

[283] Xiefei Yan and Yun Zou. Optimal and sub-optimal quarantine and isolation control in sars epidemics. *Mathematical and Computer Modelling*, 47(1):235–245, 2008.

[284] Ziming Yan and Yan Xu. A multi-agent deep reinforcement learning method for cooperative load frequency control of a multi-area power system. *IEEE Transactions on Power Systems*, 35(6):4599–4608, 2020.

[285] Ziming Yan and Yan Xu. Real-time optimal power flow: A Lagrangian based deep reinforcement learning approach. *IEEE Transactions on Power Systems*, 35(4):3270–3273, 2020.

[286] Insoon Yang. A convex optimization approach to distributionally robust Markov decision processes with Wasserstein distance. *IEEE Control Systems Letters*, 1(1):164–169, 2017.

[287] Insoon Yang. A dynamic game approach to distributionally robust safety specifications for stochastic systems. *Automatica*, 94:94–101, 2017.

[288] Qiuling Yang, Gang Wang, Alireza Sadeghi, Georgios B Giannakis, and Jian Sun. Two-timescale voltage control in distribution grids using deep reinforcement learning. *IEEE Transactions on Smart Grid*, 11(3):2313–2323, 2019.

[289] Ye Yao and Divyanshu Kumar Shekhar. State of the art review on model predictive control (mpc) in heating ventilation and air-conditioning (hvac) field. *Building and Environment*, 200:107952, 2021.

[290] Abdollah Younesi, Hossein Shayeghi, and Pierluigi Siano. Assessing the use of reinforcement learning for integrated voltage/frequency control in AC microgrids. *Energies*, 13(5):1250, 2020.

[291] Mengmeng Yu and Seung Hong. Supply–demand balancing for power management in smart grid: A stackelberg game approach. *Applied Energy*, 164:702–710, 2016.

[292] Mengmeng Yu and Seung Ho Hong. A real-time demand-response algorithm for smart grids: A stackelberg game approach. *IEEE Transactions on smart grid*, 7(2):879–888, 2015.

[293] Mengmeng Yu and Seung Ho Hong. Incentive-based demand response considering hierarchical electricity market: A Stackelberg game approach. *Applied Energy*, 203(C):267–279, 2017.

[294] Pengqian Yu and Huan Xu. Distributionally robust counterpart in Markov decision processes. *IEEE Transactions on Automatic Control*, 61(9):2538 – 2543, 2016.

[295] Xian Yu and Siqian Shen. Multistage distributionally robust mixed-integer programming with decision-dependent moment-based ambiguity sets. *Arxiv Preprint ArXiv:2002.12518*, 2020.

[296] Tom Zahavy, Matan Haroush, Nadav Merlis, Daniel J Mankowitz, and Shie Mannor. Learn what not to learn: Action elimination with deep reinforcement learning. *ArXiv Preprint*, page arXiv:1809.02121, 2018.

[297] Lixin Zhan, Jeff Chen, and Wing-Ki Liu. Monte Carlo basin paving: An improved global optimization method. *Physical Review E*, 73:015701, 2006.

[298] Amy Zhang, Rowan Thomas McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[299] Jie Zhang, Huifu Xu, and Liwei Zhang. Quantitative stability analysis for distributionally robust optimization with moment constraints. *SIAM Journal on Optimization*, 26:1855–1882, 2016.

[300] Ruiyi Zhang, Changyou Chen, Chunyuan Li, and Lawrence Carin. Policy optimization as Wasserstein gradient flows. In *Proceedings of the 35th International Conference on Machine Learning*, pages 5737–5746, 2018.

[301] Xiangyu Zhang, Dave Biagioni, Mengmeng Cai, Peter Graf, and Saifur Rahman. An edge-cloud integrated solution for buildings demand response using reinforcement learning. *IEEE Transactions on Smart Grid*, 12(1):420–431, 2021.

[302] Xiangyu Zhang, Yue Chen, Andrey Bernstein, Rohit Chintala, Peter Graf, Xin Jin, and David Biagioni. Two-stage reinforcement learning policy search for grid-interactive building control. *IEEE Transactions on Smart Grid*, 13(3):1976–1987, 2022.

[303] Xiaoshun Zhang, Tao Bao, Tao Yu, Bo Yang, and Chuanjia Han. Deep transfer Q-learning with virtual leader-follower for supply-demand stackelberg game of smart grid. *Energy*, 133:348–365, 2017.

[304] Zhiang Zhang, Adrian Chong, Yuqi Pan, Chenlu Zhang, and Khee Poh Lam. Whole building energy model for hvac optimal control: A practical framework based on deep reinforcement learning. *Energy and Buildings*, 199:472–490, 2019.

[305] Zhiang Zhang and Khee Poh Lam. Practical implementation and evaluation of deep reinforcement learning control for a radiant heating system. In *Proceedings of the 5th Conference on Systems for Built Environments*, BuildSys '18, page 148–157. Association for Computing Machinery, 2018.

[306] Zhiang Zhang and Khee Poh Lam. Practical implementation and evaluation of deep reinforcement learning control for a radiant heating system. In *Proceedings of the 5th Conference on Systems for Built Environments*, BuildSys '18, page 148–157, 2018.

[307] Chaoyue Zhao. *Data-driven risk-averse stochastic program and renewable energy integration*. University of Florida, 2014.

[308] Chaoyue Zhao and Yongpei Guan. Unified stochastic and robust unit commitment. *IEEE Transactions on Power Systems*, 28(3):3353–3361, 2013.

[309] Chaoyue Zhao and Yongpei Guan. Data-driven risk-averse stochastic optimization with Wasserstein metric. *Operations Research Letters*, 46(2):262–267, 2018.

[310] Chaoyue Zhao and Ruiwei Jiang. Distributionally robust contingency-constrained unit commitment. *IEEE Transactions on Power Systems*, 33(1):94–102, 2017.

[311] Chaoyue Zhao, Jianhui Wang, Jean-Paul Watson, and Yongpei Guan. Multi-stage

robust unit commitment considering wind and demand response uncertainties. *IEEE Transactions on Power Systems*, 28(3):2708–2717, 2013.

[312] Chaoyue Zhao, Qianfan Wang, Jianhui Wang, and Yongpei Guan. Expected value and chance constrained stochastic unit commitment ensuring wind power utilization. *IEEE Transactions on Power Systems*, 29(6):2696–2705, 2014.

[313] Shilei Zhao and Hua Chen. Modeling the epidemic dynamics and control of COVID-19 outbreak in China. *Quantative Biology*, pages 1–9, 2020.

[314] Steve Zymler, Daniel Kuhn, and Berç Rustem. Distributionally robust joint chance constraints with second-order moment information. *Mathematical Programming*, 137(1–2):167–198, 2011.

Appendix A

APPENDIX FOR CHAPTER 2

A.1 Proof of Theorem 1

Theorem 1. *If for any $a \in \mathcal{A}$, the ambiguity set defined in* (2.6) *is nonempty, then* (2.7) *can be reformulated as:*

$$\begin{aligned} V^t(\xi) = \max_{a\in\mathcal{A},\boldsymbol{w},\boldsymbol{u},q} \; & \tilde{r}_{a\xi} + q - \boldsymbol{w}^T\tilde{\boldsymbol{\eta}}^U_{a\xi} + \boldsymbol{u}^T\tilde{\boldsymbol{\eta}}^L_{a\xi} \\ s.t. \quad & q\mathbf{1} \le \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}, \\ & \boldsymbol{w} + \boldsymbol{u} \le k\mathbf{1}, \\ & \boldsymbol{w}, \boldsymbol{u} \ge \mathbf{0}. \end{aligned} \tag{2.8}$$

Proof. We first obtain the dual formulation of (2.7) as follows:

$$\max_{z,\boldsymbol{w},\boldsymbol{u}} L'(z,\boldsymbol{w},\boldsymbol{u}) = \; \tilde{r}_{a\xi} + z - \boldsymbol{w}^{\mathrm{T}}\tilde{\boldsymbol{\eta}}^U_{a\xi} + \boldsymbol{u}^{\mathrm{T}}\tilde{\boldsymbol{\eta}}^L_{a\xi} \tag{A.1a}$$

$$s.t. \quad \lambda \boldsymbol{p}^{\mathrm{T}}_{a\xi}\boldsymbol{V}^{t+1} - z + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{p}_{a\xi} - \boldsymbol{u}^{\mathrm{T}}\boldsymbol{p}_{a\xi} \ge 0, \quad \forall \boldsymbol{p}_{a\xi} \in \Delta(\tilde{\mathcal{S}}), \tag{A.1b}$$

$$\boldsymbol{w} + \boldsymbol{u} \le k\mathbf{1}, \tag{A.1c}$$

$$\boldsymbol{w}, \boldsymbol{u} \ge \mathbf{0} \tag{A.1d}$$

where z with unrestricted sign, $\boldsymbol{w} \ge 0$, and $\boldsymbol{u} \ge 0$ are dual variables for constraints (2.7b), (2.7c), and (2.7d) respectively. Next, we prove that the strong duality is met. If there exists a point $\bar{\boldsymbol{x}}, \bar{\mu}_{a\xi}$ in the feasible region of (2.7) such that all inequality constraints are non-binding, the Slater condition is satisfied and strong duality holds. Note that for any $\mu_{a\xi}, \boldsymbol{x}$ that are in the feasible region of (2.7), if one of the inequality constraints (2.7c) or (2.7d) is binding, we can let $\bar{\boldsymbol{x}} = \boldsymbol{x} + \epsilon$ for any $\epsilon > 0$. Letting $\bar{\mu}_{a\xi} = \mu_{a\xi}$ we have $\bar{\boldsymbol{x}}, \bar{\mu}_{a\xi}$ satisfy the Slater condition. Therefore strong duality holds and the primal and dual objectives (2.7a) and (A.1a) are equal.

For some $z, \boldsymbol{w}$ and $\boldsymbol{u}$, (A.1b) is satisfied if its left-hand-side is non-negative for all $\boldsymbol{p}_{a\xi} \in \Delta(\tilde{\mathcal{S}})$. Therefore, this is equaivalent to the left hand side being non-negative for the

minimum such $\boldsymbol{p}_{a\xi}$ that is in the probability simplex. Therefore (A.1b) can be reformulated as the following.

$$\begin{aligned} \min_{\boldsymbol{p}_{a\xi} \geq 0} \quad & \lambda \boldsymbol{p}_{a\xi}^{\mathrm{T}} \boldsymbol{V}^{t+1} - z + \boldsymbol{w}^{\mathrm{T}} \boldsymbol{p}_{a\xi} - \boldsymbol{u}^{\mathrm{T}} \boldsymbol{p}_{a\xi} \geq 0 \\ s.t. \quad & \mathbf{1}^{\mathrm{T}} \boldsymbol{p}_{a\xi} = 1. \end{aligned} \tag{A.2}$$

Next, we take the dual of the (A.2). We introduce the dual variable q, and using the same dualization procedure described before, we arrive at the following formulation for the dual of (A.2):

$$\begin{aligned} \max_{q} \quad & q - z \geq 0 \\ s.t. \quad & q\mathbf{1} \leq \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}. \end{aligned} \tag{A.3}$$

Substituing (A.3) into (A.1b), we arrive at the following reformulation of (A.1):

$$\begin{aligned} \max_{z, \boldsymbol{w}, \boldsymbol{u}, q} \quad & \tilde{r}_{a\xi} + z - \boldsymbol{w}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}_{a\xi}^{U} + \boldsymbol{u}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}_{a\xi}^{L} \\ s.t. \quad & q - z \geq 0, \\ & q\mathbf{1} \leq \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}, \\ & \boldsymbol{w} + \boldsymbol{u} \leq k\mathbf{1}, \\ & \boldsymbol{w}, \boldsymbol{u} \geq \mathbf{0}, \end{aligned}$$

which can be further simplified as the following because $q = z$ at optimality:

$$\begin{aligned} \max_{\boldsymbol{w}, \boldsymbol{u}, q} \quad & \tilde{r}_{a\xi} + q - \boldsymbol{w}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}_{a\xi}^{U} + \boldsymbol{u}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}_{a\xi}^{L} \\ s.t. \quad & q\mathbf{1} \leq \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}, \\ & \boldsymbol{w} + \boldsymbol{u} \leq k\mathbf{1}, \\ & \boldsymbol{w}, \boldsymbol{u} \geq \mathbf{0}. \end{aligned}$$

We have now reformulated the inner problem. The final step is to combine this result with the outer maximization problem that includes the action $a \in \mathcal{A}$ as a decision variable to

obtain a reformulation of (2.5)

$$
\begin{aligned}
V^t(\xi) = \max_{a \in \mathcal{A}, \boldsymbol{w}, \boldsymbol{u}, q} \; & \tilde{r}_{a\xi} + q - \boldsymbol{w}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}^U_{a\xi} + \boldsymbol{u}^{\mathrm{T}} \tilde{\boldsymbol{\eta}}^L_{a\xi} \\
s.t. \quad & q\mathbf{1} \leq \lambda \boldsymbol{V}^{t+1} + \boldsymbol{w} - \boldsymbol{u}, \\
& \boldsymbol{w} + \boldsymbol{u} \leq k\mathbf{1}, \\
& \boldsymbol{w}, \boldsymbol{u} \geq \mathbf{0}.
\end{aligned}
$$

□

Appendix B

APPENDIX FOR CHAPTER 3

B.1 Proof of Theorem 3

Theorem 3. *If Assumption 1 holds, then the KL trust-region constrained optimization problem in* (3.8) *is equivalent to the following problem:*

$$\min_{\beta\geq 0}\{l_0(\beta) := \beta\delta + \mathbb{E}_{s\sim\rho^\pi}\beta\log\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}]\}. \tag{3.9}$$

Proof. Let $L_s(a) = \dfrac{\pi'(a|s)}{\pi(a|s)}$. Denote $\mathbb{L}_s = \{L'_s \mid \mathbb{E}_{a\sim\pi(\cdot|s)}[L'_s(a)] = 1,\ L'_s \geq 0\}$. It's easy to prove that $L_s \in \mathbb{L}_s$. By using the importance sampling and the definition of KL divergence, we have: $\mathbb{E}_{a\sim\pi'(\cdot|s)}[A^\pi(s,a)] = \mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)L_s(a)]$, and $d_{KL}(\pi'(\cdot|s),\pi(\cdot|s)) = \mathbb{E}_{a\sim\pi'(\cdot|s)}[\log L_s(a)] = \mathbb{E}_{a\sim\pi(\cdot|s)}[L_s(a)\log L_s(a)]$.

Thus, we can reformulate (3.8) with KL divergence based trust region as:

$$\begin{aligned}\max_{L_s\in\mathbb{L}_s}\quad & \mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)L_s(a)]\\ s.t.\quad & \mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[L_s(a)\log L_s(a)]\leq\delta.\end{aligned} \tag{B.1}$$

First, it is easy to prove that for $\forall s,a$, $\mathbb{E}_{a\sim\pi(\cdot|s)}[L_s(a)]$ and $\mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)L_s(a)]$ are linear functions of $L_s(a)$, and $\mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[L_s(a)\log L_s(a)]$ is a convex function of $L_s(a)$. In addition, Slater's condition holds for (B.1) since there is an interior point $L_s(a) = 1$ $\forall s,a$. Meanwhile, since $A^\pi(s,a)$ is bounded following from Assumption 1, the objective is bounded above. Therefore, strong duality holds for (B.1). To reformulate (B.1), we consider its Lagrangian duality function:

$$\begin{aligned}l_0(\beta, L_s) &= \mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)L_s(a)]\\ &\quad -\beta\{\mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[L_s(a)\log L_s(a)]-\delta\}\\ &= \mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)L_s(a)\\ &\quad -\beta L_s(a)\log L_s(a)]+\beta\delta,\end{aligned}$$

where β is the dual variable. Then, (B.1) is equivalent to its dual problem as follows:

$$\min_{\beta \geq 0} \max_{L_s \in \mathbb{L}_s} l_0(\beta, L_s). \tag{B.2}$$

The inner maximization problem of (B.2) is equivalent to:

$$\begin{aligned} \max_{L_s \geq 0} \quad & \mathbb{E}_{s\sim\rho^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)L_s(a) - \beta L_s(a)\log L_s(a)] + \beta\delta \\ & s.t. \quad \mathbb{E}_{a\sim\pi(\cdot|s)}[L_s(a)] = 1, \quad \forall s \in \mathcal{S}. \end{aligned} \tag{B.3}$$

By Theorem 1 of [117], we can obtain the optimal solution L_s^* and the optimal objective value of the inner maximization problem (B.3) respectively as follows:

$$L_s^*(a) = \frac{e^{A^\pi(s,a)/\beta}}{\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}]},$$

$$l_0(\beta, L_s^*) = \beta\delta + \mathbb{E}_{s\sim\rho^\pi}\beta\log\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}].$$

Therefore, (B.2) can be further reformulated as:

$$\min_{\beta\geq 0} l_0(\beta, L_s^*) = \min_{\beta\geq 0}\{\beta\delta + \mathbb{E}_{s\sim\rho^\pi}\beta\log\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta}]\}.$$

□

B.2 Proof of Theorem 4

Theorem 4. *If Assumption 1 holds and β^* is the global optimal solution to* (3.9)*, then the optimal policy solution to the KL trust-region constrained optimization problem in* (3.8) *is:*

$$\pi'^*(a|s) = \mathbb{F}(\pi) = \frac{e^{A^\pi(s,a)/\beta^*}\pi(a|s)}{\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta^*}]}, \quad \forall s \in \mathcal{S}, a \in \mathcal{A}. \tag{3.10}$$

Proof. Based on the proof of Theorem 3, the optimal solution $L_s^*(a)$ to (B.2) is: $L_s^*(a) = \frac{e^{A^\pi(s,a)/\beta^*}}{\mathbb{E}_{a\sim\pi(\cdot|s)}[e^{A^\pi(s,a)/\beta^*}]}$. Since $\pi'^*(a|s) = L_s^*(a)\pi(a|s)$, we have (3.10) holds. □

Appendix C

APPENDIX FOR CHAPTER 4

C.1 Implementation Details and Additional Results

The implementation of our WPO/SPO can be found in https://github.com/efficientwpo/EfficientWPO. We use the implementations of TRPO, PPO and A2C from OpenAI Baselines [67] for MuJuCo tasks and Stable Baselines [111] for other tasks. For BGPG, we adopt the same implementation as Pacchinao et al., (2020) based on the released code https://github.com/behaviorguidedRL/BGRL.

C.1.1 Visitation Frequencies Estimation:

The unnormalized discounted visitation frequencies are needed to compute the global optimal β^*. At the k-th iteration, the visitation frequencies ρ_k^π are estimated using samples of the trajectory set $\mathcal{D}_k$. Specifically, we first initialize $\rho_k^\pi(s) = 0,\ \forall s \in S$. Then for each timestep t in each trajectory from $\mathcal{D}_k$, we update ρ_k^π as $\rho_k^\pi(s_t) \leftarrow \rho_k^\pi(s_t) + \gamma^t/|\mathcal{D}_k|$.

C.1.2 Optimal-then-decay Beta Strategy:

During the training of multiple tasks, including Taxi, Chain and CartPole, we observe a consistent trend in the behavior of the optimal β value during the policy updates: It initially fluctuates, then stabilizes and decays slowly towards 0. In the Taxi task, the optimal β stabilizes after approximately 18% of the total training iterations. If we decay β before this stabilization point (e.g, using optimal beta for only first 5% or 10% updates), we observe a drop in performance. However, we do not observe any notable performance difference when we decay β after this stabilization point (e.g., using optimal β for first 20% or 30% updates). We also observe that the optimal β decays at a very slow rate, and $\Theta(1/\log(k))$ matches this trend best. If we employ a faster decaying function, such as $\Theta(1/k)$ or $\Theta(1/k^2)$, we observe a drop in performance.

Based on these findings, when implementing the optimal-then-decay β strategy on other tasks, we compute the optimal β for each policy update until we observe that its value stabilizes across updates. At this point, we stop calculating the optimal β and decay it using $\Theta(1/\log(k))$ for the remaining policy updates. The specific iteration at which the optimal β value stabilizes varies across tasks, and we denote this point as k_β, which is reported in Table C.1.

C.1.3 Hyperparameters and Performance Summary

Our main experimental results are reported in section 4.7. In addition, we provide the setting of hyperparameters and network sizes of our WPO/SPO algorithms in Table C.1, and a summary of performance in Table C.2.

Table C.1: Hyperparameters and network sizes

	Taxi-v3	NChain-v0 CliffWalking-v0	CartPole-v1	Acrobot-v1	MuJuCo tasks
γ	0.9	0.9	0.95	0.95	0.99
lr_π	\	\	10^{-2}	5×10^{-3}	10^{-4}
lr_{value}	10^{-2}	10^{-2}	10^{-2}	5×10^{-3}	10^{-3}
$\lvert\mathcal{D}_k\rvert$	60 (Taxi)	1 (Chain) 3 (CliffWalking)	2	3	partial
π size	2D array	2D array	$[64, 64]$	$[64, 64]$	$[400, 300]$
Q/v size	$[10, 7, 5]$	$[10, 7, 5]$	$[64, 64]$	$[64, 64]$	$[400, 300]$
$\lvert\mathcal{S}_k\rvert$	all states, $\lvert\mathcal{S}\rvert$	all states, $\lvert\mathcal{S}\rvert$	128	128	64
$\lvert\mathcal{A}_k\rvert$	all actions, $\lvert\mathcal{A}\rvert$	all actions, $\lvert\mathcal{A}\rvert$	all actions, $\lvert\mathcal{A}\rvert$	all actions, $\lvert\mathcal{A}\rvert$	32

$d(a, a')$	0-1 distance [1]	0-1 distance	0-1 distance	0-1 distance	L1 distance
k_β	250	100 (Chain) 50 (CliffWalking)	150	150	1000

Table C.2: Averaged rewards over last 10% episodes during the training process

Environment	WPO	SPO	TRPO	PPO	A2C	BGPG	WNPG
Taxi-v3	-45 ± 27	-87 ± 11	-202 ± 3	-381 ± 34	-338 ± 30	-	-
NChain-v0	3549 ± 197	3432 ± 131	3522 ± 258	3506 ± 237	1606 ± 10	-	-
CliffWalking-v0	-35 ± 15	-25 ± 1	-159 ± 94	-3290 ± 2106	-5587 ± 1942	-	-
CartPole-v1	388 ± 54	370 ± 30	297 ± 65	193 ± 45	267 ± 61	-	-
Acrobot-v1	-162 ± 8	-185 ± 15	-248 ± 33	-103 ± 5	-379 ± 39	-	-
HalfCheetah-v2	2050 ± 108	1750 ± 172	1158 ± 35	1628 ± 136	-645 ± 31	1697 ± 195	1832 ± 125
Hopper-v2	3208 ± 259	2834 ± 305	2035 ± 248	2321 ± 233	43 ± 21	1982 ± 218	2361 ± 272
Walker2d-v2	3739 ± 298	3489 ± 257	2535 ± 369	3290 ± 354	28 ± 1	2775 ± 301	3059 ± 209
Ant-v2	1863 ± 271	1780 ± 257	21 ± 10	1487 ± 206	-39 ± 8	1622 ± 235	1587 ± 221
Humanoid-v2	965 ± 76	914 ± 93	725 ± 112	632 ± 73	107 ± 15	797 ± 85	820 ± 91

[1] We note that specifying distance based on control relevance leads to higher performance in this test case: i.e., $d = 1$ to distinct actions from set $A = \{$ move north, move south, move west, move east$\}$, $d = 1$ to distinct actions from set $B = \{$ pickup, dropoff$\}$, and $d = 4$ to actions from different sets.

C.1.4 Additional Results for Ablation Studies

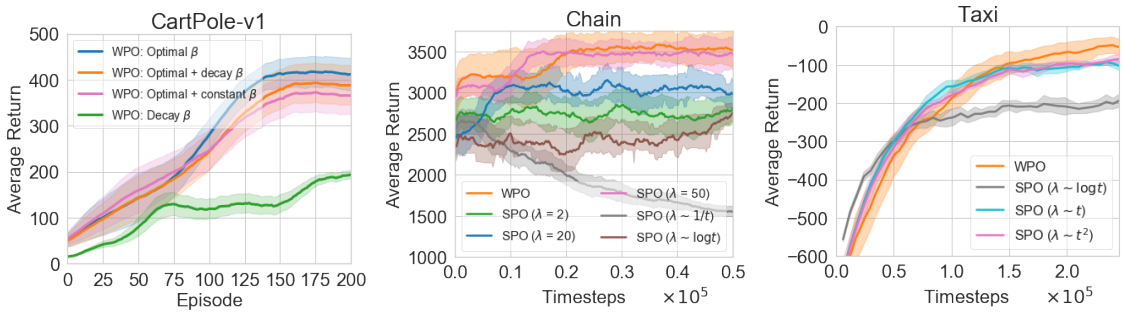
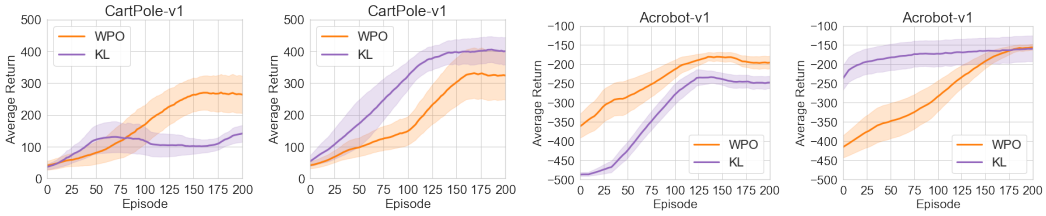


Figure C.1: Episode rewards during the training process for different β and λ settings, averaged across 5 runs with a random initialization. The shaded area depicts the mean $\pm$ the standard deviation.

C.1.5 Additional Comparison of Wasserstein and KL Trust Regions



(a) $N_A = 100$ (b) $N_A = 500$ (c) $N_A = 100$ (d) $N_A = 500$

Figure C.2: Episode rewards during the training process for the locomotion tasks, averaged across 5 runs with a random initialization. The shaded area depicts the mean $\pm$ the standard deviation.

Table C.3: Average runtime (seconds) of WPO, SPO and KL

	WPO	SPO	KL
Taxi-v3 (per 10^3 steps)	71.0 ± 7.3	69.5 ± 8.7	74.3 ± 9.5
NChain-v0 (per 10^3 steps)	58.4 ± 9.1	63.1 ± 7.4	59.9 ± 8.7
CartPole-v1 (per 10^6 steps)	11.4 ± 1.8	10.2 ± 2.3	9.7 ± 1.9
Acrobot-v1 (per 10^5 steps)	10.4 ± 1.9	9.7 ± 2.5	10.9 ± 2.3
Humanoid-v2 (per 10^5 steps)	422.7 ± 65.4	409.1 ± 46.5	438.5 ± 61.2

C.1.6 Additional Results for Large-scale Continuous Control

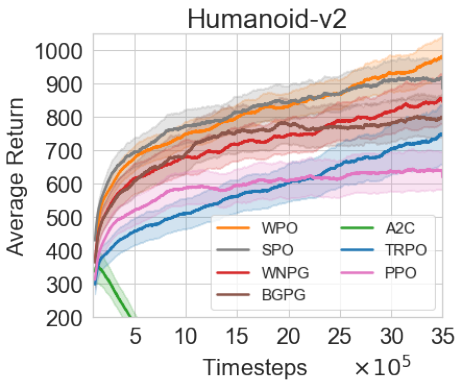


Figure C.3: Episode rewards during training for MuJuCo Humanoid task, averaged across 10 runs with random initialization. The shaded area depicts the mean ± the standard deviation.

Table C.4: Average runtime (seconds per 10^5 timesteps) for the MuJuCo continuous control tasks

Environment	WPO	SPO	TRPO	PPO	A2C	BGPG	WNPG
HalfCheetah-v2	297 ± 31	289 ± 25	290 ± 28	292 ± 36	293 ± 27	306 ± 33	298 ± 22
Hopper-v2	233 ± 38	226 ± 42	242 ± 56	167 ± 36	254 ± 49	201 ± 32	197 ± 31
Walker2d-v2	289 ± 55	312 ± 61	253 ± 39	307 ± 52	259 ± 46	322 ± 62	214 ± 45
Ant-v2	307 ± 51	290 ± 57	296 ± 63	251 ± 47	291 ± 41	286 ± 63	269 ± 54
Humanoid-v2	423 ± 65	401 ± 47	446 ± 52	395 ± 57	230 ± 31	425 ± 58	398 ± 49

C.2 Proof of Theorem 5

Theorem 5. ***(Closed-form policy update)*** *Let $\kappa_s^\pi(\beta, j) = argmax_{k=1\ldots N}\{A^\pi(s, a_k) - \beta D_{kj}\}$, where D denotes the cost matrix. If Assumption 1 holds, then an optimal solution to (4.4) is:*

$$\pi^*(a_i|s) = \sum\nolimits_{j=1}^{N} \pi(a_j|s) f_s^*(i, j), \tag{4.5}$$

where $f_s^*(i, j) = 1$ if $i = \kappa_s^\pi(\beta^*, j)$ and $f_s^*(i, j) = 0$ otherwise, and β^* is an optimal Lagrangian multiplier corresponds to the following dual formulation:

$$\min_{\beta \geq 0} F(\beta) = \min_{\beta \geq 0} \{\beta\delta + \mathbb{E}_{s\sim\rho_v^\pi} \sum\nolimits_{j=1}^{N} \pi(a_j|s) \max_{i=1\ldots N} (A^\pi(s, a_i) - \beta D_{ij})\}. \tag{4.6}$$

Moreover, we have $\beta^* \leq \bar{\beta}$, where $\bar{\beta} := \max_{s\in\mathcal{S}, k, j=1\ldots N, k\neq j} (D_{kj})^{-1}(A^\pi(s, a_k) - A^\pi(s, a_j))$.

Proof of Theorem 5. First, we denote Q^s as the joint distribution of $\pi(\cdot|s)$ and $\pi'(\cdot|s)$ with $\sum_{i=1}^N Q_{ij}^s = \pi(a_j|s)$ and $\sum_{j=1}^N Q_{ij}^s = \pi'(a_i|s)$. Also, let $f_s(i, j)$ represent the conditional distribution of $\pi'(a_i|s)$ under $\pi(a_j|s)$. Then $Q_{ij}^s = \pi(a_j|s) f_s(i, j)$, $\pi'(a_i|s) = \sum_{j=1}^N Q_{ij}^s = \sum_{j=1}^N \pi(a_j|s) f_s(i, j)$. In addition:

$$\begin{aligned} d_{\mathrm{W}}(\pi'(\cdot|s), \pi(\cdot|s)) &= \min_{Q_{ij}^s} \sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij} Q_{ij}^s = \min_{f_s(i,j)} \sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}\pi(a_j|s) f_s(i, j), \text{ and} \\ \mathbb{E}_{a\sim\pi'(\cdot|s)}[A^\pi(s, a)] &= \sum_{i=1}^{N} A^\pi(s, a_i)\pi'(a_i|s) = \sum_{i=1}^{N}\sum_{j=1}^{N} A^\pi(s, a_i)\pi(a_j|s) f_s(i, j). \end{aligned}$$

Thus, the WPO problem in (3.8) can be reformulated as:

$$\max_{f_s(i,j)\geq 0} \quad \mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^{N}\sum_{j=1}^{N} A^\pi(s,a_i)\pi(a_j|s)f_s(i,j) \tag{C.1a}$$

$$s.t. \quad \mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}\pi(a_j|s)f_s(i,j) \leq \delta, \tag{C.1b}$$

$$\sum_{i=1}^{N} f_s(i,j) = 1, \qquad \forall s\in\mathcal{S}, j = 1\dots N. \tag{C.1c}$$

Note here that (C.1b) is equivalent to $\mathbb{E}_{s\sim\rho_v^\pi} \min_{f_s(i,j)} \sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}\pi(a_j|s)f_s(i,j) \leq \delta$ because if we have a feasible $f_s(i,j)$ to make (C.1b) hold, we must have $\mathbb{E}_{s\sim\rho_v^\pi} \min_{f_s(i,j)} \sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}\pi(a_j|s)f_s(i,j) \leq \delta$.

Since both the objective function and the constraint are linear in $f_s(i,j)$, (C.1) is a convex optimization problem. Also, Slater's condition holds for (C.1) as the feasible region has an interior point, which is $f_s(i,i) = 1\ \forall i$, and $f_s(i,j) = 0\ \forall i \neq j$. Meanwhile, since $A^\pi(s,a)$ is bounded based on Assumption 1, the objective is bounded above. Therefore, strong duality holds for (C.1). At this point we can derive the dual problem of (C.1) as its equivalent reformulation:

$$\begin{aligned} &\min_{\beta\geq 0,\zeta_j^s} \quad \beta\delta + \int_{s\in\mathcal{S}} \sum_{j=1}^{N} \zeta_j^s ds \\ &s.t. \quad A^\pi(s,a_i)\pi(a_j|s) - \beta D_{ij}\pi(a_j|s) - \frac{\zeta_j^s}{\rho_v^\pi(s)} \leq 0, \qquad \forall s\in\mathcal{S}, i,j = 1\dots N. \end{aligned} \tag{C.2}$$

We observe that with a fixed β, the optimal ζ_j^s will be achieved at:

$$\zeta_j^{s*}(\beta) = \max_{i=1\dots N} \rho_v^\pi(s)\pi(a_j|s)(A^\pi(s,a_i) - \beta D_{ij}). \tag{C.3}$$

Denote β^* as an optimal solution to (C.2) and $f_s^*(i,j)$ as an optimal solution to (C.1). Due to the complimentary slackness, the following equations hold:

$$(A^\pi(s,a_i)\pi(a_j|s) - \beta^* D_{ij}\pi(a_j|s) - \frac{\zeta_j^{s*}(\beta^*)}{\rho_v^\pi(s)}) f_s^*(i,j) = 0, \qquad \forall s,i,j.$$

In this case, $f_s^*(i,j)$ can have non-zero values only when $A^\pi(s,a_i)\pi(a_j|s) - \beta^* D_{ij}\pi(a_j|s) - \frac{\zeta_j^{s*}(\beta^*)}{\rho_v^\pi(s)} = 0$, which means $\zeta_j^{s*}(\beta^*) = \rho_v^\pi(s)\pi(a_j|s)(A^\pi(s,a_i) - \beta^* D_{ij})$. Given the expression of

the optimal ζ_j^{s*} in (C.3), $f_s^*(i,j)$ can have non-zero values only when $i \in \mathcal{K}_s^\pi(\beta^*, j)$, where $\mathcal{K}_s^\pi(\beta, j) = \text{argmax}_{k=1\ldots N} A^\pi(s, a_k) - \beta D_{kj}$.

When there exists a unique optimizer, i.e., $|\mathcal{K}_s^\pi(\beta^*, j)| = 1$, let $\kappa_s^\pi(\beta^*, j)$ denote the optimizer. Since $\sum_{i=1}^N f_s^*(i,j) = 1$ as indicated in (C.1c), the only optimal solution is:

$$f_s^*(i,j) = \begin{cases} 1 & \text{if } i = \kappa_s^\pi(\beta^*, j), \\ 0 & \text{otherwise.} \end{cases}$$

When there exists multiple optimizers, i.e., $|\mathcal{K}_s^\pi(\beta^*, j)| > 1$, the optimal weights $f_s^*(i,j)$ for $i \in \mathcal{K}_s^\pi(\beta^*, j)$ could be determined by solving the following linear programming:

$$\begin{aligned} \max_{f_s^*(i,j)\geq 0, i\in\mathcal{K}_s^\pi(\beta^*,j)} \quad & \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^N \pi(a_j|s) \sum_{i\in\mathcal{K}_s^\pi(\beta^*,j)} A^\pi(s,a_i) f_s^*(i,j) \\ s.t. \quad & \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^N \pi(a_j|s) \sum_{i\in\mathcal{K}_s^\pi(\beta^*,j)} D_{ij} f_s^*(i,j) \leq \delta, \\ & \sum_{i\in\mathcal{K}_s^\pi(\beta^*,j)} f_s^*(i,j) = 1, \qquad \forall s \in \mathcal{S}, j = 1\ldots N. \end{aligned} \tag{C.4}$$

And then the corresponding optimal solution is, $\pi^*(a_i|s) = \sum_{j=1}^N \pi(a_j|s) f_s^*(i,j)$.

Last, by substituting $\zeta_j^{s*}(\beta) = \rho_v^\pi(s)\pi(a_j|s)\max_{i=1\ldots N}(A^\pi(s,a_i) - \beta D_{ij})$ into the dual problem (C.2), we can reformulate (C.2) into:

$$\min_{\beta\geq 0}\{\beta\delta + \int_{s\in\mathcal{S}} \sum_{j=1}^N \zeta_j^{s*}(\beta) ds\} = \min_{\beta\geq 0}\{\beta\delta + \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^N \pi(a_j|s) \max_{i=1\ldots N}(A^\pi(s,a_i) - \beta D_{ij})\}. \tag{C.5}$$

The optimal β can then be obtained by solving (C.5).

We will further show that $\beta^* \leq \bar{\beta} := \max_{s\in\mathcal{S}, k, j=1\ldots N, k\neq j} (D_{kj})^{-1}(A^\pi(s,a_k) - A^\pi(s,a_j))$.

In the general case, i.e., $\beta \geq 0$, (C.1a) is non-negative because:

$$\mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^N \sum_{j=1}^N A^\pi(s,a_i)\pi(a_j|s) f_s^*(i,j) \tag{C.6a}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^N \pi(a_j|s) \sum_{i=1}^N A^\pi(s,a_i) f_s^*(i,j) \tag{C.6b}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \sum_{i\in\mathcal{K}_s^\pi(\beta^*,j)} f_s^*(i,j) A^\pi(s,a_i) \tag{C.6c}$$

$$\geq \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \sum_{i\in\mathcal{K}_s^\pi(\beta^*,j)} f_s^*(i,j)[A^\pi(s,a_j) + \beta^* D_{ij}] \tag{C.6d}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) A^\pi(s,a_j) + \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \sum_{i\in\mathcal{K}_s^\pi(\beta^*,j)} f_s^*(i,j)\beta^* D_{ij} \tag{C.6e}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s)\beta^* \sum_{i\in\mathcal{K}_s^\pi(\beta^*,j)} f_s^*(i,j) D_{ij} \tag{C.6f}$$

$$\geq 0, \tag{C.6g}$$

where (C.6d) holds since for $i \in \mathcal{K}_s^\pi(\beta^*, j)$, $A^\pi(s,a_i) - \beta^* D_{ij} \geq A^\pi(s,a_j) - \beta^* D_{jj} = A^\pi(s,a_j)$. When $\beta^* > \max_{s\in\mathcal{S},k,j=1\ldots N,k\neq j}\{\frac{A^\pi(s,a_k)-A^\pi(s,a_j)}{D_{kj}}\}$, we have that for all $s \in \mathcal{S}$, $\kappa_s^\pi(\beta^*, j) = j$. Thus, $f_s^*(i,i) = 1$, $\forall i$ and $f_s^*(i,j) = 0$, $\forall i \neq j$. The objective value (C.1a) will be 0 because $\mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^N \sum_{j=1}^N A^\pi(s,a_i)\pi(a_j|s) f_s^*(i,j) = \mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^N A^\pi(s,a_i)\pi(a_i|s) = 0$. The left hand side of (C.1b) equals to $\mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^N \sum_{j=1}^N D_{ij}\pi(a_j|s) f_s^*(i,j) = \mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^N D_{ii}\pi(a_i|s) = 0$. Thus, for any $\delta > 0$, (C.1b) is always satisfied.

Since the objective of the primal Wasserstein trust-region constrained problem in (4.6) constantly evaluates to 0 when $\beta^* > \max_{s\in\mathcal{S},k,j=1\ldots N,k\neq j}\{\frac{A^\pi(s,a_k)-A^\pi(s,a_j)}{D_{kj}}\}$, and is non-negative when $\beta^* \leq \max_{s\in\mathcal{S},k,j=1\ldots N,k\neq j}\{\frac{A^\pi(s,a_k)-A^\pi(s,a_j)}{D_{kj}}\}$, we can use $\max_{s\in\mathcal{S},k,j=1\ldots N,k\neq j}\{\frac{A^\pi(s,a_k)-A^\pi(s,a_j)}{D_{kj}}\}$ as an upper bound for the optimal dual variable β^*.

□

C.3 Optimal Beta for a Special Distance

Proposition 1. *Let $k_s = argmax_{i=1,\ldots,N} A^\pi(s,a_i)$, we have:*

(1). If the initial point β_0 is in $[\max_{s,j}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}, +\infty)$, the local optimal β solution is $\max_{s,j}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$.

(2). If the initial point β_0 is in $[0, \min_{s,j\neq k_s}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}]$: if $\delta - \int_{s\in\mathcal{S}} \rho^\pi(s)(1 - \pi(a_{k_s}|s))ds < 0$, the local optimal β is $\min_{s,j\neq k_s}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$; otherwise, the local optimal β solution is 0.

(3). If the initial point β_0 is in $(\min_{s,j\neq k_s}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}, \max_{s,j}\{A^\pi(s,a_{k_s}) -$

$A^\pi(s,a_j)\}$), *we construct sets* I_s^1 *and* I_s^2 *as:*

for $s \in \mathcal{S}, j \in \{1,2\dots N\}$: **if** $\beta_0 \geq A^\pi(s,a_{k_s}) - A^\pi(s,a_j)$ **then** Add j to I_s^1 **else** Add j to I_s^2. *Then, if* $\delta - \mathbb{E}_{s\sim\rho^\pi}\sum_{j\in I_s^2}\pi(a_j|s) < 0$, *the local optimal* β *is* $\min_{s\in\mathcal{S},j\in I_s^2}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$*; otherwise, the local optimal* β *is* $\max_{s\in\mathcal{S},j\in I_s^1}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$.

Proof of Proposition 1. (1). When $\beta \in [\max_{s,j}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}, +\infty)$, we have $A^\pi(s,a_j) \geq A^\pi(s,a_{k_s}) - \beta$ for all $s \in \mathcal{S}$, $j = 1\dots N$. Since $A^\pi(s,a_{k_s}) - \beta \geq A^\pi(s,a_k) - \beta$ for all $k = 1\dots N$, we have $A^\pi(s,a_j) \geq A^\pi(s,a_k) - \beta$ for all $s \in \mathcal{S}$, $j = 1\dots N$, $k = 1\dots N$. Thus, $j \in \text{argmax}_{k=1\dots N}\{A^\pi(s,a_k) - \beta D_{kj}\}$, for all $s \in \mathcal{S}$, $j = 1\dots N$. Therefore, (4.6) can be reformulated as:

$$\min_{\beta\geq 0}\{\beta\delta + \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^N \pi(a_j|s)A^\pi(s,a_j)\}.$$

Since $\delta \geq 0$, we have the local optimal $\beta = \max_{s,j}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$.

(2). When $\beta \in [0, \min_{s,j\neq k_s}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}]$, we have $A^\pi(s,a_j) \leq A^\pi(s,a_{k_s}) - \beta$ for all $s \in \mathcal{S}$, $j = 1\dots N$, $j \neq k_s$. Thus $k_s \in \text{argmax}_{k=1\dots N}\{A^\pi(s,a_k) - \beta D_{kj}\}$ for all $s \in \mathcal{S}$, $j = 1\dots N$. The inner part of (4.6) then is:

$$\begin{aligned}
&\beta\delta + \mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j=1,j\neq k_s}^N \pi(a_j|s)(A^\pi(s,a_{k_s}) - \beta) + \pi(a_{k_s}|s)A^\pi(s,a_{k_s})\} \\
&= \beta(\delta - \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1,j\neq k_s}^N \pi(a_j|s)) + \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^N \pi(a_j|s)A^\pi(s,a_{k_s}) \\
&= \beta(\delta - \int_{s\in\mathcal{S}} \rho_v^\pi(s)(1-\pi(a_{k_s}|s))ds) + \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^N \pi(a_j|s)A^\pi(s,a_{k_s}).
\end{aligned}$$

If $\delta - \int_{s\in\mathcal{S}} \rho_v^\pi(s)(1-\pi(a_{k_s}|s))ds < 0$, we have the local optimal $\beta = \min_{s,j\neq k_s}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$. If $\delta - \int_{s\in\mathcal{S}} \rho_v^\pi(s)(1-\pi(a_{k_s}|s))ds \geq 0$, we have the local optimal $\beta = 0$.

(3). For an initial point β_0 in $(\min_{s,j\neq k_s}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}, \max_{s,j}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\})$, we construct partitions I_s^1 and I_s^2 of the set $\{1,2\dots N\}$ in the way described in Proposition 1 for all $s \in \mathcal{S}$. Consider β in the neighborhood of β_0, i.e., $\beta \geq A^\pi(s,a_{k_s}) - A^\pi(s,a_j)$ for $s \in \mathcal{S}, j \in I_s^1$ and $\beta \leq A^\pi(s,a_{k_s}) - A^\pi(s,a_j)$ for $s \in \mathcal{S}, j \in I_s^2$. Then the inner

part of (4.6) can be reformulated as:

$$\beta\delta + \mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j\in I_s^1}\pi(a_j|s)A^\pi(s,a_j) + \sum_{j\in I_s^2}\pi(a_j|s)(A^\pi(s,a_{k_s})-\beta)\}$$
$$= \beta(\delta - \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j\in I_s^2}\pi(a_j|s)) + \mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j\in I_s^1}\pi(a_j|s)A^\pi(s,a_j) + \sum_{j\in I_s^2}\pi(a_j|s)A^\pi(s,a_{k_s})\}.$$

If $\delta - \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j\in I_s^2}\pi(a_j|s) < 0$, we have the local optimal $\beta = \min_{s\in\mathcal{S},j\in I_s^2}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$. If $\delta - \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j\in I_s^2}\pi(a_j|s) \geq 0$, we have the local optimal $\beta = \max_{s\in\mathcal{S},j\in I_s^1}\{A^\pi(s,a_{k_s}) - A^\pi(s,a_j)\}$. □

C.4 Proof of Theorem 6

Theorem 6. *If Assumption 1 holds, then the optimal solution to (4.4) with Sinkhorn divergence is:*

$$\pi_\lambda^*(a_i|s) = \sum_{j=1}^N \pi(a_j|s) f_{s,\lambda}^*(i,j), \tag{4.8}$$

where D denotes the cost matrix, $f_{s,\lambda}^*(i,j) = \frac{\exp(\frac{\lambda}{\beta_\lambda^*}A^\pi(s,a_i)-\lambda D_{ij})}{\sum_{k=1}^N \exp(\frac{\lambda}{\beta_\lambda^*}A^\pi(s,a_k)-\lambda D_{kj})}$ and β_λ^* is an optimal solution to the following dual formulation:

$$\min_{\beta\geq 0} F_\lambda(\beta) = \min_{\beta\geq 0}\Big\{\beta\delta - \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^N \pi(a_j|s)(\frac{\beta}{\lambda} + \frac{\beta}{\lambda}\ln(\pi(a_j|s)) - \frac{\beta}{\lambda}\ln[\sum_{i=1}^N \exp(\frac{\lambda}{\beta}A^\pi(s,a_i) - \lambda D_{ij})])$$
$$\mathbb{E}_{s\sim\rho_v^\pi}\sum_{i=1}^N\sum_{j=1}^N \frac{\beta}{\lambda}\frac{\exp(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij})\cdot\pi(a_j|s)}{\sum_{k=1}^N \exp(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj})}\Big\}. \tag{4.9}$$

Moreover, we have $\beta_\lambda^* \leq \frac{2A^{max}}{\delta}$.

Proof of Theorem 6. Invoking the definition of Sinkhorn divergence in (4.3), the trust region constrained problem with Sinkhorn divergence can be reformulated as:

$$\max_Q \quad \mathbb{E}_{s\sim\rho_v^\pi}[\sum\nolimits_{i=1}^N A^\pi(s,a_i)\sum\nolimits_{j=1}^N Q_{ij}^s] \tag{C.7a}$$
$$s.t. \quad \mathbb{E}_{s\sim\rho_v^\pi}[\sum\nolimits_{i=1}^N\sum\nolimits_{j=1}^N D_{ij}Q_{ij}^s + \frac{1}{\lambda}Q_{ij}^s\log Q_{ij}^s] \leq \delta \tag{C.7b}$$
$$\sum\nolimits_{i=1}^N Q_{ij}^s = \pi(a_j|s), \quad \forall j = 1,\ldots,N, s\in\mathcal{S}. \tag{C.7c}$$

Let β and ω represent the dual variables of constraints (C.7b) and (C.7c) respectively, then the Lagrangian duality of (C.7) can be derived as:

$$\max_{Q}\min_{\beta\geq 0,\omega} L(Q,\beta,\omega) = \max_{Q}\min_{\beta\geq 0,\omega} \mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N} A^\pi(s,a_i)\sum_{j=1}^{N} Q_{ij}^s] + \int_{s\in\mathcal{S}}\sum_{j=1}^{N}\omega_j^s(\sum_{i=1}^{N} Q_{ij}^s - \pi(a_j|s))ds + \beta(\delta - \mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}Q_{ij}^s + \frac{1}{\lambda}Q_{ij}^s\log Q_{ij}^s]) \tag{C.8a}$$

$$= \max_{Q}\min_{\beta\geq 0,\omega} \mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N} A^\pi(s,a_i)\sum_{j=1}^{N} Q_{ij}^s] + \int_{s\in\mathcal{S}}\sum_{j=1}^{N}\sum_{i=1}^{N}\frac{\omega_j^s}{\rho_v^\pi(s)}Q_{ij}^s\rho_v^\pi(s)ds - \int_{s\in\mathcal{S}}\sum_{j=1}^{N}\omega_j^s\pi(a_j|s)ds + \beta\delta - \beta\mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}Q_{ij}^s + \frac{1}{\lambda}Q_{ij}^s\log Q_{ij}^s]) \tag{C.8b}$$

$$= \max_{Q}\min_{\beta\geq 0,\omega} \beta\delta - \int_{s\in\mathcal{S}}\sum_{j=1}^{N}\omega_j^s\pi(a_j|s)ds + \mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N}\sum_{j=1}^{N}(A^\pi(s,a_i) - \beta D_{ij} + \frac{\omega_j^s}{\rho_v^\pi(s)})Q_{ij}^s - \frac{\beta}{\lambda}Q_{ij}^s\log Q_{ij}^s] \tag{C.8c}$$

$$= \min_{\beta\geq 0,\omega}\max_{Q} \beta\delta - \int_{s\in\mathcal{S}}\sum_{j=1}^{N}\omega_j^s\pi(a_j|s)ds + \mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N}\sum_{j=1}^{N}(A^\pi(s,a_i) - \beta D_{ij} + \frac{\omega_j^s}{\rho_v^\pi(s)})Q_{ij}^s - \frac{\beta}{\lambda}Q_{ij}^s\log Q_{ij}^s], \tag{C.8d}$$

where (C.8d) holds since the Lagrangian function $L(Q,\beta,\omega)$ is concave in Q and linear in β and ω, and we can exchange the max and the min following the Minimax theorem [230].

Note that the inner max problem of (C.8d) is an unconstrained concave problem, and we can obtain the optimal Q by taking the derivatives and setting them to 0. That is,

$$\frac{\partial L}{\partial Q_{ij}^s} = A^\pi(s,a_i) - \beta D_{ij} + \frac{\omega_j^s}{\rho_v^\pi(s)} - \frac{\beta}{\lambda}(\log Q_{ij}^s + 1) = 0, \quad \forall i,j = 1,\cdots,N, s\in\mathcal{S}. \tag{C.9}$$

Therefore, we have the optimal Q_{ij}^{s*} as:

$$Q_{ij}^{s*} = \exp(\frac{\lambda}{\beta}A^\pi(s,a_i) - \lambda D_{ij})\exp(\frac{\lambda\omega_j^s}{\beta\rho_v^\pi(s)} - 1), \quad \forall i,j = 1,\cdots,N, s\in\mathcal{S}. \tag{C.10}$$

In addition, since $\sum_{i=1}^{N} Q_{ij}^{s*} = \pi(a_j|s)$, we have the following hold:

$$\exp(\frac{\lambda\omega_j^s}{\beta\rho_v^\pi(s)} - 1) = \frac{\pi(a_j|s)}{\sum_{i=1}^{N}\exp(\frac{\lambda}{\beta}A^\pi(s,a_i) - \lambda D_{ij})}. \tag{C.11}$$

By substituting the left hand side of (C.11) into (C.10), we can further reformulate the optimal Q_{ij}^{s*} as:

$$Q_{ij}^{s*} = \frac{\exp\left(\frac{\lambda}{\beta}A^{\pi}(s,a_i) - \lambda D_{ij}\right)}{\sum_{k=1}^{N}\exp\left(\frac{\lambda}{\beta}A^{\pi}(s,a_k) - \lambda D_{kj}\right)}\pi(a_j|s), \quad \forall i,j = 1,\cdots,N, s \in \mathcal{S}. \tag{C.12}$$

To obtain the optimal dual variables, based on (C.11), we have the optimal ω^* as:

$$\omega_j^{s*} = \rho_v^{\pi}(s)\{\frac{\beta}{\lambda} + \frac{\beta}{\lambda}\ln(\pi(a_j|s)) - \frac{\beta}{\lambda}\ln[\sum_{i=1}^{N}\exp\left(\frac{\lambda}{\beta}A^{\pi}(s,a_i) - \lambda D_{ij}\right)]\}, \quad \forall j = 1,\cdots,N, s \in \mathcal{S} \tag{C.13}$$

By substituting (C.12) and (C.13) into (C.8d), we can obtain the optimal β^* via:

$$\begin{aligned}\min_{\beta\geq 0} \quad & \beta\delta - \mathbb{E}_{s\sim\rho_v^{\pi}}\sum_{j=1}^{N}\pi(a_j|s)\{\frac{\beta}{\lambda} + \frac{\beta}{\lambda}\ln(\pi(a_j|s)) - \frac{\beta}{\lambda}\ln[\sum_{i=1}^{N}\exp\left(\frac{\lambda}{\beta}A^{\pi}(s,a_i) - \lambda D_{ij}\right)]\} \\ & + \mathbb{E}_{s\sim\rho_v^{\pi}}\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\beta}{\lambda}\frac{\exp\left(\frac{\lambda}{\beta}A^{\pi}(s,a_i) - \lambda D_{ij}\right)\cdot\pi(a_j|s)}{\sum_{k=1}^{N}\exp\left(\frac{\lambda}{\beta}A^{\pi}(s,a_k) - \lambda D_{kj}\right)}.\end{aligned}$$

The proof for the upper bound of sinkhorn optimal β can be found in Appendix C.5. □

C.5 Upper bound of Sinkhorn Optimal Beta

In this section, we will derive the upper bound of Sinkhorn optimal β. First, for a given β, the optimal $Q_{ij}^{s*}(\beta)$ to the Lagrangian dual $L(Q,\beta,\omega)$ can be expressed in (C.12). With this, we will present the following two lemmas:

Lemma 1. *The objective function (C.7a) with respect to $Q_{ij}^{s*}(\beta)$ decreases as the dual variable β increases.*

Lemma 2. *If Assumption 1 holds, then for every $\delta > 0$, $Q_{ij}^{s*}(\frac{2A^{max}}{\delta})$ is feasible to (C.7b) for any λ.*

We provide proofs for Lemma 1 and Lemma 2 in Appendix C.5.1 and Appendix C.5.2 respectively. Given the above two lemmas, we are able to prove the following proposition on the upper bound of Sinkhorn optimal β:

Proposition 2. *If β_λ^* is the optimal dual solution to the Sinkhorn dual formulation (4.9), then $\beta_\lambda^* \leq \frac{2A^{max}}{\delta}$ for any λ.*

Proof of Proposition 2. We will prove it by contradiction. According to Lemma 2, $Q_{ij}^{s*}(\frac{2A^{\max}}{\delta})$ is feasible to (C.7b). Since β_λ^* is the optimal dual solution, $Q_{ij}^{s*}(\beta_\lambda^*)$ is optimal to (C.7). If $\beta_\lambda^* > \frac{2A^{\max}}{\delta}$, according to Lemma 1, the objective value in (C.7a) with respect to $\frac{2A^{\max}}{\delta}$ is smaller than the objective value in (C.7a) with respect to β_λ^*, which contradicts the fact that $Q_{ij}^{s*}(\beta_\lambda^*)$ is the optimal solution to (C.7). □

C.5.1 Proof of Lemma 1

Lemma 1. *The objective function (C.7a) with respect to $Q_{ij}^{s*}(\beta)$ decreases as the dual variable β increases.*

Proof of Lemma 1. Let $G_\lambda(\beta)$ represent the objective function (C.7a). By substituting the optimal Q_{ij}^{s*} in (C.12) into (C.7a), we have:

$$G_\lambda(\beta) = \mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N} A^\pi(s,a_i) \sum_{j=1}^{N} \frac{\exp(\frac{\lambda}{\beta}A^\pi(s,a_i) - \lambda D_{ij})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta}A^\pi(s,a_k) - \lambda D_{kj})}\pi(a_j|s)] \tag{C.14a}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi}[\sum_{j=1}^{N} \pi(a_j|s) \sum_{i=1}^{N} A^\pi(s,a_i) \frac{\exp(\frac{\lambda}{\beta}A^\pi(s,a_i) - \lambda D_{ij})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta}A^\pi(s,a_k) - \lambda D_{kj})}]. \tag{C.14b}$$

For any $\beta_2 > \beta_1 > 0$, we have:

$$\begin{aligned}
&G_\lambda(\beta_1) - G_\lambda(\beta_2) \\
&= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \sum_{i=1}^{N} A^\pi(s,a_i)\{\frac{\exp(\frac{\lambda}{\beta_1}A^\pi(s,a_i) - \lambda D_{ij})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta_1}A^\pi(s,a_k) - \lambda D_{kj})} \\
&- \frac{\exp(\frac{\lambda}{\beta_2}A^\pi(s,a_i) - \lambda D_{ij})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta_2}A^\pi(s,a_k) - \lambda D_{kj})}\}
\end{aligned} \tag{C.15a}$$

$$\begin{aligned}
&= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \sum_{i=1}^{N} A^\pi(s,a_{[i]})\{\frac{\exp(\frac{\lambda}{\beta_1}A^\pi(s,a_{[i]}) - \lambda D_{[i]j})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta_1}A^\pi(s,a_{[k]}) - \lambda D_{[k]j})} \\
&- \frac{\exp(\frac{\lambda}{\beta_2}A^\pi(s,a_{[i]}) - \lambda D_{[i]j})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta_2}A^\pi(s,a_{[k]}) - \lambda D_{[k]j})}\},
\end{aligned} \tag{C.15b}$$

where $[i]$ denotes sorted indices that satisfy $A^\pi(s,a_{[1]}) \geq A^\pi(s,a_{[2]}) \geq \cdots \geq A^\pi(s,a_{[N]})$. Let

$$f_s(i) = \frac{\exp(\frac{\lambda}{\beta_1}A^\pi(s,a_{[i]}) - \lambda D_{[i]j})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta_1}A^\pi(s,a_{[k]}) - \lambda D_{[k]j})} - \frac{\exp(\frac{\lambda}{\beta_2}A^\pi(s,a_{[i]}) - \lambda D_{[i]j})}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta_2}A^\pi(s,a_{[k]}) - \lambda D_{[k]j})} \tag{C.16a}$$

$$= \frac{\exp\left((\frac{\lambda}{\beta_1} - \frac{\lambda}{\beta_2})A^\pi(s, a_{[i]})\right) \exp\left(\frac{\lambda}{\beta_2} A^\pi(s, a_{[i]}) - \lambda D_{[i]j}\right)}{\sum_{k=1}^N \exp\left((\frac{\lambda}{\beta_1} - \frac{\lambda}{\beta_2})A^\pi(s, a_{[k]})\right) \exp\left(\frac{\lambda}{\beta_2} A^\pi(s, a_{[k]}) - \lambda D_{[k]j}\right)}$$
$$- \frac{\exp\left(\frac{\lambda}{\beta_2} A^\pi(s, a_{[i]}) - \lambda D_{[i]j}\right)}{\sum_{k=1}^N \exp\left(\frac{\lambda}{\beta_2} A^\pi(s, a_{[k]}) - \lambda D_{[k]j}\right)}. \tag{C.16b}$$

For notation brevity, we let $m_s(i) = \exp\left((\frac{\lambda}{\beta_1} - \frac{\lambda}{\beta_2})A^\pi(s, a_{[i]})\right) > 0$, $w_s(i) = \exp\left(\frac{\lambda}{\beta_2} A^\pi(s, a_{[i]}) - \lambda D_{[i]j}\right) > 0$ and $q_s(i) = \frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} - \frac{1}{\sum_{k=1}^N m_s(i) w_s(k)}$. Then we have

$$(C.16b) = \frac{m_s(i) w_s(i)}{\sum_{k=1}^N m_s(k) w_s(k)} - \frac{w_s(i)}{\sum_{k=1}^N w_s(k)} \tag{C.17a}$$
$$= m_s(i) w_s(i) \left(\frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} - \frac{1}{\sum_{k=1}^N m_s(i) w_s(k)}\right) \tag{C.17b}$$
$$= m_s(i) w_s(i) q_s(i). \tag{C.17c}$$

Since $\frac{\lambda}{\beta_1} - \frac{\lambda}{\beta_2} > 0$, $m_s(i)$ decreases as i increases. Thus, $q_s(i)$ decreases as i increases. Since $m_s(1) \geq m_s(k)$ and $m_s(N) \leq m_s(k)$ for all $k = 1, \ldots, N$, we have $q_s(1) = \frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} - \frac{1}{\sum_{k=1}^N m_s(1) w_s(k)} \geq \frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} - \frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} = 0$, and $q_s(N) = \frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} - \frac{1}{\sum_{k=1}^N m_s(N) w_s(k)} \leq \frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} - \frac{1}{\sum_{k=1}^N m_s(k) w_s(k)} = 0$. Since $q_s(1) \geq 0$, $q_s(N) \leq 0$ and $q_s(i)$ decreases as i increases, there exists an index $1 \leq k_s \leq N$ such that $q_s(i) \geq 0$ for $i \leq k_s$ and $q_s(i) < 0$ for $i > k_s$. Since $m_s(i), w_s(i) > 0$, we have $f_s(i) \geq 0$ for $i \leq k_s$ and $f_s(i) < 0$ for $i > k_s$. In addition, we have $\sum_{i=1}^N f_s(i) = 0$ directly follows from the definition. Thus, $\sum_{i=1}^N f_s(i) = \sum_{i=1}^{k_s} |f_s(i)| - \sum_{i=k_s+1}^N |f_s(i)| = 0$. Therefore,

$$G_\lambda(\beta_1) - G_\lambda(\beta_2) = \mathbb{E}_{s \sim \rho_v^\pi} \sum_{j=1}^N \pi(a_j|s) \sum_{i=1}^N A^\pi(s, a_{[i]}) f_s(i) \tag{C.18a}$$
$$= \mathbb{E}_{s \sim \rho_v^\pi} \sum_{j=1}^N \pi(a_j|s) \{\sum_{i=1}^{k_s} A^\pi(s, a_{[i]}) |f_s(i)| - \sum_{i=k_s+1}^N A^\pi(s, a_{[i]}) |f_s(i)|\} \tag{C.18b}$$
$$\geq \mathbb{E}_{s \sim \rho_v^\pi} \sum_{j=1}^N \pi(a_j|s) \{\sum_{i=1}^{k_s} A^\pi(s, a_{[k_s]}) |f_s(i)| - \sum_{i=k_s+1}^N A^\pi(s, a_{[k_s+1]}) |f_s(i)|\} \tag{C.18c}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s)\{A^\pi(s, a_{[k_s]}) \sum_{i=1}^{k_s} |f_s(i)| - A^\pi(s, a_{[k_s+1]}) \sum_{i=k_s+1}^{N} |f_s(i)|\} \tag{C.18d}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s)\{A^\pi(s, a_{[k_s]}) \sum_{i=1}^{k_s} |f_s(i)| - A^\pi(s, a_{[k_s+1]}) \sum_{i=1}^{k_s} |f_s(i)|\} \tag{C.18e}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s)(A^\pi(s, a_{[k_s]}) - A^\pi(s, a_{[k_s+1]})) \sum_{i=1}^{k_s} |f_s(i)| \tag{C.18f}$$

$$\geq 0. \tag{C.18g}$$

where (C.18c) and (C.18g) hold since $A^\pi(s, a_{[i]})$ is non-increasing as i increases. Furthermore, at least one inequality of (C.18c) and (C.18g) will not hold at equality since $\sum_{i=1}^{N} \pi(a_i|s)A^\pi(s, a_i) = 0$, $\forall s \in \mathcal{S}$, and for non-trivial cases, $Pr\{A^\pi(s, a) = 0, \forall s \in \mathcal{S}, \forall a \in \mathcal{A}\} < 1$, which means $Pr\{\exists s_1, s_2 \in \mathcal{S}, a_1, a_2 \in \mathcal{A},\ s.t.\ A^\pi(s_1, a_1) \neq A^\pi(s_2, a_2)\} > 0$. Therefore, we have $G_\lambda(\beta_1) - G_\lambda(\beta_2) > 0$. □

C.5.2 Proof of Lemma 2

Lemma 2. *If Assumption 1 holds, then for every $\delta > 0$, $Q_{ij}^{s*}(\frac{2A^{max}}{\delta})$ is feasible to (C.7b) for any λ.*

Proof of Lemma 2. By substituting the optimal Q_{ij}^{s*} in (C.12) into (C.7b), we can reformulate the left hand side of (C.7b) as follows:

$$\mathbb{E}_{s\sim\rho_v^\pi}[\sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}Q_{ij}^{s*} + \frac{1}{\lambda}Q_{ij}^{s*}\log Q_{ij}^{s*}] \tag{C.19a}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}Q_{ij}^{s*} + \frac{1}{\lambda}Q_{ij}^{s*}[\frac{\lambda}{\beta}A^\pi(s, a_i) - \lambda D_{ij} + \log\frac{\pi(a_j|s)}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta}A^\pi(s, a_k) - \lambda D_{kj})}]\} \tag{C.19b}$$

$$= \mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^{N}\sum_{j=1}^{N} \frac{1}{\beta}Q_{ij}^{s*}A^\pi(s, a_i) + \frac{1}{\lambda}Q_{ij}^{s*}\log\frac{\pi(a_j|s)}{\sum_{k=1}^{N}\exp(\frac{\lambda}{\beta}A^\pi(s, a_k) - \lambda D_{kj})}\}. \tag{C.19c}$$

Now we prove that when $\beta = \frac{2A^{\max}}{\delta}$, $\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{1}{\beta}Q_{ij}^{s*}(\beta)A^\pi(s, a_i)\} \leq \frac{\delta}{2}$ and

$\mathbb{E}_{s\sim\rho_v^\pi}\{\frac{1}{\lambda}Q_{ij}^{s*}(\beta)\log\frac{\pi(a_j|s)}{\sum_{k=1}^N\exp(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj})}\}\leq\frac{\delta}{2}$ hold. For the first part, we have:

$$\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^N\sum_{j=1}^N\frac{1}{\beta}Q_{ij}^{s*}A^\pi(s,a_i)\} \tag{C.20a}$$

$$=\frac{1}{\beta}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^N[\sum_{j=1}^N Q_{ij}^{s*}]A^\pi(s,a_i)\} \tag{C.20b}$$

$$=\frac{1}{\beta}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^N\pi'(a_i|s)A^\pi(s,a_i)\} \tag{C.20c}$$

$$\leq\frac{1}{\beta}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^N\pi'(a_i|s)|A^\pi(s,a_i)|\} \tag{C.20d}$$

$$\leq\frac{A^{max}}{\beta}=\frac{\delta}{2}. \tag{C.20e}$$

For the second part, the followings hold:

$$\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{i=1}^N\sum_{j=1}^N\frac{1}{\lambda}Q_{ij}^{s*}\log\frac{\pi(a_j|s)}{\sum_{k=1}^N\exp(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj})}\} \tag{C.21a}$$

$$=\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j=1}^N\frac{1}{\lambda}(\sum_{i=1}^N Q_{ij}^{s*})\log\frac{\pi(a_j|s)}{\sum_{k=1}^N\exp(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj})}\} \tag{C.21b}$$

$$=\frac{1}{\lambda}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j=1}^N\pi(a_j|s)\log\frac{\pi(a_j|s)}{\sum_{k=1}^N\exp(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj})}\} \tag{C.21c}$$

$$\leq\frac{1}{\lambda}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j=1}^N\pi(a_j|s)\log\frac{\pi(a_j|s)}{\exp(\frac{\lambda}{\beta}A^\pi(s,a_j))}\} \tag{C.21d}$$

$$\leq\frac{1}{\lambda}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j=1}^N\pi(a_j|s)\log\frac{1}{\exp(\frac{\lambda}{\beta}A^\pi(s,a_j))}\} \tag{C.21e}$$

$$=\frac{1}{\lambda}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j=1}^N\pi(a_j|s)(-\frac{\lambda}{\beta}A^\pi(s,a_j))\} \tag{C.21f}$$

$$\leq\frac{1}{\beta}\mathbb{E}_{s\sim\rho_v^\pi}\{\sum_{j=1}^N\pi(a_j|s)|A^\pi(s,a_j)|\} \tag{C.21g}$$

$$\leq\frac{A^{max}}{\beta}=\frac{\delta}{2}. \tag{C.21h}$$

Therefore, $Q_{ij}^{s*}(\frac{2A^{\max}}{\delta})$ is feasible to (C.7b) for any λ. □

C.6 Gradient of the Objective in the Sinkhorn Dual Formulation

The closed-form gradient of the objective in the Sinkhorn dual formulation (4.9) is as follows:

$$
\begin{aligned}
&\delta - \mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\Big\{\frac{1}{\lambda}+\frac{1}{\lambda}\ln(\pi(a_j|s)) - \frac{1}{\lambda}\ln[\sum_{i=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)]\\
&-\frac{\beta}{\lambda}\cdot\frac{1}{\sum_{i=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)}\times\sum_{i=1}^{N}[\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)\times-\lambda A^\pi(s,a_i)\beta^{-2}]\Big\}\\
&+\mathbb{E}_{s\sim\rho_v^\pi}\sum_{i=1}^{N}\sum_{j=1}^{N}\Big\{\frac{\pi(a_j|s)}{\lambda}\frac{\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)}{\sum_{k=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj}\big)}\\
&+\frac{\beta\pi(a_j|s)}{\lambda}\cdot\frac{\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)\times-\lambda A^\pi(s,a_i)\beta^{-2}\times\sum_{k=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj}\big)}{(\sum_{k=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj}\big))^2}\\
&-\frac{\beta\pi(a_j|s)}{\lambda}\cdot\frac{\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)\times\sum_{k=1}^{N}[\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj}\big)\times-\lambda A^\pi(s,a_k)\beta^{-2}]}{(\sum_{k=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj}\big))^2}\Big\}.
\end{aligned}
$$

C.7 Proof of Theorem 7

Given the upper bound of Wassertein optimal β in Theorem 5 and the upper bound of Sinkhorn optimal β in Proposition 2, we are able to derive the following theorem:

Theorem 7. *Define* $\beta_{UB}=\max\{\frac{2A^{max}}{\delta},\bar{\beta}\}$. *We have:*

1. $F_\lambda(\beta)$ *converges to* $F(\beta)$ *uniformly on* $[0,\beta_{UB}]$*:* $\lim\limits_{\lambda\to\infty}\sup\limits_{0\le\beta\le\beta_{UB}}\Big|F_\lambda(\beta)-F(\beta)\Big|\le\lim\limits_{\lambda\to\infty}\frac{\beta_{UB}}{\lambda}N\ln N=0.$

2. $\lim\limits_{\lambda\to\infty}$ *argmin*$_{0\le\beta\le\beta_{UB}}F_\lambda(\beta)\subseteq$ *argmin*$_{0\le\beta\le\beta_{UB}}F(\beta)$.

Proof of Theorem 7. To show that $F_\lambda(\beta)$ converges to $F(\beta)$ uniformly on $[0,\beta_{\text{UB}}]$, it is equivalent to show that $\lim_{\lambda\to\infty}\sup_{0\le\beta\le\beta_{\text{UB}}}\Big|F_\lambda(\beta)-F(\beta)\Big|=0$. Let $\epsilon_s^\pi(\beta,i,j)=\max_{k=1\dots N}(A^\pi(s,a_k)-\beta D_{kj})-[A^\pi(s,a_i)-\beta D_{ij}]$, and $\epsilon_s^\pi(\beta,i,j)\ge 0$. First, we have

$$
\begin{aligned}
&\Big|F_\lambda(\beta)-F(\beta)\Big|\\
&=\Big|\beta\delta-\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\{\frac{\beta}{\lambda}+\frac{\beta}{\lambda}\ln(\pi(a_j|s))-\frac{\beta}{\lambda}\ln[\sum_{i=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)]\}\\
&+\mathbb{E}_{s\sim\rho_v^\pi}\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\beta}{\lambda}\frac{\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_i)-\lambda D_{ij}\big)\cdot\pi(a_j|s)}{\sum_{k=1}^{N}\exp\big(\frac{\lambda}{\beta}A^\pi(s,a_k)-\lambda D_{kj}\big)}-\beta\delta
\end{aligned}
$$

$$- \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \max_{i=1\dots N} (A^\pi(s,a_i) - \beta D_{ij}) \Big| \tag{C.22a}$$

$$\leq \Big| \frac{\beta}{\lambda} \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \Big| + \Big| \frac{\beta}{\lambda} \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \ln(\pi(a_j|s)) \Big|$$

$$+ \Big| \mathbb{E}_{s\sim\rho_v^\pi} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\beta}{\lambda} \frac{\exp\left(\frac{\lambda}{\beta} A^\pi(s,a_i) - \lambda D_{ij}\right) \cdot \pi(a_j|s)}{\sum_{k=1}^{N} \exp\left(\frac{\lambda}{\beta} A^\pi(s,a_k) - \lambda D_{kj}\right)} \Big|$$

$$+ \Big| \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \frac{\beta}{\lambda} \ln[\sum_{i=1}^{N} \exp\left(\frac{\lambda}{\beta} A^\pi(s,a_i) - \lambda D_{ij}\right)]$$

$$- \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \max_{i=1\dots N} (A^\pi(s,a_i) - \beta D_{ij}) \Big| \tag{C.22b}$$

$$\leq 2 \Big| \frac{\beta}{\lambda} \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \Big| + \Big| \frac{\beta}{\lambda} \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \ln(\pi(a_j|s)) \Big|$$

$$+ \Big| \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \frac{\beta}{\lambda} \ln[\sum_{i=1}^{N} \exp\left(\frac{\lambda}{\beta} A^\pi(s,a_i) - \lambda D_{ij}\right)]$$

$$- \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \max_{i=1\dots N} (A^\pi(s,a_i) - \beta D_{ij}) \Big|. \tag{C.22c}$$

In addition,

$$\Big| \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \frac{\beta}{\lambda} \ln[\sum_{i=1}^{N} \exp\left(\frac{\lambda}{\beta} A^\pi(s,a_i) - \lambda D_{ij}\right)]$$

$$- \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \max_{i=1\dots N} (A^\pi(s,a_i) - \beta D_{ij}) \Big| \tag{C.23a}$$

$$= \Big| \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \frac{\beta}{\lambda} \ln[\exp\left(\frac{\lambda}{\beta} \max_{k=1\dots N} (A^\pi(s,a_k) - \beta D_{kj})\right) \sum_{i=1}^{N} \exp\left(-\frac{\lambda}{\beta} \epsilon_s^\pi(\beta,i,j)\right)]$$

$$- \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \max_{i=1\dots N} (A^\pi(s,a_i) - \beta D_{ij}) \Big| \tag{C.23b}$$

$$= \Big| \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \frac{\beta}{\lambda} \{\ln[\exp\left(\frac{\lambda}{\beta} \max_{k=1\dots N} (A^\pi(s,a_k) - \beta D_{kj})\right)] + \ln[\sum_{i=1}^{N} \exp\left(-\frac{\lambda}{\beta} \epsilon_s^\pi(\beta,i,j)\right)]\}$$

$$- \mathbb{E}_{s\sim\rho_v^\pi} \sum_{j=1}^{N} \pi(a_j|s) \max_{i=1\dots N} (A^\pi(s,a_i) - \beta D_{ij}) \Big| \tag{C.23c}$$

$$=\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\frac{\beta}{\lambda}\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta}\epsilon_s^\pi(\beta,i,j)\right)]\Big|. \tag{C.23d}$$

Therefore,

$$\lim_{\lambda\to\infty}\sup_{0\le\beta\le\beta_{\mathrm{UB}}}\Big|F_\lambda(\beta)-F(\beta)\Big| \tag{C.24a}$$

$$\le\lim_{\lambda\to\infty}\frac{2\beta_{\mathrm{UB}}}{\lambda}\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\Big|+\lim_{\lambda\to\infty}\frac{\beta_{\mathrm{UB}}}{\lambda}\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln(\pi(a_j|s))\Big|$$

$$+\lim_{\lambda\to\infty}\sup_{0\le\beta\le\beta_{\mathrm{UB}}}\frac{\beta}{\lambda}\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta}\epsilon_s^\pi(\beta,i,j)\right)]\Big| \tag{C.24b}$$

$$=\lim_{\lambda\to\infty}\sup_{0\le\beta\le\beta_{\mathrm{UB}}}\frac{\beta}{\lambda}\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta}\epsilon_s^\pi(\beta,i,j)\right)]\Big|. \tag{C.24c}$$

In addition, $\forall\beta\in[0,\beta_{\mathrm{UB}}]$ and $\forall\lambda$, $\epsilon_s^\pi(\beta,i,j)$ is bounded since

$$\Big|\epsilon_s^\pi(\beta,i,j)\Big|=\Big|\max_{k=1\dots N}(A^\pi(s,a_k)-\beta D_{kj})-[A^\pi(s,a_i)-\beta D_{ij}]\Big| \tag{C.25}$$

$$\le 2\max_{s,a}A^\pi(s,a)+\beta_{\mathrm{UB}}\max_{i,j}D_{ij}$$

$$\le 2A^{\max}+\beta_{\mathrm{UB}}\max_{i,j}D_{ij}<\infty. \tag{C.26}$$

Then, $\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta}\epsilon_s^\pi(\beta,i,j)\right)]\Big|$ is bounded. Therefore in (C.24c), the optimal β can be achieved. Let $\beta^\lambda=\mathrm{argmax}_{0\le\beta\le\beta_{\mathrm{UB}}}\frac{\beta}{\lambda}\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta}\epsilon_s^\pi(\beta,i,j)\right)]\Big|$, and then we have:

$$\lim_{\lambda\to\infty}\sup_{0\le\beta\le\beta_{\mathrm{UB}}}\frac{\beta}{\lambda}\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta}\epsilon_s^\pi(\beta,i,j)\right)]\Big| \tag{C.27a}$$

$$=\lim_{\lambda\to\infty}\frac{\beta^\lambda}{\lambda}\Big|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)]\Big|. \tag{C.27b}$$

Let $\mathcal{K}_s^\pi(\beta,j)=\mathrm{argmax}_{k=1\dots N}A^\pi(s,a_k)-\beta D_{kj}$. Define $\sigma_s(j)=\min_{0\le\beta\le\beta_{\mathrm{UB}}}\min_{i=1\dots N,i\notin\mathcal{K}_s^\pi(\beta,j)}\epsilon_s^\pi(\beta,i,j)$. Then since $\epsilon_s^\pi(\beta,i,j)>0$ for $i\notin\mathcal{K}_s^\pi(\beta,j)$ based on its definition, we have $\sigma_s(j)>0$. On one hand, we have

$$\lim_{\lambda\to\infty}\ln[\sum_{i=1}^{N}\exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)] \tag{C.28a}$$

$$= \lim_{\lambda\to\infty} \ln[\sum_{i=1|i\notin\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right) + \sum_{i=1|i\in\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)] \tag{C.28b}$$

$$\leq \lim_{\lambda\to\infty} \ln[\sum_{i=1|i\notin\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(-\frac{\lambda}{\beta_{\text{UB}}}\sigma_s(j)\right) + \sum_{i=1|i\in\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(0\right)] \tag{C.28c}$$

$$= \lim_{\lambda\to\infty} \ln[\sum_{i=1|i\notin\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(-\frac{\lambda}{\beta_{\text{UB}}}\sigma_s(j)\right) + |\mathcal{K}_s^\pi(\beta_\lambda,j)|] \tag{C.28d}$$

$$= \lim_{\lambda\to\infty} \ln[|\mathcal{K}_s^\pi(\beta_\lambda,j)|]. \tag{C.28e}$$

On the other hand, we have

$$\lim_{\lambda\to\infty} \ln[\sum_{i=1}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)] \tag{C.29a}$$

$$= \lim_{\lambda\to\infty} \ln[\sum_{i=1|i\notin\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right) + \sum_{i=1|i\in\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)] \tag{C.29b}$$

$$\geq \lim_{\lambda\to\infty} \ln[\sum_{i=1|i\in\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)] \tag{C.29c}$$

$$= \lim_{\lambda\to\infty} \ln[\sum_{i=1|i\in\mathcal{K}_s^\pi(\beta_\lambda,j)}^{N} \exp\left(0\right)] \tag{C.29d}$$

$$= \lim_{\lambda\to\infty} \ln[|\mathcal{K}_s^\pi(\beta_\lambda,j)|]. \tag{C.29e}$$

Therefore, $\lim_{\lambda\to\infty} \left|\ln[\sum_{i=1}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)]\right| = \lim_{\lambda\to\infty} \ln[|\mathcal{K}_s^\pi(\beta_\lambda,j)|]$. Based on that, we have

$$\lim_{\lambda\to\infty} \frac{\beta^\lambda}{\lambda}\left|\mathbb{E}_{s\sim\rho_v^\pi}\sum_{j=1}^{N}\pi(a_j|s)\ln[\sum_{i=1}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)]\right| \tag{C.30a}$$

$$\leq \lim_{\lambda\to\infty} \frac{\beta^\lambda}{\lambda}\left|\sum_{j=1}^{N}\ln[\sum_{i=1}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)]\right| \tag{C.30b}$$

$$\leq \lim_{\lambda\to\infty} \frac{\beta^\lambda}{\lambda}\sum_{j=1}^{N}\left|\ln[\sum_{i=1}^{N} \exp\left(-\frac{\lambda}{\beta^\lambda}\epsilon_s^\pi(\beta^\lambda,i,j)\right)]\right| \tag{C.30c}$$

$$= \lim_{\lambda\to\infty} \frac{\beta^\lambda}{\lambda}\sum_{j=1}^{N}\ln[|\mathcal{K}_s^\pi(\beta_\lambda,j)|] \tag{C.30d}$$

$$\leq \lim_{\lambda\to\infty} \frac{\beta_{\mathrm{UB}}}{\lambda} N \ln N = 0, \tag{C.30e}$$

which means $\lim_{\lambda\to\infty} \sup_{0\leq\beta\leq\beta_{\mathrm{UB}}} \left|F_\lambda(\beta) - F(\beta)\right| \leq 0$. Furthermore, since $\lim_{\lambda\to\infty} \sup_{0\leq\beta\leq\beta_{\mathrm{UB}}} |F_\lambda(\beta) - F(\beta)| \geq 0$ holds naturally, we have $\lim_{\lambda\to\infty} \sup_{0\leq\beta\leq\beta_{\mathrm{UB}}} |F_\lambda(\beta) - F(\beta)| = 0$. Therefore, $F_\lambda(\beta)$ converges to $F(\beta)$ uniformly on $[0, \beta_{\mathrm{UB}}]$, which also indicates $F_\lambda(\beta)$ epi-converges to $F(\beta)$ on $[0, \beta_{\mathrm{UB}}]$ [208, 209]. By properties of epi-convergence, we have that $\lim_{\lambda\to\infty} \operatorname{argmin}_{0\leq\beta\leq\beta_{\mathrm{UB}}} F_\lambda(\beta) \subseteq \operatorname{argmin}_{0\leq\beta\leq\beta_{\mathrm{UB}}} F(\beta)$ [208]. □

C.8 Proof of Lemma 4.5

As $\lambda_k \to \infty$, SPO update converges to WPO update: $\lim_{\lambda_k\to\infty} \mathbb{F}^{\mathrm{SPO}}(\pi_k) \in \mathbb{F}^{\mathrm{WPO}}(\pi_k)$.

Proof of Lemma 4.5. Let $\xi_s^k(i,j) = \frac{\lambda}{\beta_k}\{\max_{l=1,\dots,N}(\hat{A}^{\pi_k}(s,a_l) - \beta_k D_{lj}) - [\hat{A}^{\pi_k}(s,a_i) - \beta_k D_{ij}]\}$. The SPO update with $\lambda \to \infty$ equals to:

$$\pi_{k+1}(a_i|s) = \lim_{\lambda\to\infty} \mathbb{F}^{\mathrm{SPO}}(\pi_k) = \lim_{\lambda\to\infty} \sum_{j=1}^{N} \frac{\exp\left(\frac{\lambda}{\beta_k}\hat{A}^{\pi_k}(s,a_i) - \lambda D_{ij}\right)}{\sum_{l=1}^{N} \exp\left(\frac{\lambda}{\beta_k}\hat{A}^{\pi_k}(s,a_l) - \lambda D_{lj}\right)} \pi_k(a_j|s) \tag{C.31a}$$

$$= \lim_{\lambda\to\infty} \sum_{j=1}^{N} \frac{\exp\left(\hat{A}^{\pi_k}(s, a_{\hat{k}_s^{\pi_k}(\beta_k,j)}) - \beta_k D_{\hat{k}_s^{\pi_k}(\beta_k,j)j}\right) \cdot \exp\left(-\xi_s^k(i,j)\right)}{\exp\left(\hat{A}^{\pi_t}(s, a_{\hat{k}_s^{\pi_k}(\beta_k,j)}) - \beta_k D_{\hat{k}_s^{\pi_k}(\beta_k,j)j}\right) \cdot \sum_{l=1}^{N} \exp(-\xi_s^k(l,j))} \pi_k(a_j|s) \tag{C.31b}$$

$$= \lim_{\lambda\to\infty} \sum_{j=1}^{N} \frac{\exp\left(-\xi_s^k(i,j)\right)}{\sum_{l=1}^{N} \exp(-\xi_s^k(l,j))} \pi_k(a_j|s) \tag{C.31c}$$

$$= \lim_{\lambda\to\infty} \sum_{j=1}^{N} \frac{\exp\left(-\xi_s^k(i,j)\right) \cdot \pi_k(a_j|s)}{\sum_{l\in\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} \exp(-\xi_s^k(l,j)) + \sum_{l\notin\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} \exp(-\xi_s^k(l,j))} \tag{C.31d}$$

$$= \sum_{j=1}^{N} \frac{\lim_{\lambda\to\infty} \exp\left(-\xi_s^k(i,j)\right) \cdot \pi_k(a_j|s)}{\sum_{l\in\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} \lim_{\lambda\to\infty} \exp(-\xi_s^k(l,j)) + \sum_{l\notin\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} \lim_{\lambda\to\infty} \exp(-\xi_s^k(l,j))} \tag{C.31e}$$

$$= \sum_{j=1}^{N} \frac{I_{\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)}(i)}{|\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)|} \pi_k(a_j|s), \tag{C.31f}$$

where I denotes the indicator function; (C.31f) holds because as $\lambda \to \infty$, $\xi_s^k(i,j) = \infty$ for

$i \notin \hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j)$ and 0 otherwise, thus $\lim_{\lambda\to\infty} \exp\left(-\xi_s^k(i,j)\right) = 0$ for $i \notin \hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j)$ and 1 otherwise.

Let $f_s^k(i,j) = \frac{1}{|\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)|}$ if $i \in \hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j)$, and $f_s^k(i,j) = 0$ otherwise. Therefore, SPO update with $\lambda \to \infty$ equals to the following WPO update, $\mathbb{F}^{\text{WPO}}(\pi_k) = \sum_{j=1}^N \pi_k(a_j|s) f_s^k(i,j)$. □

C.9 Proof of Theorem 8

Theorem 8. ***(Performance improvement)*** *For any initial state distribution υ and any $\beta_k \geq 0$, if $||\hat{A}^\pi - A^\pi||_\infty \leq \epsilon$ for some $\epsilon > 0$, let $\hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j) = argmax_{i=1,\ldots,N}\{\hat{A}^{\pi_k}(s,a_i) - \beta_k D_{ij}\}$, WPO policy update (and SPO with $\lambda \to \infty$) guarantee the following performance improvement bound when the inaccurate advantage function $\hat{A}^\pi$ is used,*

$$J(\pi_{k+1}) \geq J(\pi_k) + \beta_k \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{j=1}^N \pi_k(a_j|s) \sum_{i\in\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} f_s^k(i,j) D_{ij} - \frac{2\epsilon}{1-\gamma}. \tag{4.10}$$

Proof of Theorem 8.

$$\begin{aligned}
J(\pi_{k+1}) - J(\pi_k) &= \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \mathbb{E}_{a\sim\pi_{k+1}}[A^{\pi_k}(s,a)] && \text{(C.32a)}\\
&= \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{i=1}^N \pi_{k+1}(a_i|s) A^{\pi_k}(s,a_i) && \text{(C.32b)}\\
&= \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{i=1}^N \sum_{j=1}^N \pi_k(a_j|s) f_s^k(i,j) A^{\pi_k}(s,a_i) && \text{(C.32c)}\\
&= \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{j=1}^N \pi_k(a_j|s) \sum_{i=1}^N f_s^k(i,j) A^{\pi_k}(s,a_i) && \text{(C.32d)}\\
&= \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{j=1}^N \pi_k(a_j|s) \sum_{i\in\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} f_s^k(i,j) A^{\pi_k}(s,a_i) && \text{(C.32e)}\\
&\geq \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{j=1}^N \pi_k(a_j|s) \sum_{i\in\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} f_s^k(i,j)[A^{\pi_k}(s,a_j) + \beta_k D_{ij} - 2\epsilon] && \text{(C.32f)}\\
&= \beta_k \mathbb{E}_{s\sim\rho_\upsilon^{\pi_{k+1}}} \sum_{j=1}^N \pi_k(a_j|s) \sum_{i\in\hat{\mathcal{K}}_s^{\pi_k}(\beta_k,j)} f_s^k(i,j) D_{ij} - \frac{2\epsilon}{1-\gamma}, && \text{(C.32g)}
\end{aligned}$$

where (C.32a) holds due to the performance difference lemma in [129]; (C.32f) follows from the definition of $\hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j)$ and the fact that $||\hat{A}^{\pi_k} - A^{\pi_k}||_\infty \leq \epsilon$, therefore for $i \in \hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j)$, $[A^{\pi_k}(s, a_i)+\epsilon]-\beta_k D_{ij} \geq \hat{A}^{\pi_k}(s, a_i)-\beta_k D_{ij} \geq \hat{A}^{\pi_k}(s, a_j)-\beta_k D_{jj} = \hat{A}^{\pi_k}(s, a_j) \geq A^{\pi_k}(s, a_j)-\epsilon$; (C.32g) holds since $\mathbb{E}_{a\sim\pi}[A^\pi(s, a)] = 0$. □

C.10 Proof of Theorem 9

Theorem 9. ***(Global convergence)*** *Under Assumption 2, we have for any $\beta_k \geq 0$, (WPO) satisfies that*

$$\|V^\star - V^{\pi_{k+1}}\|_\infty \leq \gamma\|V^\star - V^{\pi_k}\|_\infty + \beta_k\|D\|_\infty, \tag{4.11}$$

and (SPO) satisfies that

$$\|V^\star - V^{\pi_{k+1}}\|_\infty \leq \gamma\|V^\star - V^{\pi_k}\|_\infty + 2\frac{\beta_k}{1-\gamma}\left(\|D\|_\infty + 2\frac{\log N}{\lambda}\right). \tag{4.12}$$

If $\lim_{k\to\infty}\beta_k = 0$, we further have $\lim_{k\to\infty} J(\pi_k) = J^\star$.

Proof of Theorem 9. Our proof is inspired by the work [29].

We use the shorthand π_s for the probability distribution $\pi(\cdot \mid s)$ on the actions and denote the probability distribution on the action space $\mathcal{A}$ as Δ. To save notations, we rewrite π_{k+1}, π_k and β_k as π^+, π and β respectively. We use d for either d_{W} or d_{S} in the following derivation. Note $d \leq \|D\|_\infty =: D$ for both cases[2], and $d_{\mathrm{S}} \geq -2\frac{\log N}{\lambda}$. [3]

Since a policy π is indeed just a member of $\prod_{i=1}^{|S|}\Delta$, we find that the problem (4.7) can be split into $|\mathcal{S}|$ many optimization problems. For each $s \in \mathcal{S}$, we need to solve

$$\max_{\pi_s'\in\Delta} \quad \rho^\pi(s)\mathbb{E}_{a\sim\pi'(\cdot|s)}[A^\pi(s, a)] - \beta\rho^\pi(s)d(\pi_s', \pi_s). \tag{C.33}$$

Denote the quality function of π as $Q^\pi(s, a) = \mathbb{E}[R_t|s_t = s, a_t = a; \pi]$, and the value function of π as $V^\pi(s) = \mathbb{E}[R_t|s_t = s; \pi]$, we find that $A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s)$. Since the second

[2]For Sinkhorn divergence, note that the entropy function is always nonnegative.

[3]This lower bound is obtained via $d_{\mathrm{S}}(\pi', \pi|\lambda) \geq \min_{Q\geq 0, \sum_{i,j} Q_{ij}=1}\left\{\langle Q, D\rangle - \frac{1}{\lambda}h(Q)\right\} \overset{(a)}{=} \langle Q, D\rangle - \frac{1}{\lambda}h(Q)|_{Q_{ij}=\frac{\exp(-\lambda D_{ij})}{\sum_{i,j}\exp(-\lambda D_{ij})}} = -\frac{1}{\lambda}\log\left(\sum_{i,j}\exp(-\lambda D_{ij})\right) \overset{(b)}{\geq} -\frac{2\log N}{\lambda}$. Here in the step (a), we use the Lagrangian multiplier method to derive the optimal $Q_{ij} = \frac{\exp(-\lambda D_{ij})}{\sum_{i,j}\exp(-\lambda D_{ij})}$. In the step (b), we use the fact that $\log(\sum_{i=1}^n \exp(x_i)) \leq \max\{x_1, \ldots, x_n\} + \log n$ for any $x_1, \ldots, x_n \in \mathbb{R}$ and $D_{ii} = 0$ for any i.

term is only a function of the current policy π and the state s, we find that Problem (C.33) is further equivalent to (in the sense of the same solution set):

$$\max_{\pi'_s \in \Delta} \quad \mathbb{E}_{a\sim\pi'_s}[Q^\pi(s,a)] - \beta d(\pi'_s, \pi_s). \tag{C.34}$$

Here we use $\rho_0(s) > 0$ for all s. Let $\bar{\pi}$ be a solution of the policy iteration:

$$\bar{\pi}_s \in \arg\max_{\pi'_s} \ \mathbb{E}_{a\sim\pi'_s}[Q^\pi(s,a)]. \tag{C.35}$$

Also define the bellman operator $T : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$ and the operator $T^{\pi'} : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$: for any $V \in \mathbb{R}^{|\mathcal{S}|}$,

$$(TV)_s = \max_{a\in\mathcal{A}} r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}[V(s')], \tag{C.36}$$

$$(T^{\pi'}V)_s = \mathbb{E}_{a\sim\pi'_s}[r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}V(s')]. \tag{C.37}$$

Using the relation between the quality function and the value function, $Q^\pi(s,a) = r(s,a) + \mathbb{E}_{s'\sim P(\cdot|s,a)}[V^\pi(s')]$, we can rewrite the above equations in terms of the quality function for $V = V^\pi$:

$$(TV^\pi)_s = \max_{a\in\mathcal{A}} r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}[V^\pi(s')] = \max_{a\in\mathcal{A}} Q(s,a) = T^{\bar{\pi}}V^\pi, \tag{C.38}$$

$$(T^{\pi'}V^\pi)_s = \mathbb{E}_{a\sim\pi'_s}[Q^\pi(s,a)]. \tag{C.39}$$

Let us consider $d = d_{\mathrm{W}}$ first. Using the optimality of π^+ for the problem (C.33), we know that

$$\begin{aligned} &\mathbb{E}_{a\in\pi_s^+}[Q^\pi(s,a)] - \beta d(\pi_s^+, \pi_s) \geq \mathbb{E}_{a\in\bar{\pi}_s}[Q^\pi(s,a)] - \beta d(\bar{\pi}_s, \pi_s) \\ \implies &\mathbb{E}_{a\in\pi_s^+}[Q^\pi(s,a)] \geq \mathbb{E}_{a\in\bar{\pi}_s}[Q^\pi(s,a)] - \beta D. \end{aligned} \tag{C.40}$$

and

$$\begin{aligned} &\mathbb{E}_{a\in\pi_s^+}[Q^\pi(s,a)] - \beta d(\pi_s^+, \pi_s) \geq \mathbb{E}_{a\in\pi_s}[Q^\pi(s,a)] - \beta d(\pi_s, \pi_s) \\ \implies &\mathbb{E}_{a\in\pi_s^+}[Q^\pi(s,a)] \geq \mathbb{E}_{a\in\pi_s}[Q^\pi(s,a)] = V^\pi(s). \end{aligned} \tag{C.41}$$

Using the notation in (C.38) and (C.39), (C.40) and (C.41) become

$$T^{\pi^+}V^\pi \geq TV^\pi - \beta D\mathbf{1}_{|\mathcal{S}|}, \tag{C.42}$$

$$T^{\pi^+}V^\pi \geq V^\pi. \tag{C.43}$$

Here $\mathbf{1}_{|\mathcal{S}|}$ is a vector of all one entries and the inequality $\geq$ means entrywisely larger than or equal to. By iteratively applying T^{π^+} to (C.43) and use the fact that T^{π^+} is a monotone and contraction map with V^{π^+} as the unique fixed point, we have

$$V^{\pi^+} \geq \cdots \geq (T^{\pi^+})^2 V^{\pi} \geq T^{\pi^+} V^{\pi} \geq V^{\pi}. \tag{C.44}$$

Hence we have

$$0 \overset{(a)}{\leq} V^\star - V^{\pi^+} \overset{(b)}{\leq} V^\star - T^{\pi^+} V^{\pi} \overset{(c)}{\leq} V^\star - TV^{\pi} + \beta D \mathbf{1}_{|\mathcal{S}|}. \tag{C.45}$$

Here the inequality (a) is due to the optimality of $V^\star$. The inequality (b) is due to (C.44), and the inequality (c) is due to (C.42). Now using the fact $V^\star$ is the unique fixed point of T, and T is a monotone and contraction map, we have from (C.45) that

$$\|V^\star - V^{\pi^+}\|_\infty \leq \|TV^\star - TV^{\pi}\|_\infty + \beta D \leq \gamma \|V^\star - V^{\pi}\|_\infty + \beta D. \tag{C.46}$$

Next consider $d = d_{\mathrm{S}}$. The optimality of π^+ reveals that for $\tilde{\pi} = \bar{\pi}$ or π:

$$\begin{aligned} &\mathbb{E}_{a \in \pi_s^+}[Q^\pi(s,a)] - \beta d(\pi_s^+, \pi_s) \geq \mathbb{E}_{a \in \tilde{\pi}_s}[Q^\pi(s,a)] - \beta d(\tilde{\pi}_s, \pi_s) \\ \Longrightarrow\ &\mathbb{E}_{a \in \pi_s^+}[Q^\pi(s,a)] \geq \mathbb{E}_{a \in \tilde{\pi}_s}[Q^\pi(s,a)] - \beta(D + 2\frac{\log N}{\lambda}). \end{aligned} \tag{C.47}$$

Thus we have the following

$$T^{\pi^+} V^{\pi} \geq TV^{\pi} - \beta(D + \frac{2\log N}{\lambda}) \mathbf{1}_{|\mathcal{S}|}, \tag{C.48}$$

$$T^{\pi^+} V^{\pi} \geq V^{\pi} - \beta(D + \frac{2\log N}{\lambda}) \mathbf{1}_{|\mathcal{S}|}. \tag{C.49}$$

By iteratively applying T^{π^+} to (C.49) and use the fact that T^{π^+} is a monotone and contraction map with V^{π^+} as the unique fixed point, we have

$$V^{\pi^+} \geq V^{\pi} - \frac{\beta}{1-\gamma}(D + 2\frac{\log N}{\lambda}) \mathbf{1}_{|\mathcal{S}|}. \tag{C.50}$$

Hence we have

$$\begin{aligned} 0 \overset{(a)}{\leq} V^\star - V^{\pi^+} \overset{(b)}{\leq}& V^\star - T^{\pi^+} V^{\pi} + \frac{\beta}{1-\gamma}(D + 2\frac{\log N}{\lambda}) \mathbf{1}_{|\mathcal{S}|} \\ \overset{(c)}{\leq}& V^\star - TV^{\pi} + 2\frac{\beta}{1-\gamma}(D + 2\frac{\log N}{\lambda}) \mathbf{1}_{|\mathcal{S}|}. \end{aligned} \tag{C.51}$$

Here the inequality (a) is due to the optimality of $V^\star$. The inequality (b) is due to (C.50), and the inequality (c) is due to (C.48). A similar derivation as (C.46) shows the inequality in the theorem. Hence the theorem is established. $\square$

C.11 Computational Complexity of the Algorithm 3

Our overall algorithm applies a general actor-critic framework: the actor follows the proposed WPO or SPO update while the critic follows TD methods. The computational complexity depends on (i) the per-iteration computation cost of the policy and critic update and (ii) the iteration complexity of the actor-critic method. Here we mainly discuss the per-iteration computation cost of the policy update, as studies on the iteration complexity of actor-critic framework for constrained policy optimization are limited.

The computation cost of WPO and SPO updates at each iteration depends on the selection of β_k. If β_k is chosen time dependently, the computation cost of WPO/SPO policy update is $O(n_a^2 n_s)$, where n_a and n_s are the number of actions and states to perform policy update. If we set β_k as the dual optimizer, there will be additional cost to run gradient descent to solve the one-dimensional dual formulation. As discussed in our experiments, we can set β_k to be the dual optimizer only in the first a few iterations and use a decaying afterward. Therefore, the average computational complexity of a policy update step can be $O(n_a^2 n_s)$.

C.12 Difference between SPO/WPO and Other Exponential Style Updates

Sinkhorn divergence smooths the original Wasserstein by adding an entropy term, which causes the SPO update to contain exponential components similar to standard exponential style updates such as NPG [130, 198]. Thus, SPO can be viewed as a smoother version of WPO update. Nonetheless, it's important to note that SPO/WPO updates differ fundamentally from standard exponential style updates that are based solely on entropy or KL divergence. In both SPO and WPO, the probability mass at action a is redistributed to neighboring actions with high value (i.e., those a' with high $A^\pi(s, a') - \beta d(a', a)$). In contrast, in these standard exponential style updates, probability mass at action a is reweighted according to its exponential advantage or Q value.

C.13 Exploration Properties of WPO/SPO

Compared to the Wasserstein metric, the KL divergence between policies is often larger, especially when considering the policy shifts of closely related actions, as shown in Figure

4.2. In practice, when employing the same trust region size δ, Wasserstein metric allows for more admissible policies within the trust region compared to KL, thereby leading to better exploration. This advantage is demonstrated in our motivating example in Figure 4.3.

Furthermore, Sinkhorn divergence has even more exploration advantages than using Wasserstein. As Sinkhorn smooths the original Wasserstein with an entropy term, it includes additional smoother (more uniform) policies in the trust region, leading to even faster exploration.

Our numerical results in Section 4.7 also support that WPO/SPO explores better than KL; and SPO achieves faster exploration than WPO.

C.14 Policy Parametrization, Prior Work on Nonparametric Policy

As noted in [248], the suboptimality of policy gradient is not due to parametrization (e.g., neural network), but is a result of the parametric distribution assumption imposed on policy, which constrains policies to a predefined set. In our work, we strive to avoid suboptimality by circumventing the parametric distribution assumption imposed on policy, while still allowing for parametrization of policy in our empirical studies.

Previous research, such as [5, 198], has investigated theoretical policy update rules based on KL divergence without making explicit parametric assumptions about the policy being used. However, to our best knowledge, no prior work has explored theoretical policy update rules based on Wasserstein metric or Sinkhorn divergence.

C.15 T-tests to Compare the Performance of WPO, SPO with BGPG and WNPG

We conduct independent two-sample one-tailed t-tests [240] to compare the mean performance of our proposed methods (WPO and SPO) with two other Wasserstein-based policy optimization approaches: BGPG [189] and WNPG [177]. Specifically, we formulate four alternative hypotheses for each task: $J_{\text{WPO}} > J_{\text{BGPG}}$, $J_{\text{WPO}} > J_{\text{WNPG}}$, $J_{\text{SPO}} > J_{\text{BGPG}}$, and $J_{\text{SPO}} > J_{\text{WNPG}}$.

MuJuCo continuous control tasks are considered for the t-tests, with a sample size of 10 for each algorithm. All t-tests are conducted at a confidence level of 90%. The results of

the t-tests are presented in Table C.5, where a checkmark (✓) indicates that the alternative hypothesis is supported with 90% confidence, and a dash (−) indicates a failure to support the alternative hypothesis.

Based on the results presented in Table C.5, we can conclude the following:

- The mean performance of WPO is higher than BGPG with 90% confidence for all tasks.
- The mean performance of WPO is higher than WNPG with 90% confidence for all tasks.
- The mean performance of SPO is higher than BGPG with 90% confidence for all tasks except Ant-v2.
- The mean performance of SPO is higher than WNPG with 90% confidence for all tasks except HalfCheetah-v2.

We note that though SPO's performance is not statistically significantly higher than BGPG or WNPG in Ant-v2 and HalfCheetah-v2 tasks, SPO demonstrates a faster convergence speed than WNPG and BGPG in these two tasks.

Table C.5: T-tests results on the performance of WPO, SPO, BGPG and WNPG

Environment	$J_{\text{WPO}} > J_{\text{BGPG}}$	$J_{\text{WPO}} > J_{\text{WNPG}}$	$J_{\text{SPO}} > J_{\text{BGPG}}$	$J_{\text{SPO}} > J_{\text{WNPG}}$
HalfCheetah-v2	✓	✓	✓	–
Hopper-v2	✓	✓	✓	✓
Walker2d-v2	✓	✓	✓	✓
Ant-v2	✓	✓	–	✓
Humanoid-v2	✓	✓	✓	✓

Appendix D

APPENDIX FOR CHAPTER 5

D.1 Proof of Theorem 10

Theorem 10. ***(Closed-form policy update)*** *Let* $k_s^\pi(\beta, a) = argmax_{a_k \in \mathcal{A}} \zeta A^\pi(s, a_k) - (1-\zeta)\log D_\omega(s, a_k) - \beta d(a, a_k)$. *Assume that* $A^\pi(s,a)$ *and* $D_\omega(s,a)$ *are bounded, then an optimal solution to the EGWPO problem* (5.4) *is:*

$$\pi^*(a'|s) = \int_{a \in \mathcal{A}} \pi(a|s) f_s^*(a, a') da, \tag{5.5}$$

where $f_s^*(a,a') = 1$ *if* $a' = k_s^\pi(\beta^*, a)$ *and* $f_s^*(a,a') = 0$ *otherwise, and* β^* *is an optimal Lagrangian multiplier corresponding to the following dual formulation:*

$$\begin{aligned} \min_{\beta \geq 0} \{\beta\delta + \mathbb{E}_{s \sim \rho_v^\pi} \int_{a \in \mathcal{A}} \pi(a|s) \max_{a' \in \mathcal{A}} (\zeta A^\pi(s, a') \\ - (1-\zeta)\log D_\omega(s, a') - \beta d(a, a'))\}. \end{aligned} \tag{5.6}$$

Proof. First, we denote Q_s as the joint distribution of $\pi(\cdot|s)$ and $\pi'(\cdot|s)$ with $\int_{a'} Q_s(a,a') = \pi(a|s)$ and $\int_a Q_s(a,a') = \pi'(a'|s)$. Also, let $f_s(a,a')$ represent the conditional distribution of $\pi'(a'|s)$ under $\pi(a|s)$. Then $Q_s(a,a') = \pi(a|s) f_s(a,a')$, $\pi'(a'|s) = \int_a Q_s(a,a') = \int_a \pi(a|s) f_s(a,a')$. In addition:

$$\begin{aligned} d_{\mathrm{W}}(\pi'(\cdot|s), \pi(\cdot|s)) &= \inf_{Q_s \in \Pi(\pi, \pi')} \int d(a,a') dQ_s(a,a') \\ &= \inf_{f_s} \int_a \int_{a'} d(a,a') \pi(a|s) f_s(a,a') da da', \end{aligned} \tag{D.1}$$

$$\begin{aligned} &\mathbb{E}_{a \sim \pi'(\cdot|s)}[\zeta A^\pi(s,a) - (1-\zeta)\log D_\omega(s,a)] \\ &= \int_{a'} [\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a')] \pi'(a'|s) da' \\ &= \int_a \int_{a'} [\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a')] \\ &\qquad \pi(a|s) f_s(a,a') da da'. \end{aligned} \tag{D.2}$$

Thus, the EGWPO problem in (5.4) can be reformulated as:

$$\max_{f_s(a,a')\geq 0} \mathbb{E}_{s\sim\rho_v^\pi} \int_a \int_{a'} [\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a')] \pi(a|s) f_s(a,a') da da' \tag{D.3a}$$

$$s.t. \quad \mathbb{E}_{s\sim\rho_v^\pi} \int_a \int_{a'} d(a,a')\pi(a|s) f_s(a,a') da da' \leq \delta, \tag{D.3b}$$

$$\int_{a'} f_s(a,a') da' = 1, \quad \forall s \in \mathcal{S}, a \in \mathcal{A}. \tag{D.3c}$$

Note that (D.3b) is equivalent to $\mathbb{E}_{s\sim\rho_v^\pi} \min_{f_s(a,a')} \int_a \int_{a'} d(a,a')\pi(a|s) f_s(a,a') da da' \leq \delta$. Since if (D.3b) has feasible $f_s(a,a')$, $\mathbb{E}_{s\sim\rho_v^\pi} \min_{f_s(a,a')} \int_a \int_{a'} d(a,a')\pi(a|s) f_s(a,a') da da' \leq \delta$ must hold.

Since both the objective function and the constraint are linear in $f_s(a,a')$, (D.3) is a convex optimization problem. Also, Slater's condition holds for (D.3) as the feasible region has an interior point, which is $f_s(a,a) = 1\ \forall a$, and $f_s(a,a') = 0\ \forall a \neq a'$. Meanwhile, since $A^\pi(s,a)$ and $D_\omega(s,a)$ are assumed to be bounded, the objective is bounded above. Therefore, strong duality holds for (D.3). At this point we can derive the dual problem of (D.3) as its equivalent reformulation:

$$\begin{aligned} \min_{\beta\geq 0, \zeta_s(a)} \quad & \beta\delta + \int_{s\in\mathcal{S}} \int_{a\in\mathcal{A}} \zeta_s(a) da ds \\ s.t. \quad & (\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a'))\pi(a|s) \\ & - \beta d(a,a')\pi(a|s) - \frac{\zeta_s(a)}{\rho_v^\pi(s)} \leq 0, \quad \forall s \in \mathcal{S}, a, a' \in \mathcal{A}. \end{aligned} \tag{D.4}$$

We observe that with a fixed β, the optimal $\zeta_s(a)$ will be achieved at:

$$\begin{aligned} \zeta_s^\beta(a) = & \max_{a'\in\mathcal{A}} \rho_v^\pi(s)\pi(a|s)(\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a') \\ & - \beta d(a,a')). \end{aligned} \tag{D.5}$$

Denote β^* as an optimal solution to (D.4)) and $f_s^*(a,a')$ as an optimal solution to (D.3). Due to the complimentary slackness, the following equations hold:

$$\begin{aligned} & \{(\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a'))\pi(a|s) - \beta d(a,a')\pi(a|s) \\ & - \frac{\zeta_s^{\beta^*}(a)}{\rho_v^\pi(s)}\} f_s^*(a,a') = 0, \quad \forall s,a,a'. \end{aligned}$$

In this case, $f_s^*(a,a')$ can have non-zero values only when

$$(\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a'))\pi(a|s) - \beta d(a,a')\pi(a|s) - \frac{\zeta_s^{\beta^*}(a)}{\rho_v^\pi(s)} = 0,$$

which means $\zeta_s^{\beta^*}(a) = \rho_v^\pi(s)\pi(a|s)(\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a') - \beta d(a,a'))$. Given the expression of the optimal $\zeta_s^{\beta^*}(a)$ in (D.5), $f_s^*(i,j)$ can have non-zero values only when $a' \in \mathcal{K}_s^\pi(\beta^*, a)$, where $\mathcal{K}_s^\pi(\beta^*, a) = \text{argmax}_{a_k \in \mathcal{A}} \zeta A^\pi(s,a_k) - (1-\zeta)\log D_\omega(s,a_k) - \beta d(a,a_k)$.

When there exists a unique optimizer, i.e., $|\mathcal{K}_s^\pi(\beta^*,a)| = 1$, let $k_s^\pi(\beta^*,a)$ denote the optimizer. Since $\int_{a'} f_s^*(a,a')da' = 1$ as indicated in (D.3c), the only optimal solution is:

$$f_s^*(a,a') = \begin{cases} 1 & \text{if } a' = k_s^\pi(\beta^*,a), \\ 0 & \text{otherwise.} \end{cases}$$

When there exists multiple optimizers, i.e., $|\mathcal{K}_s^\pi(\beta^*,a)| > 1$, the optimal weights $f_s^*(a,a')$ for $i \in \mathcal{K}_s^\pi(\beta^*,a)$ could be determined by solving the following linear programming:

$$\begin{aligned} \max_{f_s^*(a,a')\geq 0, a' \in \mathcal{K}_s^\pi(\beta^*,a)} \quad & \mathbb{E}_{s\sim\rho_v^\pi} \int_a \pi(a|s) \sum_{a' \in \mathcal{K}_s^\pi(\beta^*,a)} \\ & [\zeta A^\pi(s,a') - (1-\zeta)\log D_\omega(s,a')] f_s^*(a,a')da \\ s.t. \quad & \mathbb{E}_{s\sim\rho_v^\pi} \int_a \pi(a|s) \sum_{a' \in \mathcal{K}_s^\pi(\beta^*,a)} d(a,a') f_s^*(a,a')da \leq \delta, \\ & \sum_{a' \in \mathcal{K}_s^\pi(\beta^*,a)} f_s^*(a,a') = 1, \quad \forall s \in \mathcal{S}, a \in \mathcal{A}. \end{aligned} \tag{D.6}$$

And then the corresponding optimal solution is, $\pi^*(a'|s) = \int_a \pi(a|s) f_s^*(a,a')da$.

Last, by substituting (C.3) into the dual problem (D.4), we can reformulate (D.4) into:

$$\begin{aligned} & \min_{\beta\geq 0}\{\beta\delta + \int_{s\in\mathcal{S}} \int_{a\in\mathcal{A}} \zeta_s(a)dads\} \\ & = \min_{\beta\geq 0}\{\beta\delta + \mathbb{E}_{s\sim\rho_v^\pi} \int_{a\in\mathcal{A}} \pi(a|s) \max_{a'\in\mathcal{A}} (\zeta A^\pi(s,a') \\ & \quad - (1-\zeta)\log D_\omega(s,a') - \beta d(a,a'))\}. \end{aligned} \tag{D.7}$$

The optimal β can then be obtained by solving (D.7). □