Your Project Title (e.g. A Note on Double Q-Learning)

Cheng-En Hsieh
Department of Computer Science
National Yang Ming Chiao Tung University
a0816183.cs08@nycu.edu.tw

1 Introduction

We use the neural network to implement TPRO and PPO, and get the competitive result in the applications such as games and robotics. However, the global convergence of policy optimization remains less understood because of multiple sources of nonconvexity. Therefore, PPO and TRPO are only ensured to monotonically improve the expected total reward over the infinite-dimensional policy space, when the global optimal policy, the rate of convergence, and the impact of using nerual network to approximate policy and action-value function all remain unclear.

This paper try to solve the three main questions to bridge the gap between theory and practice:
(i) In the ideal case which allows for infinite-dimensional policy updates by exact action-value function. how do PPO and TRPO converge to the optimal policy ?
(ii) When we use a neural network to approximate the action-value function, how does TD learning converge to an action-value fuction with sufficient accuracy within iteration of PPO and TRPO?
(iii)When we use another neural network to approximate the policy with the approximate action-value function which is attained by TD learning, how does SGD converge to a better policy which approximates the optimal policy within iteration of PPO and TRPO?

To answer question(i),this paper cast the infinite-dimensional policy updates in the ideal case as mirror descent iterations. To avoid the lack of convexity, we prove that the expected total reward satisfies a notation of one-point monotonicity, which guaranteed that the optimal policy exisits. We show that the exact action-value function plays the role of dual iterate in the context of infinite-dimensional mirror descent, while the ideal policy plays the role of primal iterate.The dual and primal errors arise from using neural networks to approximate the exact action-value function and the ideal improved policy.

To analyze such errors in questions (ii) and (iii), we use the convergence analysis of TD for minimizing the MSBE and SGD for minimizing the MSE. We show that the approximate functions induced by the overparametrization of neural networks ensure the global convergence of both the MSBE and the MSE, ,which is related to the dual and primal errors at a sublinear rate to zero. We can make a analysis with the dual and primal errors to establish the global rate of convergence of PPO.

2 Problem Formulation

2.1 Markov Decision Process:

We consider the Markov decision process (S,A,P,r,γ), where S is a compact state space, A is a finite action space, P : S $\times$ S $\times$ A $\to$ R is the transition kernel, r : S $\times$ A $\to$ R is the reward function, and $\gamma \in (0, 1)$ is the discount factor. We track the performance of a policy π : A$\times$ S $\to$ R using its action-value function (Q-function) Q^π : S $\times$ A $\to$ R, which is defined as
Q^π(s,a)=(1-γ)$\cdot$E[$\sum_{t=0}^{\infty}\gamma\cdot$r$^t(s_t, a_t)|s_0 = s, a_0 = a, a_t \sim \pi(\cdot|s_t), s_{t+1}\ P(\cdot|s_t, a_t)]$

Correspondingly, the state-value function V$^{:S}\to$ R of a policy π is defined as V(s) $=$ $(1-\gamma)$E[$\sum_{t=0}^{\infty}\gamma_t\cdot$r(s$_t, a_t)|s_0 = s, a_t \sim \pi\ (\cdot|s_t), s_{t+1}\sim$P$(\cdot|s_t, a_t)]$ The advantage function Aπ: S$\times$A

$\rightarrow$ R of a policy π is defined as $A^{\pi}(s,a)$ = Q^{π}(s, a)$V^{\pi}(s)$. We denote by π(s) and δ^{π}(s, a) = π(a | s) $\cdot$ π(s) the stationary state distribution and the stationary state-action distribution associated with a policy , respectively. Correspondingly, we denote by $E_{\delta_{\pi}}[\cdot]$ and $E_{v_{\pi}}[\cdot]$ the expectations $E_{(s,a)\sim\delta_{\pi}}[\cdot]$ = $E_{a\sim\pi(\cdot|s),s\sim\nu_{\pi}(\cdot)}[\cdot]$ and $E_{s\sim\nu_{\pi}}[\cdot]$. Meanwhile, we denote by <$\cdot$,$\cdot$> the inner product over A, e.g., we have V^{π}(s) = $E_{a\sim\pi(\cdot|s)}$[$Q^{\pi}(s,a)$] = <Q^{π}(s, $\cdot$),$\pi(\cdot|s)$ > .

2.2 PPO and TRPO:

At the k-th iteration of PPO, the policy parameter θ is updated by $\theta_{k+1} \leftarrow argmax_{\theta}$ E[$(\pi_{\theta}(a|s)/\pi_{k}(a|s))\cdot A_{k}(s,a)\beta_{k}\cdot$KL($\pi_{\theta}(\cdot|s)||\pi_{\theta_{k}}(\cdot|s)$)] where A_k is an estimator of $A^{\pi_{\theta_k}}$ and $E[\cdot]$ is taken with respect to the empirical version of $\delta_{\pi_{\theta_k}}$ that is, the empirical stationary state-action distribution associated with the current policy π_{θ_k}. In practice, the penalty parameter β_k is adjusted by line search. A_t the k-th iteration of TRPO, the policy parameter θ is updated by k+1$\leftarrow argmax_{\theta}$ E[(π_{θ}(a|s)/π_{θ_k}(a|s))$\cdot$Ak(s, a)], subject to KL($\pi_{\theta}(\cdot|s)||$ $\pi_{\theta_k}(\cdot|s)$) $\leq \delta$ where δ is the radius of the trust region. The PPO update can be viewed as a Lagrangian relaxation of the TRPO update with Lagrangian multiplier β_k, which implies their updates are equivalent if β_k is properly chosen. Without loss of generality, we focus on PPO in this paper. Compared with the original versions of PPO and TRPO, we use KL($\pi_{\theta}(\cdot|s)||\pi_{\theta_k}(\cdot|s)$) instead of KL($\pi_{\theta_k(\cdot|s)}||\pi_{\theta_k}(\cdot|s)$). This paper show that such variants also allow us to approximately get the better policy.

2.3 Neural PPO:

For notational simplicity, we denote by ν_k and σ_k the stationary state distribution $\nu_{\pi_{\theta_k}}$ and the stationary state-action distribution $\sigma_{\pi_{\theta_k}}$, respectively. Also, we define an auxiliary distribution $\sigma_k^{\sim}$ over $S \times A$ as $\sigma_k^{\sim}$ = $\nu_k\pi_0$.

2.3.1 Neural Network Parametrization:

Without loss of generality, we assume that (s, a) $\in R^d$ for all s $\in$ S and a $\in$ A We parametrize a function u : S $\times$ A $\rightarrow$ R, e.g., policy π or actionvalue function Q^{π} ,by the following two-layer neural network, which is denoted by NN(α; m), $u_{\alpha}(s,a) = (1/\sqrt{m}) * \Sigma_i^m b_i\sigma([\alpha]_i(s,a))$. Here m is the width of the neural network, $b_i \in 1, 1$ (i $\in$[m]) are the output weights, $\sigma(\cdot)$ is ReLU, and $\alpha = ([\alpha]1, ..., [\alpha]_m) \in$ R^{md} with $[\alpha]_i R^d$ (i $\in$ [m]) are the input weights. We consider the random initialization bi i.i.d. Unif(1, 1), $[\alpha(0)]_i$ i.i.d. N (0, Id/d), for all i $\in$ [m]. We restrict the input weights α to an l2-ball centered at the initialization $\alpha(0)$ by the projection $\Pi_{B^0(R_\alpha)}(\alpha')$= $argmin_{\alpha^0(R_\alpha)}||\alpha\ \ \alpha'||^2$, where $B^0(R_\alpha)$ = α : ||$\alpha - \alpha(0)||^2 \leq R_\alpha$. We only update α. Therefore, we omit the dependency on b_i (i $\in$ [m]) in NN(α; m) and u_{α}(s, a)

2.3.2 Policy Improvement:

We consider the population version of the objective function in PPO: L(θ) = E_{ν_k}[<Q_{ω_k}(s, $cdot$), β_k($\cdot$ | s)> - $\beta_k\cdot$ KL($\pi_{\theta}(\cdot$ | s)|| $\pi_{\theta_k}(\cdot$| s))] where $Q_{\}k}$ is an estimator of $Q^{\pi_{\theta_k}}$, that is, the exact action-value function of π_{θ_k}. In the following, we convert the subproblem max L(θ) of policy improvement into a least-squares subproblem. We consider the energy-based policy π(als) $\propto exp^1 f(s,a)$,which is abbreviated as $\pi \propto exp\tau^1 f$ Here f:S$\times$A$\rightarrow$R is the energy function and τ > 0 is the temperature parameter.

Proposition 3.1. Let $\pi_{k+1} \propto exp\tau_k^1 f_{\theta_k}$ be an energy-based policy. Given an estimator Q_{ω_k} of $Q^{\pi_{\theta_k}}$, the update $\pi_{k+1} \leftarrow argmax_{\pi}$ E_{ν_k}[<Q_{ω_k}(s, $\cdot$), $\pi(\cdot$ | s)>$\beta_k\cdot$KL($\pi(\cdot|s)||\pi_{\theta_k}(\cdot|s)$)] gives $b_{k+1} \propto exp_k^1 Q_{\omega_k} + \tau^1 f_{\theta_k}$. To represent the ideal improved policy k+1 using the energy-based policy $\pi_{\theta_{k+1}}$ $\propto exp\tau_{k+1}^1 f_{\theta_{k+1}}$, ,we solve the subproblem of minimizing the MSE, $\theta_{k+1} \leftarrow argmin_{\theta\in B^0(Rf)}$ $E\sigma_k^{\sim}[(f_{\theta}(s,a)$- $\tau_{k+1}\cdot((\beta_k^1 Q_{\omega_k}(s,a) + \tau^1 f_{\theta_k}(s,a))^2]$

Here we use the neural network parametrization f = NN(θ; m), where θ denotes the input weights and m_f is the width. we use the SGD update to solve this subproblem. θ(t + 1/2) $\leftarrow$ $\theta(t)$ - $\eta\cdot$ $(f_{\theta(t)}(s,a)\tau_{k+1}(\beta_k^1 Q_{\omega_k}\cdot(s,a) + \tau_k^1 f_{\theta_k}(s,a)))$ where (s, a) $\sim \sigma_k^{\sim}$ and θ(t + 1) $\leftarrow Pi_{B^0(R_f)}$(θ(t + 1/2)). Here η is the stepsize.

2.3.3 Policy Evaluation:

To obtain the estimator Q_{ω_k} of Qk, we solve the subproblem of minimizing the MSBE, $omega_k \leftarrow argmin_{\omega \in B^0(R_Q)} E_{\sigma_k}[(Q_\omega(s,a)[T_{\pi_{\theta_k}} Q_\omega](s,a))^2]$ Here the Bellman evaluation operator T^π of a policy π is defined as [T^π Q](s, a)=E[(1 γ) · r(s, a) + $\gamma \cdot Q(s^{'}, a^{'})|s^{'}$ P(· | s, a), $a^{'}$ $\pi(|s^{'})$] We use the neural network parametrization Q_ω = NN(ω; m_Q) where denotes the input weights and m_Q is the width. we use the TD update ω(t + 1/2) ← ω(t) η·(Q_ω(t)(s, a) (1 γ) · r(s, a) $\gamma \cdot Q_\omega$(t)(s$^{'}$, a$^{'}$))$\nabla_\omega Q_\omega$(t)(s, a), where (s, a) $\sim \sigma_k$, $s^{'}$ P(· | s, a), and ω(t + 1) = $\Pi B^0_{(R_Q)}(\omega$(t + 1/2)). Here η is the stepsize.

2.3.4 Neural PPO

Require: MDP (S, A,P, r, γ), penalty parameter β, widths m_f and m_Q, number of SGD and TD iterations T, number of TRPO iterations K, and projection radii $R_f \geq R_Q$ 1: Initialize with uniform policy: $\tau_0 \leftarrow 1$, $f_{\theta_0} \leftarrow 0$, $_{\theta_0}$,$\pi_0 \propto exp\tau_0^{-1} f_{\theta_0}$
2: for k = 0, . . . , K 1 do
3: Set temperature parameter $\tau_{k+1} \leftarrow \beta \cdot \sqrt{k}$/(k + 1) and penalty parameter $\beta_k \leftarrow \beta \cdot \sqrt{k}$
4: Sample $\{(s_t, a_t, a_t^0, s_t^{'}, a_t^{'})\}_{t=1}^T$ with $(s_t, a_t) \sim, a_t^0 \sim \pi_0(|s_t), s_t^{'} \sim P(|s_t, a_t) and a_t^{'} t \sim \pi_{\theta_k}(|s_k^{'})$
5: Solve for Q_{ω_k} = NN(ω_k; m_Q) using the TD update
6: Solve for $f_{\theta_{k+1}}$ = NN(θ_{k+1}; m_f) using the SGD update
7: Update policy: $\pi_{\theta_{k+1}} \propto exp\tau_{k+1}^1 f_{\theta_{k+1}}$
8: end for

3 Theoretical Analysis

3.1 Error of Policy Improvement and Policy Evaluation

We establish the global convergence of PPO relying on characterizing the error from solving the subproblems of policy evaluation and policy improvement. Our analysis relies on the following regularity condition on the boundedness of reward.

3.1.1 Assumption 4.1 (Bounded Reward).

There exists a constant R_{max} > 0 such that R_{max} =sup(s,a)∈ $S \times A$ |r(s, a)| ,which implies |$V^\pi(s)$| R_{max} and |Q^π(s, a)| ≤ R_{max} for any policy π. To ensure the compatibility between the policy and the action-value function, we set m_f = m_Q and use the following random initialization.

3.2 Error Propagation

3.2.1 Definition 4.2.

For any constant R > 0, we define the function class $F_{R,m}$ = $(1/sqrtm) * \sigma_{i=1}^m b_i 1[\alpha(0)]_i^T(s,a) > 0[\alpha]_i^T(s,a) : ||\alpha\alpha(0)||2 \leq R$ where $[(0)]_i$ and b_i (i ∈ [m]) are the random initialization. As m → ∞, F$_{R,m}$$NN(\alpha$(0); m) approximates a subset of the reproducing kernel Hilbert space (RKHS) induced by the kernel K(x, y) = $E_z \sim$ N(0,Id/d)[1$z^T x$ > 0, z^Ty > 0$x^T y$] Such a subset is a ball with radius R in the corresponding H-norm, which is known to be a rich function class.

3.2.2 Assumption 4.3 (Action-Value Function Class)

It holds that $Q^\pi(s,a) \in F_{RQ,mQ}$ for any π. This assumption state that $F_{RQ,mQ}$ is closed under the Bellman evaluation operator T^π,as Q^π is the fixed-point solution of the Bellman equation $T^\pi Q^\pi$ = Q^π.

3.2.3 Assumption 4.4 (Regularity of Stationary Distribution)

There exists a constant c > 0 such that for any vector z ∈ R^d and ζ > 0, it holds almost surely that $E\sigma_\pi[1|z^T(s,a)|\zeta|z]c\zeta/||z||2$ for any π.

3.2.4 Theorem 4.5 (Policy Improvement Error)

Suppose that the above assumptions hold. We set T 64 and the stepsize to be $\eta = T^{1/2}$. Within the k-th iteration of Algorithm Neural PPO, the output f_θ of Policy Improvement via SGD satisfies $E_{init,\sigma_k^\sim}[(s,a)\tau_{k+1}(_k^{-1}Q_{\omega_k}(s,a) + \tau_k^{-1}f_{\theta_k}(s,a))^2] = O(R_f^2T^{1/2} + R_f^{5/2}m_f^{-1/4} + R_f^3m_f^{1/2})$

3.2.5 Theorem 4.6 (Policy Evaluation Error)

Suppose that above assumptions hold. We set $T \geq 64/(1\gamma)^2$ and the stepsize to be $\eta = T^{1/2}$. Within the k-th iteration of Neural PPO the output Q_ω of Policy Evaluation via TD satisfies $E_{init,\sigma_k}[(Q_\omega(s,a)Q^{\pi_{\theta_k}}(s,a))^2] = O(R^2QT^{-1/2} + R_Q^{5/2}m_Q^{1/4} + R_Q^3m^(1/2)_Q)$ We characterize the primal and dual errors of the infinite-dimensional mirror descent corresponding to neural PPO. These errors decay to zero at the rate of $1/sqrt(T)$ when the width $m_f = m_Q$ is sufficiently large, where T is the number of TD and SGD iterations in Neural PPO.

3.3 Global Convergence of Neural PPO

We denote by π^* the optimal policy with ν^* being its stationary state distribution and σ^* being its stationary state-action distribution. π_{k+1} is the ideal improved policy based on Q_{ω_k}, which is an estimator of the exact action-value function $Q_{\pi_{\theta_k}}$. we define the ideal improved policy based on $Q_{\pi_{\theta_k}}$ as
$\pi_{k+1} = argmax_\pi E_{\nu_k}[< Q_{\pi_{\theta_k}}(s,), \pi(,s) > -\beta_k KL(\pi(|s)||\pi_{\theta_k}(|s))]$
We define the following functions which are related to density ratios between policies or stationary distributions,
$\phi_k^* = E_{\sigma_k^\sim}[|d\pi^*/d\pi_0 d\pi_{\theta_k}/d\pi_0|^2]^{1/2}, \psi_k^* = E_{\sigma_k}[|d\sigma^*/d\sigma_k d\nu^*/d\nu_k|^2]^{1/2}$, where $d\pi^*/d\pi_0$, $d\pi^{\theta_k}/d\pi_0$, $d\sigma^*/d\sigma_k$, and $d\nu^*/d\nu_k$ are the Radon-Nikodym derivatives.

3.3.1 Lemma 4.7 (Error Propagation).

Suppose that the policy improvement error in Neural PPO satisfies, $E_{\sigma_k^\sim}[(f_{\theta_{k+1}}(s,a) - \tau_{k+1}(\beta_k^1 Q_{\omega_k}(s,a)\tau_k^1 f_{\theta_k}(s,a))^2] \leq \epsilon_{k+1}$ and the policy evaluation error in Neural PPO satisfies, $E_{\sigma_k}[(Q_{\omega_k}(s,a)Q^{\pi_{\theta_k}}(s,a))^2]\epsilon_k^{'}{}_{k+1}$ and $_{\theta_{k+1}}$ from Neural PPO satisfies, $|E_{\nu_k^*}[< log(\pi_{\theta_{k+1}}(|s)/\pi_{k+1}(|s)), \pi^*(|s)\pi^{\theta_k}(|s) >]| \leq \epsilon_k$ where $\epsilon_k = \tau_{k+1}^1\epsilon_{k+1}\phi_{k+1}^* +_k^1 \epsilon_k^{'}\psi_k^*$

3.3.2 Lemma 4.8 (Stepwise Energy Difference)

$E_{\nu_*}[||\tau_{k+1}^1 f_{\theta_{k+1}}(s,)\tau_k^1 f_{\theta_k}(s,)||^2] \leq 2\epsilon_k^{'} + 2\beta_k^2 M$ where $\epsilon_k^{'} = |A|_{k+1}^2\epsilon_{k+1}^2$ and $M = 2E_{\nu^*}[max_a(Q_{\omega_0}(s,a))^2] + 2R_f^2$, the bounded difference between fk+1 and fk+1 quantified in this Lemma is due to the KL-regularization.

3.4 Global Convergence of Neural PPO

We will use the expected total reward L() = E [V(s)] = E [<Q(s, ·), (· | s)>], where is the stationary state distribution of the optimal policy The following theorem characterizes the global convergence of L(k) towards L().

3.4.1 Theorem 4.9 (Global Rate of Convergence of Neural PPO).

Suppose that the above assumptions hold, For the policy sequence $\pi_{\theta_k}{}_{k=1}^K$ attained by neural PPO, we can get $min_{0\leq k\leq K}L(\pi^*)L(\pi_{\theta_k}) \leq (\beta^2 log|A| + M + \beta^2\Sigma_{k=0}^{K1}(\epsilon_k + \epsilon_k^{'}))/((1\gamma)\beta\sqrt{K})$ $\epsilon_k = \tau_{k+1}^1\epsilon_{k+1}\phi_k^* + \beta_k^1\epsilon_k^{'}\psi_k^*$ and $\epsilon_k^{'} = |A|\tau_{k+1}^2\epsilon_{k+1}^2$, where $\epsilon_{k+1} = O(R_f^2T^{1/2} + R_f^{5/2}m_f^{1/4} + R_f^3m_f^{1/2})$, $\epsilon_k^{'} = O(R^2Q_T^{1/2} + R_Q^{5/2}m_Q^{1/4} + R_Q^3m_Q^{1/2})$. we have $M = 2E_{\nu^*}[max_a(Q_{\omega_0}(s,a))^2] + 2R_f^2$ we consider the infinite-dimensional policy update based on the exact action-value function, that is, $\epsilon_{k+1} = \epsilon_k^{'} = 0$ for any k + 1 [K]. In such an ideal case, by Theorem 4.9, neural PPO globally converges to the optimal policy at the rate of $min_{0\leq k\leq K}L(\pi^*)L(\pi^{\theta_k}) \leq 2sqrt(Mlog|A|)/(1\gamma)K$ with the optimal choice of the penalty parameter $\beta_k = sqrt(MK/log|A|)$.

3.4.2 Corollary 4.10 (Iteration Complexity of Subproblems and Minimum Widths of Neural Networks).

Suppose that the above assumptions hold, Let $m_f = W(K^{6}R_f^{10}\phi_k^{*4}+K^4R_f^{10}|A|^2), m_Q = W(K^2R_Q^{10}\psi_k^{*4})$,and $T = W(K^3R_f^4\phi_k^{*2} + K^2R_f^4|A| + KR_Q^4\psi_k^{*2})$ for any $0 \leq k \leq K$. We have $min_{0\leq k\leq K}L(\pi^*)L(\pi^{\theta_k}) \leq (\beta^2 log|A| + M + O(1)/((1\gamma)\beta sqrt(K))$ This corollary quantifies the minimum width m_f and m_Q and the minimum number of SGD and TD iterations T that ensure the $O(1/sqrt(K))$ rate of convergence. This corollary points out that the errors of policy improvement and policy evaluation play distinct roles in the global convergence of neural PPO. The total error based on Theorem 4.9 is $^1_{k+1}\epsilon_{k+1}\phi_k^* + \beta_k^1\epsilon_k^{'}\psi_k^* + |A|\tau_{k+1}^2\epsilon_{k+1}^2$, where the weight τ_{k+1}^1 of the policy improvement error ϵ_{k+1} is much larger than the weight $\beta-1_k$ of the policy evaluation error $\epsilon_k^{'}$, and $|A|\tau_{k+1}^2\epsilon_{k+1}^2$ is a high-order term when ϵ_{k+1} is sufficiently small. Therefore, the errors of policy improvement is more important than policy evaluation.

3.5 Proof

3.5.1 Lemma 5.1 (Performance Difference)

we have $L(\pi)L(\pi^*) = (1\gamma)1E_{\nu^*}[< Q^\pi(s,), \pi(|s)\pi(|s) >]$. Since the optimal policy maximizes the value function $\nu_\pi(s)$ with respect to π for any $s \in S$, we have $L(\pi^*) = E_{\nu^*}[V_{\pi^*}(s)]E_{\nu^*}[\nu_\pi(s)] = L(\pi)$ for any . As a result, we have $E_{\nu^*}[< Q_\pi(s,), \pi(|s)\pi^*(|s) >]0$, for any π. Under the variational inequality framework, Lemma 5.1 corresponds to the monotonicity of the mapping Q evaluated at and any . Therefore, we regard (5.1) as one-point monotonicity.

3.5.2 Lemma 5.2 (One-Step Descent)

For the ideal improved policy k+1 and the current policy k, we can get that, for any $sS, KL(\pi^*(|s)||\pi^{\theta_{k+1}}(|s)) - KL(\pi^*(|s)||\pi^{\theta_k+}(|s)) \leq< log(\pi_{\theta_{k+1}}(|s)/\pi_{k+1}(|s)), \pi_{\theta_k}(|s)\pi^*(|s) > -\beta_k^1 < Q_{\pi_{\theta_k}}(s,), \pi^*(|s)_{\theta_k}(|s) > 1/2||\pi_{\theta_{k+1}}(|s)_{\theta_k}(|s)||_1^2 - < \tau_{k+1}^1 f_{\theta_{k+1}}(s,)\tau_k^1 f_{\theta_k}(s,), \pi_{\theta_k}(|s)\pi_{\theta_{k+1}}(|s) >$

3.5.3 Proof of Theorem 4.9

Based on Lemmas 5.1 and 5.2, we prove Theorem 4.9 by casting neural PPO as infinite dimensional mirror descent with primal and dual errors, whose impact is characterized in Lemma 4.7. In particular, we employ the 1- pair of primal-dual norms. Taking expectation with respect to $s \sim \nu^*$and invoking Lemmas 4.7 and 5.2, we have $E_{\nu^*}[KL(\pi^*(|s)||\pi_{\theta_{k+1}}(|s))]E_{\nu^*}[KL(\pi^*(|s)||\pi_{\theta_k}(|s))] \leq \epsilon_k\beta_k^1 E_{\nu_*}[< Q_{\pi_{\theta_k}}(s,), \pi^*(|s)\pi_{\theta_k}(|s) >]](1/2)E_{\nu^*}[||\pi_{\theta_{k+1}}(|s)\pi_{\theta_k}(|s)||^2 1] - E\nu^*[< \tau_{k+1}^1 f_{\theta_{k+1}}(s,)\tau_k^1 f_{\theta_k}(s,), \pi_{\theta_k}(|s)\pi_{\theta_{k+1}}(|s) >]$ By Lemma 5.1 and the H¨older's inequality, we further have $E_{\nu^*}[KL(\pi^*(|s)||\pi_{\theta_{k+1}}(|s))]E_{\nu^*}[KL(\pi^*(|s)||\pi_{k+1}(|s))] \leq \epsilon_k(1\gamma)\beta_k^1(L(\pi^*)L(\pi_{\theta_k}))1/2E_{\nu^*}[||\pi_{\theta_{k+1}}(|s)\pi_{\theta_k}(|s)||_1^2] + E_{\nu^*}[||\tau_{k+1}^1 f_{\theta_{k+1}}(s,)\tau_k^1 f_{\theta_k}(s,)||_\infty ||\pi_{\theta_k}(|s)\pi_{\theta_{k+1}}(|s)||1] \leq \epsilon_k(1\gamma)\beta_k^1(L(\pi^*)L(\pi_{\theta_k})) + 1/2E_{\nu^*}[||^1_{k+1}f_{\theta_{k+1}}(s,)\tau_k^1 f_{\theta_k}(s,)||_\infty^2] \leq \epsilon_k(1\gamma)\beta_k^1(L(\pi^*)L(\pi_{\theta_k})) + (\epsilon_k^{'} + \beta_k^2 M)$, where in the second inequality we use $2xyy^2x^2$ and in the last inequality we use Lemma4.8. Rearranging these terms , we have $(1\gamma)\beta_k^1(L(\pi^*)L(\pi_{\theta_k})) \leq E_{\nu^*}[KL(\pi^*(|s)||\pi_{\theta_{k+1}}(|s))]E_{\nu^*}[KL(\pi^*(|s)||\pi^{\theta_k}(|s))] + \beta_k^2 M + \epsilon_k + \epsilon_k^{'}$. Summarizing the k+1 [K], we obtain $\Sigma_{k=0}^{K-1}((1\gamma)\beta^1 k(L(\pi_{\theta_k})L(\pi^*))) \leq E_{\nu^*}[KL(\pi^{**}(|s)||\pi_{\theta_k}(|s))]E_{\nu^*}[KL(\pi^*(|s)||\pi_{\theta_0}(|s))] + M\Sigma_{k=0}^{K-1}(\beta_k^2) + \Sigma_{k=0}^{K-1}(\epsilon_k + \epsilon_k^{'})$ Note that we have (i) sigma $\beta_k^1(L(\pi^*)L(\pi_{\theta_k})) \geq \Sigma_{k=0}^{K-1}(\beta_k^1) min_{0\leq k\leq K}L(\pi^*)L(\pi_{\theta_k})$
(ii) $E_{\nu^*}[KL(\pi^*(|s)||\pi_{\theta_0}(|s))]log|A|$ due to the uniform initialization of policy
(iii)) the KL-divergence is nonnegative. Therefore, we obtain $min0kKL(\pi^*)L(\pi_{\theta_k}) \leq (log|A| + MSigma_{k=0}^{K-1}\beta^{-2} + k + \Sigma_{k=0}^{K-1}(\epsilon_k + \epsilon_k^{'}))/(1\gamma)\Sigma(\beta_k^1)$
Setting the penalty parameter $\beta_k = \beta K$, we have sigma $(\beta_k^1) = \beta^1\sqrt{K}$ and $\Sigma\beta_k^2 = \beta^2$, which together with (5.4) concludes the proof of Theorem 4.9.

4 Conclusion

This paper is a milestone for bridging the gap between theory and practice. It provide a proof of the convergence of PPO in the mirror descent iterations, when we use neural network parametrization to approximate policy and action-value function. However, I think that it is hard to prepare such a ideal environment to simulate this experiment. The further extension of this topic may be the proof of the convergence of PPO in simplification of real case.

References