APPENDIX:

A CONVERGENCE FOR SOFT POLICY ITERATION

Lemma 1. *Consider the soft Bellman backup operator* $\mathcal{T}^{\pi,\tau}Q(s,a) = r(s,a) + \gamma\mathbb{E}_{s'\sim\mathcal{P}(\cdot|s,a)}(V(s'))$ *where* $V(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}\left[Q(s,a) - \tau\log\frac{\pi(a|s)}{\beta(a|s)}\right]$*, then the sequence* Q*,* $\mathcal{T}^{\pi,\tau}Q$*,* $(\mathcal{T}^{\pi,\tau})^2Q$*,* $\cdots$*,* $(\mathcal{T}^{\pi,\tau})^kQ$ *will converge to the soft value of* π *as* $k\to\infty$*.*

Proof. We prove the soft Bellman backup operator is a contraction. If we apply $\mathcal{T}^\pi$ to two different value functions Q and Q', the max norm distance $\|Q-Q'\| = \max_{s,a}|Q(s,a)-Q'(s,a)|$ shrinks.

$$\begin{aligned}
\|\mathcal{T}^\pi Q - \mathcal{T}^\pi Q'\| &= \|r(s_t,a_t) + \gamma\mathbb{E}_{s_{t+1}\sim p}(V(s_{t+1})) - r(s_t,a_t) - \gamma\mathbb{E}_{s_{t+1}\sim p}(V'(s_{t+1}))\| \quad (4)\\
&= \gamma\|\mathbb{E}_{s_{t+1}\sim p}(V(s_{t+1})) - \mathbb{E}_{s_{t+1}\sim p}(V'(s_{t+1}))\|\\
&= \gamma\|\mathbb{E}_{s_{t+1}\sim p}\mathbb{E}_{a_{t+1}\sim\pi(\cdot|s_{t+1})}\left[Q(s_{t+1},a_{t+1}) - \tau\log\frac{\pi(a_{t+1}|s_{t+1})}{\beta(a_{t+1}|s_{t+1})}\right] -\\
&\quad - \mathbb{E}_{s_{t+1}\sim p}\mathbb{E}_{a_{t+1}\sim\pi(\cdot|s_{t+1})}\left[Q'(s_{t+1},a_{t+1}) - \tau\log\frac{\pi(a_{t+1}|s_{t+1})}{\beta(a_{t+1}|s_{t+1})}\right]\|\\
&= \gamma\|\mathbb{E}_{s_{t+1}\sim p}\mathbb{E}_{a_{t+1}\sim\pi(\cdot|s_{t+1})}\left[Q(s_{t+1},a_{t+1}) - Q'(s_{t+1},a_{t+1})\right]\|\\
&\le \gamma\max_{s_{t+1},a_{t+1}}|Q(s_{t+1},a_{t+1}) - Q'(s_{t+1},a_{t+1})|\\
&= \gamma\|Q-Q'\|
\end{aligned}$$

Therefore, the sequence $\mathcal{T}^{\pi,\tau}Q$, $(\mathcal{T}^{\pi,\tau})^2Q$, $\cdots$, $(\mathcal{T}^{\pi,\tau})^kQ$ only has one fixed point. Consider the soft Q value of policy π, $\tilde{Q}^{\pi,\tau}(s_t,a_t) = r_t + \mathbb{E}_{s_{t+1}\sim\mathcal{P}(\cdot|s_{t+1},a_{t+1}),a_{t+1}\sim\pi(\cdot|s_{t+1}),s_{t+l+1}\sim\mathcal{P}(\cdot|s_{t+l},a_{t+l})}\left[\sum_{l=1}^\infty\gamma^l(r(s_{t+l},a_{t+1}) - \tau\log\frac{\pi(a_{t+l}|s_{t+l})}{\beta(a_{t+l}|s_{t+l})})\right]$, we have $\mathcal{T}^{\pi,\tau}\tilde{Q}^{\pi,\tau}(s_t,a_t) = \tilde{Q}^{\pi,\tau}(s_t,a_t)$.

$$\|(\mathcal{T}^{\pi,\tau})^kQ - \tilde{Q}^{\pi,\tau}\| \le \gamma\|(\mathcal{T}^{\pi,\tau})^{k-1} - \tilde{Q}^{\pi,\tau}\| \le \gamma^2\|(\mathcal{T}^{\pi,\tau})^{k-2}Q - \tilde{Q}^{\pi,\tau}\| \le \cdots \le \gamma^k\|Q - \tilde{Q}^{\pi,\tau}\| \to 0$$

Thus, $(\mathcal{T}^{\pi,\tau})^kQ$ will converge to the fixed point $\tilde{Q}^{\pi,\tau}$.

□

Lemma 2. *let* $\pi_{old}\in\Pi$ *and* $\pi_{new}(\cdot|s) = \arg\max_{\pi\in\Pi}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}\left(\tilde{Q}^{\pi_{old},\tau}(s,a) - \tau\log\frac{\pi(a|s)}{\beta(a|s)}\right)\right]$*. Then* $\tilde{Q}^{\pi_{new},\tau}(s,a)\ge\tilde{Q}^{\pi_{old},\tau}(s,a)$ *for all* (s,a)*.*

Proof. Due to the definition of π_{new}, for any state s_t, we have
$\mathbb{E}_{a\sim\pi_{new}(\cdot|s_t)}\tilde{Q}^{\pi_{old},\tau}(s_t,a) - \tau KL(\pi_{new}(\cdot|s_t)|\beta(\cdot|s_t)) \ge \mathbb{E}_{a\sim\pi_{old}(\cdot|s_t)}\tilde{Q}^{\pi_{old},\tau}(s_t,a) - \tau KL(\pi_{old}(\cdot|s_t)|\beta(\cdot|s_t))$

$$\begin{aligned}
\tilde{Q}^{\pi_{old},\tau}(s_t,a_t) &= r_t + \gamma\mathbb{E}_{s_{t+1}\sim p}\left[\tilde{V}^{\pi_{old},\tau}(s_{t+1})\right] \quad (5)\\
&= r_t + \mathbb{E}_{s_{t+1}\sim p}\left[\mathbb{E}_{a_{t+1}\sim\pi_{old}(\cdot|s_{t+1})}\tilde{Q}^{\pi_{old},\tau} - \tau KL(\pi_{old}(\cdot|s_{t+1})|\beta(\cdot|s_{t+1}))\right]\\
&\le r_t + \mathbb{E}_{s_{t+1}\sim p}\left[\mathbb{E}_{a_{t+1}\sim\pi_{new}(\cdot|s_{t+1})}\tilde{Q}^{\pi_{old},\tau}(s_{t+1},a_{t+1}) - \tau KL(\pi_{new}(\cdot|s_{t+1})|\beta(\cdot|s_{t+1}))\right]\\
&\cdots\\
&\cdots\\
&\le \tilde{Q}^{\pi_{new},\tau}(s_t,a_t)
\end{aligned}$$

□

Theorem 2. *Repeated application of soft policy evaluation and soft policy improvement converges to a policy* π^*_τ *such that* $\tilde{Q}^{\pi^*_\tau,\tau}(s,a)\ge\tilde{Q}^{\pi,\tau}(s,a)$ *for any* $\pi\in\Pi$ *and* $(s,a)\in\mathcal{S}\times\mathcal{A}$*.*

Proof. Let π_i be the policy at iteration i. The sequence Q^{π_i} is monotonically increasing. Since Q^{π} is bounded above (because both reward and KL divergence are bounded), the sequence converges to some π^*. We will still need to show that π^* is indeed optimal. At convergence, it must be case that:

$\pi^*(\cdot|s) = \arg\max_{\pi\in\Pi}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}\left(\tilde{Q}^{\pi^*,\tau}(s,a) - \tau\log\frac{\pi(a|s)}{\beta(a|s)}\right)\right]$

So for any $\pi \in \Pi$, we have

$\mathbb{E}_{a\sim\pi^*(\cdot|s_t)}\tilde{Q}^{\pi^*,\tau}(s_t,a) - \tau KL(\pi^*(\cdot|s_t)|\beta(\cdot|s_t)) \geq \mathbb{E}_{a\sim\pi(\cdot|s_t)}\tilde{Q}^{\pi^*,\tau}(s_t,a) - \tau KL(\pi(\cdot|s_t)|\beta(\cdot|s_t))$

$$\begin{aligned}
\tilde{Q}^{\pi^*,\tau}(s_t,a_t) &= r_t + \gamma\mathbb{E}_{s_{t+1}\sim p}\left[\tilde{V}^{\pi^*_\tau,\tau}(s_{t+1})\right] \\
&= r_t + \mathbb{E}_{s_{t+1}\sim p}\left[\mathbb{E}_{a_{t+1}\sim\pi^*(\cdot|s_{t+1})}\tilde{Q}^{\pi^*_\tau,\tau} - \tau KL(\pi^*(\cdot|s_{t+1})|\beta(\cdot|s_{t+1}))\right] \\
&\geq r_t + \mathbb{E}_{s_{t+1}\sim p}\left[\mathbb{E}_{a_{t+1}\sim\pi(\cdot|s_{t+1})}\tilde{Q}^{\pi^*,\tau}(s_{t+1},a_{t+1}) - \tau KL(\pi(\cdot|s_{t+1})|\beta(\cdot|s_{t+1}))\right] \\
&\cdots \\
&\cdots \\
&\geq \tilde{Q}^{\pi,\tau}(s_t,a_t)
\end{aligned} \tag{6}$$

We denote this optimal policy π^* as π^*_τ in the following. Consider the optimization problem with hard constraint:

$$\begin{aligned}
\max_{\pi(a_1|s),\pi(a_2|s),\cdots,\pi(a_{|A|}|s)} \quad & \sum_{i=1}^{|A|}\pi(a_i|s)\tilde{Q}^{\pi^*_\tau,\tau}(s,a) - \tau\sum_{i=1}^{|A|}\pi(a_i|s)\log\frac{\pi(a_i|s)}{\beta(a_i|s)} \\
\text{s.t.} \quad & \sum_{i=1}^{|A|}\pi(a_i|s) = 1
\end{aligned}$$

Due to KKT condition, let $\tilde{Q}^{\pi^*_\tau,\tau}(s,a) - \tau\log\frac{\pi(a_i|s)}{\beta(a_i|s)} - \tau - \lambda = 0$. We have $\pi^*_\tau(a|s) = \frac{\exp\left(\frac{\tilde{Q}^{\pi^*_\tau,\tau}(s,a)}{\tau}\right)\beta(a|s)}{\sum_{a_i}\exp\left(\frac{\tilde{Q}^{\pi^*_\tau,\tau}(s,a_i)}{\tau}\right)\beta(a_i|s)}$ □

Theorem 3. *Let $\pi^*_\tau(a|s)$ be the optimal policy from soft policy iteration with fixed temperature τ. We have $\pi^*_\tau(a|s) \propto \exp\left(\frac{\tilde{Q}^{\pi^*_\tau,\tau}(s,a)}{\tau}\right)\beta(a|s)$. As $\tau \to 0$, $\pi^*_\tau(a|s)$ will take the optimal action a^* with optimal Q value for state s.*

Assume there exists a set of optimal actions $X(s)$ for state s. For each action in $X(s)$, its soft Q value is the optimal for state s. To simplify the notation, in the following proof, we use Q^*_τ to replace $\tilde{Q}^{\pi^*_\tau,\tau}$.

Assume $a_j \in X(s)$, then $\pi^*_\tau(a_j|s) = \frac{1}{\sum_{a_i\in X(s)}\frac{\beta(a_i|s)}{\beta(a_j|s)} + \sum_{a_i\notin X(s)}\exp\left(\frac{Q^*_\tau(s,a_i)-Q^*_\tau(s,a_j)}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)}}$.

For the second term in the denominator, we show it converges to 0 as $\tau \to 0$.

$0 \leq \sum_{a_i\notin X(s)}\exp\left(\frac{Q^*_\tau(s,a_i)-Q^*_\tau(s,a_j)}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)} \leq \sum_{a_i\notin X(s)}\exp\left(\frac{Q^*_\tau(s,a_{sub})-Q^*_\tau(s,a_{opt})}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)}$

$0 \leq \sum_{a_i\notin X(s)}\exp\left(\frac{Q^*_\tau(s,a_i)-Q^*_\tau(s,a_j)}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)} \leq \exp\left(\frac{Q^*_\tau(s,a_{sub})-Q^*_\tau(s,a_{opt})}{\tau}\right)\sum_{a_i\notin X(s)}\frac{\beta(a_i|s)}{\beta(a_j|s)}$

As $\tau \to 0$, $\exp\left(\frac{Q^*_\tau(s,a_{sub})-Q^*_\tau(s,a_{opt})}{\tau}\right) \to 0$ and $\sum_{a_i\notin X(s)}\frac{\beta(a_i|s)}{\beta(a_j|s)}$ is a constant.

Thus, as $\tau \to 0$, $\pi^*_\tau(a_j|s) \to \frac{1}{\sum_{a_i\in X(s)}\frac{\beta(a_i|s)}{\beta(a_j|s)}} = \frac{\beta(a_j|s)}{\sum_{a_i\in X(s)}\beta(a_i|s)}$ if $a_j \in X(s)$.

Assume $a_j \notin X(s)$, then $\pi^*_\tau(a_j|s) = \frac{1}{\sum_{a_i\in X(s)}\exp\left(\frac{Q^*_\tau(s,a_{opt})-Q^*_\tau(s,a_j)}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)} + \sum_{a_i\notin X(s)}\exp\left(\frac{Q^*_\tau(s,a_i)-Q^*_\tau(s,a_j)}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)}}$.

Obviously, $\exp\left(\frac{Q^*_\tau(s,a_{opt})-Q^*_\tau(s,a_j)}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)} \rightarrow +\infty$ as $\tau \rightarrow 0$ and $\exp\left(\frac{Q^*_\tau(s,a_i)-Q^*_\tau(s,a_j)}{\tau}\right)\frac{\beta(a_i|s)}{\beta(a_j|s)} > 0$.

Thus, as $\tau \rightarrow 0$, $\pi^*_\tau(a_j|s) \rightarrow 0$ if $a_j \notin X(s)$.

B CONVERGENCE RATE OF KL REGULARIZED POLICY GRADIENT

Recall our definitions:

$$V^{\pi}(\rho) = \mathbb{E}_{s_0\sim\rho, a_t\sim\pi(\cdot|s_t), s_{t+1}\sim\mathcal{P}(\cdot|s_t,a_t)}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\right]. \tag{7}$$

Mei et al. [26] proved that policy gradient with a softmax parameterization and true gradient optimizes $V^{\pi}(\rho)$ at a $\mathcal{O}(1/t)$ convergence rate, while the entropy regularized objective $V^{\pi}(\rho) + \tau\mathbb{H}(\rho,\pi)$ with $\mathbb{H}(\rho,\pi) = \mathbb{E}_{s_0\sim\rho, a_t\sim\pi(\cdot|s_t), s_{t+1}\sim\mathcal{P}(\cdot|s_t,a_t)}\left[\sum_{t=0}^{\infty}(-\gamma^t\log\pi(a_t|s_t))\right]$ enjoys a significantly faster linear convergence rate $\mathcal{O}(e^{-t})$.

With a behavior policy β and the "temperature" $\tau > 0$ that determines the strength of the regularization, we define the value of the policy π with the KL divergence regularization as

$$\tilde{V}^{\pi,\tau}(\rho) = V^{\pi}(\rho) - \tau D_{KL}(\pi||\beta) = \mathbb{E}_{s_0\sim\rho, a_t\sim\pi(\cdot|s_t), s_{t+1}\sim\mathcal{P}(\cdot|s_t,a_t)}\left[\sum_{t=0}^{\infty}\gamma^t\left(r(s_t,a_t) - \tau\log\frac{\pi(a_t|s_t)}{\beta(a_t|s_t)}\right)\right] \tag{8}$$

Similarly we prove policy gradient optimizes $\tilde{V}^{\pi,\tau}(\rho)$ at a $\mathcal{O}(e^{-t})$ convergence rate. Here we explain the sketch of the proof. $\tilde{V}^{\pi,\tau}(\rho)$ can be re-written as $\hat{V}^{\pi,\tau}(\rho) + \tau\mathbb{H}(\rho,\pi)$ where

$$\hat{V}^{\pi,\tau}(\rho) = \mathbb{E}_{s_0\sim\rho, a_t\sim\pi(\cdot|s_t), s_{t+1}\sim\mathcal{P}(\cdot|s_t,a_t)}\left[\sum_{t=0}^{\infty}\gamma^t(r(s_t,a_t) + \tau\log\beta(a_t|s_t))\right]. \tag{9}$$

$$\mathbb{H}(\rho,\pi) = \mathbb{E}_{s_0\sim\rho, a_t\sim\pi(\cdot|s_t), s_{t+1}\sim\mathcal{P}(\cdot|s_t,a_t)}\left[\sum_{t=0}^{\infty}-\gamma^t\log\pi(a_t|s_t)\right] \tag{10}$$

$\hat{V}^{\pi,\tau}(\rho)$ in eq.(8) is the same as $V^{\pi}(\rho)$ in eq.(7) if we replace $r(s_t,a_t)$ with $r(s_t,a_t)+\tau\log\beta(a_t|s_t)$. Mei et al. [26] assume that $r(s_t,a_t) \in [0,1]$. Here we assume $r(s_t,a_t)+\tau\log\beta(a_t|s_t) \in [-M,M]$ with a constant M, effectively requiring the behavior policy to span the entire state-action space. We can then adapt the proof in [26] for the new objective $\tilde{V}^{\pi,\tau}(\rho)$. We conclude that in t iterations, the learned policy π_{θ_t} is approaching the optimal policy π^*_τ for the new objective $\tilde{V}^{\pi,\tau}(\rho)$, satisfying $\tilde{V}^{\pi^*_\tau,\tau}(\rho) - \tilde{V}^{\pi_{\theta_t},\tau}(\rho) \leq \frac{1}{\exp\{C_\tau\Omega(1)t\}}\frac{M+\tau\log A}{(1-\gamma)^2}\|\frac{1}{\mu}\|_\infty$, where $C_\tau = \frac{(1-\gamma)^4}{(8M/\tau+4+8\log A)\cdot S}\cdot\min_s\mu(s)\cdot\|\frac{d_\mu^{\pi^*_\tau}}{\mu}\|_\infty^{-1}$ and μ is the initial state distribution used in the policy optimization algorithm.

To simplify the notation, in the following proofs, we use $\tilde{V}^{\pi}(\rho) = \tilde{V}^{\pi,\tau}(\rho)$ and $\hat{V}^{\pi}(\rho) = \hat{V}^{\pi,\tau}(\rho)$.

Assumption 1 (Bounded Reward and Behavior Probability). *$r(s,a) + \tau\log\beta(a|s) \in [-M,M]$, $\forall(s,a)$ where M is a positive constant.*

Definition 1 (Smoothness). *A function $f:\Theta\to\mathbb{R}$ is λ-smooth if for all $\theta,\theta'\in\Theta$,*

$$\left|f(\theta') - f(\theta) - \left\langle\frac{\partial f(\theta)}{\partial\theta}, \theta'-\theta\right\rangle\right| \leq \frac{\lambda}{2}\|\theta'-\theta\|_2^2$$

Lemma 3 (Monotone Increasing). *Assume a function $f:\Theta\to\mathbb{R}$ is λ-smooth, and it is updated with $\theta_{t+1} = \theta_t + \eta\frac{\partial f(\theta)}{\partial\theta}$ and $\eta = \frac{2}{\lambda}$, then the function $f(\theta)$ is monotone increasing.*

$$f(\theta_{t+1}) - f(\theta_t) - \left\langle\frac{\partial f(\theta)}{\partial\theta}, \theta_{t+1}-\theta_t\right\rangle \geq -\frac{\lambda}{2}\|\theta_{t+1}-\theta_t\|_2^2$$

$$f(\theta_{t+1}) - f(\theta_t) \geq \left\langle\frac{\partial f(\theta)}{\partial\theta}, \eta\frac{\partial f(\theta)}{\partial\theta}\right\rangle - \frac{\lambda\eta^2}{2}\|\frac{\partial f(\theta)}{\partial\theta}\|_2^2$$

$$= (1-\frac{\lambda\eta}{2})\eta\|\frac{\partial f(\theta)}{\partial\theta}\|_2^2 = 0$$

Definition 2. *Given a vector $\theta\in\mathbb{R}^{[K]}$ and the probability distribution $\pi_\theta = softmax(\theta) = \frac{exp\theta}{\sum_k exp\theta(k)}$, then $H(\pi_\theta)$ is the Jacobian of the $\theta\to\pi_\theta$ map: $\left(\frac{d\pi_\theta}{d\theta}\right)^T = H(\pi_\theta) = diag(\pi_\theta) - \pi_\theta\pi_\theta^T \in \mathbb{R}^{K\times K}$.*

Lemma 4 (Smoothness). *$\hat{V}^{\pi_\theta}(\rho)$ is $\frac{8M}{(1-\gamma)^3}$-smooth.*

Denote $\theta_\alpha = \theta + \alpha u$, where $\alpha \in \mathbb{R}$ and $u \in \mathbb{R}^{SA}$. For any $s \in S$,

$$\begin{aligned}
\sum_a \left| \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \alpha}|_{\alpha=0} \right| &= \sum_a \left| \left\langle \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \theta_\alpha}|_{\alpha=0}, \frac{\partial \theta_\alpha}{\partial \alpha} \right\rangle \right| \\
&= \sum_a \left| \left\langle \frac{\partial \pi_\theta(a|s)}{\partial \theta}, u \right\rangle \right| \\
&= \sum_a \left| \left\langle \frac{\partial \pi_\theta(a|s)}{\partial \theta(s,\cdot)}, u(s,\cdot) \right\rangle \right| \left(\text{Because } \frac{\partial \pi_\theta(a|s)}{\partial \theta(s',\cdot)} = 0, \forall s' \neq s \right) \\
&= \sum_a \pi_\theta(a|s) \cdot \left| u(s,a) - \pi_\theta(\cdot|s)^T u(s,\cdot) \right| \text{ (Because softmax parameterization)} \\
&\leq \max_a |u(s,a)| + \left| \pi_\theta(\cdot|s)^T u(s,\cdot) \right| \\
&\leq 2\|u\|_2 \qquad (11)
\end{aligned}$$

$$\begin{aligned}
\sum_a \left| \frac{\partial^2 \pi_{\theta_\alpha}(a|s)}{\partial \alpha^2}|_{\alpha=0} \right| &= \sum_a \left| \left\langle \frac{\partial}{\partial \theta_\alpha} \{ \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \alpha} \}|_{\alpha=0}, \frac{\partial \theta_\alpha}{\partial \alpha} \right\rangle \right| \\
&= \sum_a \left| \left\langle \frac{\partial^2 \pi_{\theta_\alpha}(a|s)}{\partial \theta_\alpha^2}|_{\alpha=0} \frac{\partial \theta_\alpha}{\partial \alpha}, \frac{\partial \theta_\alpha}{\partial \alpha} \right\rangle \right| \\
&= \sum_a \left| \left\langle \frac{\partial^2 \pi_\theta(a|s)}{\partial \theta^2(s,\cdot)} u(s,\cdot), u(s,\cdot) \right\rangle \right| \\
&= \sum_a \left| \sum_{i=1}^A \sum_{j=1}^A S_{i,j} u(s,i) u(s,j) \right| (S_{i,j} \text{ is the i,j-th element of } \frac{\partial^2 \pi_\theta(a|s)}{\partial \theta^2(s,\cdot)}) \\
&= \sum_a \pi_\theta(a|s) | u(s,a)^2 - 2u(s,a) \pi_\theta(\cdot|s)^T u(s,\cdot) \\
&- \pi_\theta(\cdot|s)^T (u(s,\cdot) \odot u(s,\cdot)) + 2(\pi_\theta(\cdot|s)^T u(s,\cdot))^2 | \\
&\leq \max_a \{ u(s,a)^2 + 2 \left| u(s,a) \pi_\theta(\cdot|s)^T u(s,\cdot) \right| \} \\
&+ \pi_\theta(\cdot|s)^T (u(s,\cdot) \odot u(s,\cdot)) + 2(\pi_\theta(\cdot|s)^T u(s,\cdot))^2 \\
&\leq \|u(s,\cdot)\|_2^2 + 2\|u(s,\cdot)\|_2^2 + \|u(s,\cdot)\|_2^2 + 2\|u(s,\cdot)\|_2^2 \\
&\leq 6\|u\|_2^2 \qquad (12)
\end{aligned}$$

Define $P(\alpha) \in \mathcal{R}^{S \times S}$, $\forall (s,s')$, $[P(\alpha)]_{(s,s')} = \sum_a \pi_{\theta_\alpha}(a|s) \mathcal{P}(s'|s,a)$ For any vector $x \in \mathbb{R}^S$, the l_∞ norm is

$$\begin{aligned}
\left\| \frac{\partial P(\alpha)}{\partial \alpha}|_{\alpha=0} x \right\|_\infty &= \max_s \left| \left[\frac{\partial P(\alpha)}{\partial \alpha}|_{\alpha=0} x \right]_{(s)} \right| \\
&= \max_s \left| \sum_{s'} \left[\frac{\partial P(\alpha)}{\partial \alpha}|_{\alpha=0} \right]_{(s,s')} x(s') \right| \\
&= \max_s \left| \sum_{s'} \sum_a \left[\frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \alpha}|_{\alpha=0} \right] P(s'|s,a) x(s') \right| \\
&\leq \max_s \sum_a \sum_{s'} \mathcal{P}(s'|s,a) \cdot \left| \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \alpha}|_{\alpha=0} \right| \cdot \|x\|_\infty \\
&= \max_s \sum_a \left| \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \alpha}|_{\alpha=0} \right| \cdot \|x\|_\infty \\
&\leq 2 \cdot \|u\|_2 \cdot \|x\|_\infty \qquad (13)
\end{aligned}$$

$$\begin{aligned}
\left\| \frac{\partial^2 P(\alpha)}{\partial \alpha^2}|_{\alpha=0} x \right\|_\infty &= \max_s \left| \left[\frac{\partial^2 P(\alpha)}{\partial \alpha^2}|_{\alpha=0} x \right]_{(s)} \right| \\
&= \max_s \left| \sum_{s'} \left[\frac{\partial^2 P(\alpha)}{\partial \alpha^2}|_{\alpha=0} \right]_{(s,s')} x(s') \right| \\
&= \max_s \left| \sum_{s'} \sum_a \left[\frac{\partial^2 \pi_{\theta_\alpha}(a|s)}{\partial \alpha^2}|_{\alpha=0} \right] P(s'|s,a) x(s') \right| \\
&\leq \max_s \sum_a \sum_{s'} \mathcal{P}(s'|s,a) \cdot \left| \frac{\partial^2 \pi_{\theta_\alpha}(a|s)}{\partial \alpha^2}|_{\alpha=0} \right| \cdot \|x\|_\infty \\
&= \max_s \sum_a \left| \frac{\partial^2 \pi_{\theta_\alpha}(a|s)}{\partial \alpha^2}|_{\alpha=0} \right| \cdot \|x\|_\infty \leq 6 \cdot \|u\|_2^2 \cdot \|x\|_\infty \qquad (14)
\end{aligned}$$

Consider the KL regularized value function of π_{θ_α}.
$\hat{V}^{\pi_{\theta_\alpha}}(s) = \sum_a \pi_{\theta_\alpha}(a|s)(r(s,a) + \tau \log \beta(a|s)) + \gamma \sum_a \pi_{\theta_\alpha}(a|s) \sum_{s'} \mathcal{P}(s'|s,a) \hat{V}^{\theta_\alpha}(s') = e_s^T M(\alpha) r_{\theta_\alpha}$
where e_s is an indicator vector for the starting state s, $M(\alpha) = (\mathbf{Id} - P(\alpha))^{-1} = \sum_{t=0}^{\infty} \gamma^t P(\alpha)^t$, and $r_{\theta_\alpha}(s) = \sum_a \pi_{\theta_\alpha}(a|s)(r(s,a) + \tau \log \beta(a|s))$.

Taking derivative with respect to α:

$$\frac{\partial \hat{V}^{\pi_{\theta_\alpha}}(s)}{\partial \alpha} = \gamma e_s^T M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) r_{\theta_\alpha} + e_s^T M(\alpha) \frac{\partial r_{\theta_\alpha}}{\partial \alpha} \qquad (15)$$

Taking second derivative with respect to α:

$$\begin{aligned}
\frac{\partial^2 \hat{V}^{\pi_{\theta_\alpha}}(s)}{\partial \alpha^2} &= 2\gamma^2 e_s^T M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) r_{\theta_\alpha} + \gamma^2 e_s^T M(\alpha) \frac{\partial^2 P(\alpha)}{\partial \alpha^2} M(\alpha) r_{\theta_\alpha} \\
&\quad + 2\gamma e_s^T M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) \frac{\partial r_{\theta_\alpha}}{\partial \alpha} + e_s^T M(\alpha) \frac{\partial^2 r_{\theta_\alpha}}{\partial \alpha^2} \qquad (16) \\
\|M(\alpha)x\|_\infty &= \max_i \left| [M(\alpha)]_{i,:}^T x \right| \\
&\leq \max_i \|[M(\alpha)]_{i,:}\|_1 \cdot \|x\|_\infty \\
&= \frac{1}{1-\gamma} \cdot \|x\|_\infty (\text{Because } \mathbf{1} = \frac{1}{1-\gamma} \cdot (\mathbf{Id} - \gamma P(\alpha))\mathbf{1}, M(\alpha)\mathbf{1} = \frac{1}{1-\gamma}\mathbf{1}) \qquad (17) \\
\|r_{\theta_\alpha}\|_\infty &= \max_s |r_{\theta_\alpha}(s)| \\
&= \max_s \left| \sum_a \pi_{\theta_\alpha}(a|s)(r(s,a) + \tau \log \beta(a|s)) \right| \\
&\leq M (\text{Because of the range of } r(s,a) + \tau \log \beta(a|s)) \qquad (18)
\end{aligned}$$

$$
\begin{aligned}
\left\| \frac{\partial r_{\theta_\alpha}}{\partial \alpha} \right\|_\infty &= \max_s \left| \frac{\partial r_{\theta_\alpha}(s)}{\partial \alpha} \right| \\
&= \max_s \left| \left(\frac{\partial r_{\theta_\alpha}(s)}{\partial \theta_\alpha} \right)^T \frac{\partial \theta_\alpha}{\partial \alpha} \right| \\
&= \max_s \left| \left(\frac{\partial \{\pi_{\theta_\alpha}(\cdot|s)^T (r(s,\cdot) + \tau \log \beta(\cdot|s))\}}{\partial \theta_\alpha} \right)^T u(s,\cdot) \right| \\
&= \max_s \left| (H(\pi_{\theta_\alpha}(\cdot|s))(r(s,\cdot) + \tau \log \beta(\cdot|s)))^T u(s,\cdot) \right| \\
&\leq \max_s \|H(\pi_{\theta_\alpha}(\cdot|s))(r(s,\cdot) + \tau \log \beta(\cdot|s))\|_1 \cdot \|u(s,\cdot)\|_\infty \\
&= \max_s (\sum_a \pi_{\theta_\alpha}(a|s) \cdot \left| r(s,a) + \tau \log \beta(a|s) - \pi_{\theta_\alpha}(\cdot|s)^T (r(s,\cdot) + \tau \log \beta(\cdot|s)) \right|) \|u(s,\cdot)\|_\infty \\
&\leq \max_s \max_a \left| r(s,a) + \tau \log \beta(a|s) - \pi_{\theta_\alpha}(\cdot|s)^T (r(s,\cdot) + \tau \log \beta(\cdot|s)) \right| \cdot \|u(s,\cdot)\|_\infty \\
&\leq 2M \|u\|_2 \qquad (19)
\end{aligned}
$$

$$
\begin{aligned}
\left\| \frac{\partial^2 r_{\theta_\alpha}}{\partial \alpha^2} \right\|_\infty &= \max_s \left| \frac{\partial^2 r_{\theta_\alpha}(s)}{\partial \alpha^2} \right| \\
&= \max_s \left| \left(\frac{\partial}{\partial \theta_\alpha} \left\{ \frac{\partial r_{\theta_\alpha}(s)}{\partial \alpha} \right\} \right)^T \frac{\partial \theta_\alpha}{\partial \alpha} \right| \\
&= \max_s \left| \left(\frac{\partial^2 r_{\theta_\alpha}(s)}{\partial \theta_\alpha^2} \frac{\partial \theta_\alpha}{\partial \alpha} \right)^T \frac{\partial \theta_\alpha}{\partial \alpha} \right| \\
&= \max_s \left| u(s,\cdot)^T \frac{\partial^2 \{\pi_{\theta_\alpha}(\cdot|s)^T (r(s,\cdot) + \tau \log \beta(\cdot|s))\}}{\partial \theta_\alpha(s,\cdot)^2} u(s,\cdot) \right| \\
&= \max_s \left| \sum_{i=1}^{A} \sum_{j=1}^{A} S_{i,j} u(s,i) u(s,j) \right| (S_{i,j} \text{ is the i,j-th element of the second derivative}) \\
&= \max_s | \sum_i \pi_{\theta_\alpha}(s,\cdot,i)(r(s,i) + \tau \log \beta(i|s) - \pi_{\theta_\alpha}(\cdot|s)^T (r(s,\cdot) + \tau \log \beta(\cdot|s))) u(s,i)^2 \\
&- 2 \sum_i \pi_{\theta_\alpha}(i|s)(r(s,i) + \tau \log \beta(i|s) - \pi_{\theta_\alpha}(\cdot|s)^T (r(s,\cdot) + \tau \log \beta(\cdot|s))) u(s,i) \sum_j \pi_{\theta_\alpha}(j|s) u(s,j)| \\
&= \max_s |(H(\pi_{\theta_\alpha})(r(s,\cdot) + \tau \log \beta(\cdot|s)))^T (u(s,\cdot) \odot u(s,\cdot)) \\
&- 2(H(\pi_{\theta_\alpha})(r(s,\cdot) + \tau \log \beta(\cdot|s)))^T u(s,\cdot)(\pi_{\theta_\alpha}^T u(s,\cdot))| \\
&\leq \max_s (\|H(\pi_{\theta_\alpha})(r(s,\cdot) + \tau \log \beta(\cdot|s)))\|_\infty \cdot \|u(s,\cdot) \odot u(s,\cdot)\|_1 \\
&+ 2\|H(\pi_{\theta_\alpha})(r(s,\cdot) + \tau \log \beta(\cdot|s)))\|_1 \cdot \|u(s,\cdot)\|_\infty \cdot \|\pi_{\theta_\alpha}\|_1 \cdot \|u(s,\cdot)\|_\infty) \\
&\leq \max_s (\|H(\pi_{\theta_\alpha})(r(s,\cdot) + \tau \log \beta(\cdot|s)))\|_\infty \cdot \|u(s,\cdot)\|_2^2 (\text{Because } \|u(s,\cdot) \odot u(s,\cdot)\|_1 = \|u(s,\cdot)\|_2^2) \\
&+ 2\|H(\pi_{\theta_\alpha})(r(s,\cdot) + \tau \log \beta(\cdot|s)))\|_1 \cdot \|u(s,\cdot)\|_2^2) (\text{Because } \|\pi_{\theta_\alpha}\|_1 = 1, \|u(s,\cdot)\|_\infty \leq \|u(s,\cdot)\|_2) \\
&\leq \max_s (\max_i \left| H_{i,:}(\pi_{\theta_\alpha})^T (r(s,\cdot) + \tau \log \beta(\cdot|s))) \right| \|u(s,\cdot)\|_2^2 + 2 \cdot 2M \cdot \|u(s,\cdot)\|_2^2) \\
&\leq \max_s \max_i \|H_{i,:}(\pi_{\theta_\alpha})\|_1 \cdot \|r(s,\cdot) + \tau \log \beta(\cdot|s))\|_\infty \cdot \|u(s,\cdot)\|_2^2 + 4M \|u(s,\cdot)\|_2^2 \\
&\leq \max_s \max_i (\pi_{\theta_\alpha}(i|s) - \pi_{\theta_\alpha}(i|s)^2 + \pi_{\theta_\alpha}(i|s)(1 - \pi_{\theta_\alpha}(i|s))) \cdot M \cdot \|u(s,\cdot)\|_2^2 + 4M \|u(s,\cdot)\|_2^2 \\
&\leq \frac{1}{2} M \|u(s,\cdot)\|_2^2 + 4M \|u(s,\cdot)\|_2^2 \leq 6M \|u(s,\cdot)\|_2^2 \qquad (20)
\end{aligned}
$$

For the first term of Eq. 16, according to Eq. 17,13,18

$$\begin{aligned}\left|e_s^T M(\alpha)\frac{\partial P(\alpha)}{\partial \alpha}M(\alpha)\frac{\partial P(\alpha)}{\partial \alpha}M(\alpha)r_{\theta_\alpha}|_{\alpha=0}\right| &\leq \left|M(\alpha)\frac{\partial P(\alpha)}{\partial \alpha}M(\alpha)\frac{\partial P(\alpha)}{\partial \alpha}M(\alpha)r_{\theta_\alpha}|_{\alpha=0}\right|_\infty \\ &\leq \frac{1}{1-\gamma}\cdot 2\cdot\|u\|_2\cdot\frac{1}{1-\gamma}\cdot 2\cdot\|u\|_2\cdot\frac{1}{1-\gamma}\cdot M \\ &= \frac{4M}{(1-\gamma)^3}\|u\|_2^2 \end{aligned} \tag{21}$$

For the second term of Eq. 16, according to Eq. 17,14,18,

$$\begin{aligned}\left|e_s^T M(\alpha)\frac{\partial^2 P(\alpha)}{\partial \alpha^2}M(\alpha)r_{\theta_\alpha}|_{\alpha=0}\right| &\leq \left|M(\alpha)\frac{\partial^2 P(\alpha)}{\partial \alpha^2}M(\alpha)r_{\theta_\alpha}|_{\alpha=0}\right|_\infty \\ &\leq \frac{1}{1-\gamma}\left|\frac{\partial^2 P(\alpha)}{\partial \alpha^2}M(\alpha)r_{\theta_\alpha}|_{\alpha=0}\right|_\infty \\ &\leq \frac{6\|u\|_2^2}{1-\gamma}|M(\alpha)r_{\theta_\alpha}|_{\alpha=0}|_\infty \\ &\leq \frac{6\|u\|_2^2}{(1-\gamma)^2}|r_{\theta_\alpha}|_{\alpha=0}|_\infty \leq \frac{6M}{(1-\gamma)^2}\|u\|_2^2 \end{aligned} \tag{22}$$

For the third term of Eq. 16,

$$\begin{aligned}\left|e_s^T M(\alpha)\frac{\partial P(\alpha)}{\partial \alpha}M(\alpha)\frac{\partial r_{\theta_\alpha}}{\partial \alpha}|_{\alpha=0}\right| &\leq \left|M(\alpha)\frac{\partial P(\alpha)}{\partial \alpha}M(\alpha)\frac{\partial r_{\theta_\alpha}}{\partial \alpha}|_{\alpha=0}\right|_\infty \\ &\leq \frac{1}{1-\gamma}\left|\frac{\partial P(\alpha)}{\partial \alpha}M(\alpha)\frac{\partial r_{\theta_\alpha}}{\partial \alpha}|_{\alpha=0}\right|_\infty \\ &\leq \frac{2\|u\|_2}{1-\gamma}\left|M(\alpha)\frac{\partial r_{\theta_\alpha}}{\partial \alpha}|_{\alpha=0}\right|_\infty \\ &\leq \frac{2\|u\|_2}{(1-\gamma)^2}\left|\frac{\partial r_{\theta_\alpha}}{\partial \alpha}|_{\alpha=0}\right|_\infty \\ &\leq \frac{4M}{(1-\gamma)^2}\|u\|_2^2 \end{aligned} \tag{23}$$

For the last term of Eq. 16,

$$\begin{aligned}\left|e_s^T M(\alpha)\frac{\partial^2 r_{\theta_\alpha}}{\partial \alpha^2}\right| &\leq \left|M(\alpha)\frac{\partial^2 r_{\theta_\alpha}}{\partial \alpha^2}\right|_\infty \\ &\leq \frac{1}{1-\gamma}\left|\frac{\partial^2 r_{\theta_\alpha}}{\partial \alpha^2}\right|_\infty \leq \frac{6M}{1-\gamma}\|u\|_2^2 \end{aligned} \tag{24}$$

Combining the four terms:

$$\begin{aligned}\left|\frac{\partial^2 \hat{V}^{\pi_{\theta_\alpha}}(s)}{\partial \alpha^2}|_{\alpha=0}\right| &\leq \left(2\gamma^2\cdot\frac{4M}{(1-\gamma)^3}+\gamma\frac{6M}{(1-\gamma)^2}+2\gamma\frac{4M}{(1-\gamma)^2}+\frac{6M}{1-\gamma}\right)\|u\|_2^2 \\ &\leq \frac{8M}{(1-\gamma)^3}\|u\|_2^2 \end{aligned} \tag{25}$$

which implies that for all $y \in \mathbb{R}^{SA}$ and θ:

$$\begin{aligned}
\left| y^T \frac{\partial^2 \hat{V}^{\pi_\theta}(s)}{\partial \theta^2} y \right| &= \left| \left(\frac{y}{\|y\|_2} \right)^T \frac{\partial^2 \hat{V}^{\pi_\theta}(s)}{\partial \theta^2} \left(\frac{y}{\|y\|_2} \right) \right| \|y\|_2^2 \\
&\leq \max_{\|u\|_2=1} \left| \left\langle \frac{\partial^2 \hat{V}^{\pi_\theta}(s)}{\partial \theta^2} u, u \right\rangle \right| \|y\|_2^2 \\
&= \max_{\|u\|_2=1} \left| \left\langle \frac{\partial^2 \hat{V}^{\pi_{\theta_\alpha}}(s)}{\partial \theta_\alpha^2}|_{\alpha=0} \frac{\partial \theta_\alpha}{\partial \alpha}, \frac{\partial \theta_\alpha}{\partial \alpha} \right\rangle \right| \|y\|_2^2 \\
&= \max_{\|u\|_2=1} \left| \frac{\partial^2 \hat{V}^{\pi_{\theta_\alpha}}(s)}{\partial \alpha^2}|_{\alpha=0} \right| \|y\|_2^2 \leq \frac{8M}{(1-\gamma)^3} \|y\|_2^2
\end{aligned} \tag{26}$$

Denote $\theta_\xi = \theta + \xi(\theta' - \theta)$, where $\xi \in [0, 1]$, according to Taylor's theorem,

$$\begin{aligned}
\left| \hat{V}^{\pi_{\theta'}}(s) - \hat{V}^{\pi_\theta}(s) - \left\langle \frac{\partial \hat{V}^{\pi_\theta}(s)}{\partial \theta}, \theta' - \theta \right\rangle \right| &= \frac{1}{2} \left| (\theta' - \theta)^T \frac{\partial^2 \hat{V}^{\pi_{\theta_\xi}}(s)}{\partial \theta_\xi^2} (\theta' - \theta) \right| \\
&\leq \frac{4M}{(1-\gamma)} \|\theta' - \theta\|_2^2
\end{aligned} \tag{27}$$

Lemma 5 (Smoothness)**.** $\mathbb{H}(\rho, \pi_\theta)$ *is* $\frac{4+8\log A}{(1-\gamma)^3}$*-smooth, where* $A = |\mathcal{A}|$ *is the total number of actions.*

Lemma 6 (Soft Policy Gradient)**.** *It holds that*

$$\frac{\partial \tilde{V}^{\pi_\theta}(\mu)}{\partial \theta(s,a)} = \frac{1}{1-\gamma} \cdot d_\mu^{\pi_\theta}(s) \cdot \pi_\theta(a|s) \cdot \tilde{A}^{\pi_\theta}(s,a) \tag{28}$$

$$\frac{\partial \tilde{V}^{\pi_\theta}(\mu)}{\partial \theta(s,\cdot)} = \frac{1}{1-\gamma} \cdot d_\mu^{\pi_\theta}(s) \cdot H(\pi_\theta(a|s)) \cdot \left[\tilde{Q}^{\pi_\theta}(s,a) - \tau \log \pi_\theta(\cdot|s) \right] = \frac{1}{1-\gamma} \cdot d_\mu^{\pi_\theta}(s) \cdot H(\pi_\theta(a|s)) \cdot \left[\tilde{Q}^{\pi_\theta}(s,a) - \tau \theta(s,\cdot) \right] \tag{29}$$

where $\tilde{A}^{\pi_\theta}(s,a)$ *is the 'soft' advantage function defined as:* $\tilde{A}^{\pi_\theta}(s,a) = \tilde{Q}^{\pi_\theta}(s,a) - \tau \log \pi_\theta(a|s) - \tilde{V}^{\pi_\theta}(s)$, $\tilde{Q}^{\pi_\theta}(s,a) = r(s,a) + \tau \log \beta(a|s) + \gamma \sum_s \mathcal{P}(s'|s,a) \tilde{V}^{\pi_\theta}(s')$

Lemma 7. $d_\mu^{\pi_\theta}(s) > (1-\gamma)\mu(s)$

$$\begin{aligned}
d_\mu^{\pi_\theta}(s) &= \mathbb{E}_{s_0 \sim \mu} \left[(1-\gamma) \sum_{t=0}^{\infty} \gamma^t P(s_t = s | s_0, \pi_\theta, \mathcal{P}) \right] \\
&\geq \mathbb{E}_{s_0 \sim \mu} \left[(1-\gamma) P(s_t = s | s_0) \right] \\
&= (1-\gamma)\mu(s)
\end{aligned}$$

Lemma 8 (Non-uniform Lojasiewicz)**.** *Suppose initial state distribution in the policy gradient algorithm* $\mu(s) > 0$ *for all state* $s \in \mathcal{S}$

$$\left\| \frac{\partial \tilde{V}^{\pi_\theta}(\mu)}{\partial \theta} \right\|_2 \geq \frac{\sqrt{2\tau}}{\sqrt{S}} \cdot \min_s \sqrt{\mu(s)} \cdot \min_{s,a} \pi_\theta(a|s) \cdot \left\| \frac{d_\rho^{\pi_\tau^*}}{d_\mu^{\pi_\theta}} \right\|_\infty^{-\frac{1}{2}} \cdot \left[\tilde{V}^{\pi_\tau^*}(\rho) - \tilde{V}^{\pi_\theta}(\rho) \right]^{\frac{1}{2}}$$

Lemma 9. *Using the algorithm with soft policy gradient, we have* $\inf_{t \geq 1} \min_{s,a} \pi_{\theta_t}(a|s) > 0$

The augmented value function $\tilde{V}^{\pi_\theta}(\rho)$ is monotonically increasing following the gradient update with proper η due to smoothness.

$\tilde{V}^{\pi_\theta}(\rho)$ is upper bounded as:

$$\begin{aligned}
\tilde{V}^{\pi_{\theta_t}}(\rho) &= \frac{1}{1-\gamma} \sum_s d_\rho^{\pi_{\theta_t}}(s) \left[\sum_a \pi_{\theta_t}(a|s) \cdot (r(s,a) + \tau \log \beta(a|s) - \tau \log \pi_{\theta_t}(a|s)) \right] \\
&\leq \frac{1}{1-\gamma} \sum_s d_\rho^{\pi_{\theta_t}}(s) \left(M + \tau \log A \right) \text{(Because } -\sum_a \pi_{\theta_t}(a|s) \log \pi_{\theta_t}(a|s) \leq \log A) \\
&\leq \frac{M + \tau \log A}{1-\gamma}
\end{aligned} \tag{30}$$

$\tilde{V}^{\pi_\theta}(\rho)$ is lower bounded as:

$$\begin{aligned}\tilde{V}^{\pi_{\theta_t}}(\rho) &= \frac{1}{1-\gamma}\sum_s d_\rho^{\pi_{\theta_t}}(s)\left[\sum_a \pi_{\theta_t}(a|s)\cdot(r(s,a)+\tau\log\beta(a|s)-\tau\log\pi_{\theta_t}(a|s))\right] \\ &\geq \frac{1}{1-\gamma}\sum_s d_\rho^{\pi_{\theta_t}}(s)(-M)(\text{Because entropy function is positive}) \\ &\geq \frac{-M}{1-\gamma} \end{aligned} \tag{31}$$

$\tilde{Q}^{\pi_\theta}(s,a)$ is lower bounded as:

$$\begin{aligned}\tilde{Q}^{\pi_{\theta_t}}(s,a) &= r(s,a)+\tau\log\beta(a|s)+\gamma\sum_{s'}\mathcal{P}(s'|s,a)\tilde{V}^{\pi_{\theta_t}}(s') \\ &\geq -M-\frac{\gamma}{1-\gamma}M \geq \frac{-M}{1-\gamma}\end{aligned} \tag{32}$$

According to monotone convergence theorem, $\tilde{V}^{\pi_\theta}(\rho)$ converges to a finite value, $\pi_{\theta_t}(a|s)\to\pi_{\theta_\infty}(a|s)$. For any state $s\in\mathcal{S}$, define the following sets: $\mathcal{A}_0(s)=\{a:\pi_{\theta_\infty}(a|s)=0\}$, $\mathcal{A}_+(s)=\{a:\pi_{\theta_\infty}(a|s)>0\}$. We prove that $\mathcal{A}_0(s)=\emptyset$ by contradiction.

Suppose that $\exists s\in\mathcal{S}$,such that $\mathcal{A}_0(s)$ is non-empty. For any $a_0\in\mathcal{A}_0(s)$, we have $\pi^{\theta_t}(a_0|s)\to\pi^{\theta_\infty}(a_0|s)=0$, which implies $-\log\pi^{\theta_t}(a_0|s)\to\infty$. There exists $t_0>0$, such that $\forall t\geq t_0$, $-\log\pi^{\theta_t}(a_0|s)\geq\frac{2M+\tau\log A}{\tau(1-\gamma)}$.

According to Lemma 6, $\forall t\geq t_0$:

$$\begin{aligned}\frac{\partial\tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial\theta(s,a_0)} &= \frac{1}{1-\gamma}\cdot d_\mu^{\pi_{\theta_t}}(s)\cdot\pi_{\theta_t}(a_0|s)\cdot\tilde{A}^{\pi_{\theta_t}}(s,a_0) \\ &= \frac{1}{1-\gamma}\cdot d_\mu^{\pi_{\theta_t}}(s)\cdot\pi_\theta(a_0|s)\cdot\left[\tilde{Q}^{\pi_{\theta_t}}(s,a_0)-\tau\log\pi_{\theta_t}(a_0|s)-\tilde{V}^{\pi_{\theta_t}}(s)\right] \\ &\geq \frac{1}{1-\gamma}\cdot d_\mu^{\pi_{\theta_t}}(s)\cdot\pi_\theta(a_0|s)\cdot\left[\frac{-M}{1-\gamma}-\tau\log\pi_{\theta_t}(a_0|s)-\tilde{V}^{\pi_{\theta_t}}(s)\right] \\ &\geq \frac{1}{1-\gamma}\cdot d_\mu^{\pi_{\theta_t}}(s)\cdot\pi_\theta(a_0|s)\cdot\left[\frac{-M}{1-\gamma}+\tau\frac{2M+\tau\log A}{\tau(1-\gamma)}-\frac{M+\tau\log A}{1-\gamma}\right] \\ &\geq 0\end{aligned} \tag{33}$$

This means $\theta_t(s,a_0)$ is always increasing $\forall t\geq t_0$, which implies that $\theta_\infty(s,a_0)$ is lower bounds by a constant c, and thus $\exp\{\theta_\infty(s,a_0)\}\geq e^c>0$. According to $\pi_{\theta_\infty}(a_0|s)=\frac{\exp\{\theta_\infty(s,a_0)\}}{\sum_a\exp\{\theta_\infty(s,a)\}}=0$, we have $\sum_a\exp\{\theta_\infty(s,a)\}=\infty$. On the other hand, for any $a_+\in\mathcal{A}_+(s)$, according to $\pi_{\theta_\infty}(a_+|s)=\frac{\exp\{\theta_\infty(s,a_+)\}}{\sum_a\exp\{\theta_\infty(s,a)\}}>0$, we have $\exp\{\theta_\infty(s,a_+)\}=\infty,\forall a_+\in\mathcal{A}_+(s)$, which implies $\sum_{a_+\in\mathcal{A}_+(s)}\theta_\infty(s,a_+)=\infty$. Note that $\forall t$,the summation of logit incremental over all actions is zero.

$$\begin{aligned}\sum_a\frac{\partial\tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial\theta_t(s,a)} &= \sum_{a_0\in\mathcal{A}_0(s)}\frac{\partial\tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial\theta_t(s,a_0)}+\sum_{a_+\in\mathcal{A}_+(s)}\frac{\partial\tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial\theta_t(s,a_+)} \\ &= \frac{1}{1-\gamma}\cdot d_\mu^{\pi_{\theta_t}}(s)\cdot\sum_a\pi_{\theta_t}(a|s)\tilde{A}^{\pi_{\theta_t}}(s,a)=0\end{aligned} \tag{34}$$

$\forall t\geq t_0$, $\sum_{a_0\in\mathcal{A}_0(s)}\frac{\partial\tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial\theta_t(s,a_0)}\geq 0$, then $\sum_{a_+\in\mathcal{A}_+(s)}\frac{\partial\tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial\theta_t(s,a_+)}\leq 0$. So $\sum_{a_+\in\mathcal{A}_+(s)}\theta_\infty(s,a_+)$ will always decrease for all large enough $t>t_0$. This is contradiction with $\sum_{a_+\in\mathcal{A}_+(s)}\theta_\infty(s,a_+)=\infty$. To this point, we have shown $\mathcal{A}_0(s)=\emptyset$ for any state s.

At the convergence point $\pi_{\theta_\infty}(\cdot|s)$, the gradient is zero. $\frac{\partial\tilde{V}^{\pi_{\theta_\infty}}(\mu)}{\partial\theta_\infty(s,\cdot)}=\frac{1}{1-\gamma}\cdot d_\mu^{\pi_{\theta_\infty}}(s)\cdot H(\pi_{\theta_\infty}(\cdot|s))\left[\tilde{Q}^{\pi_{\theta_\infty}}(s,\cdot)-\tau\log\pi_{\theta_\infty}(\cdot|s)\right]=\mathbf{0}$. Because of Lemma 7, $d_\mu^{\pi_{\theta_\infty}}(s)>0$, for all state s. Therefore $H(\pi_{\theta_\infty}(\cdot|s))\left[\tilde{Q}^{\pi_{\theta_\infty}}(s,\cdot)-\tau\log\pi_{\theta_\infty}(\cdot|s)\right]=\mathbf{0}$. $H(\pi_{\theta_\infty}(\cdot|s))$ has eigenvalue 0 with multiplicity 1, and the corresponding eigenvector is $c\cdot\mathbf{1}$ for some constant $c\in\mathbb{R}$. Therefore,

$\tilde{Q}^{\pi_{\theta_\infty}}(s,\cdot) - \tau \log \pi_{\theta_\infty}(\cdot|s) = c \cdot \mathbf{1}$, which is equivalent to $\pi_{\theta_\infty}(\cdot|s) = \text{softmax}(\tilde{Q}^{\pi_{\theta_\infty}}(s,\cdot)/\tau)$. Because $\tau \in \Omega(1) > 0$, $\frac{-M}{1-\gamma} \leq \tilde{Q}^{\pi_{\theta_\infty}}(s,a) \leq \frac{M+\tau \log A}{1-\gamma}$, we have $\pi_{\theta_\infty}(a|s) \in \Omega(1)$.

Since $\pi^{\theta_t}(a|s) \to \pi^{\theta_\infty}(a|s) > 0$, there exists $t_0 > 0$ such that $\forall t \geq t_0$, $0.9\pi^{\theta_\infty}(a|s) \leq \pi^{\theta_t}(a|s) \leq 1.1\pi^{\theta_\infty}(a|s)$, which means $\inf_{t \geq t_0} \min_{s,a} \pi_{\theta_t}(a|s) \in \Omega(1)$.
$\inf_{t\geq 1} \min_{s,a} \pi_{\theta_t}(a|s) = \min\{\inf_{1\leq t\leq t_0} \min_{s,a} \pi_{\theta_t}(a|s), \inf_{t\geq t_0} \min_{s,a} \pi_{\theta_t}(a|s)\} = \min\{\Omega(1), \Omega(1)\} \in \Omega(1)$

Theorem 4. *Suppose $\mu_(s) > 0$ for all state s. Using entropy regularized softmax policy gradient algorithm with $\eta = \frac{(1-\gamma)^3}{(8M+\tau(4+8\log A))}$ and $\pi_{\theta_1}(a|s) \in \Omega(1)$, $\forall(s,a)$,*

$$\tilde{V}^{\pi^*_\tau}(\rho) - \tilde{V}^{\pi_\theta}(\rho) \leq \frac{\|1/\mu\|_\infty}{\exp\{C_\tau \cdot \Omega(1) \cdot t\}} \cdot \frac{M + \tau \log A}{(1-\gamma)^2}$$

for all $t > 0$, where $C_\tau, \Omega(1) > 0$ are independent with t.

According to the soft sub-optimality lemma,

$$\begin{aligned}
\tilde{V}^{\pi^*_\tau}(\rho) - \tilde{V}^{\pi_{\theta_t}}(\rho) &= \frac{1}{1-\gamma} \sum_s \left[d_\rho^{\pi_{\theta_t}}(s) \cdot \tau \cdot D_{KL}(\pi_{\theta_t}(\cdot|s) \| \pi^*_\tau(\cdot|s)) \right] \\
&= \frac{1}{1-\gamma} \sum_s \frac{d_\rho^{\pi_{\theta_t}}(s)}{d_\mu^{\pi_{\theta_t}}(s)} \left[d_\mu^{\pi_{\theta_t}}(s) \cdot \tau \cdot D_{KL}(\pi_{\theta_t}(\cdot|s) \| \pi^*_\tau(\cdot|s)) \right] \\
&\leq \frac{1}{1-\gamma} \sum_s \frac{1}{(1-\gamma)\mu(s)} \left[d_\mu^{\pi_{\theta_t}}(s) \cdot \tau \cdot D_{KL}(\pi_{\theta_t}(\cdot|s) \| \pi^*_\tau(\cdot|s)) \right] \\
&\leq \frac{1}{(1-\gamma)^2} \| \frac{1}{\mu} \|_\infty \sum_s \left[d_\mu^{\pi_{\theta_t}}(s) \cdot \tau \cdot D_{KL}(\pi_{\theta_t}(\cdot|s) \| \pi^*_\tau(\cdot|s)) \right] \\
&\leq \frac{1}{1-\gamma} \| \frac{1}{\mu} \|_\infty \left[\tilde{V}^{\pi^*_\tau}(\mu) - \tilde{V}^{\pi_{\theta_t}}(\mu) \right] \qquad (35)
\end{aligned}$$

Because of the property of smoothness, $\tilde{V}^{\pi_\theta}(\mu) = \hat{V}^{\pi_\theta}(\mu) + \tau \mathbb{H}(\mu, \pi_\theta)$ is λ-smooth with $\lambda = \frac{8M+\tau(4+8\log A)}{(1-\gamma)^3}$.

Denote $\tilde{\delta}_t = \tilde{V}^{\pi^*_\tau}(\mu) - \tilde{V}^{\pi_{\theta_t}}(\mu)$,
$\tilde{\delta}_{t-1} - \tilde{\delta}_t = \tilde{V}^{\pi_{\theta_t}}(\mu) - \tilde{V}^{\pi_{\theta_{t+1}}}(\mu)$

$$\begin{aligned}
&\leq -\left\langle \frac{\partial \tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial \theta_t}, \theta_{t+1} - \theta_t \right\rangle + \frac{4M + \tau(2 + 4\log A)}{(1-\gamma)^3} \|\theta_{t+1} - \theta_t\|_2^2 \\
&= \left(-\eta + \frac{4M + \tau(2+4\log A)}{(1-\gamma)^3} \cdot \eta \right) \left\| \frac{\partial \tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial \theta_t} \right\|_2^2 \\
&= -\frac{(1-\gamma)^3}{16M + \tau(8 + 16\log A)} \left\| \frac{\partial \tilde{V}^{\pi_{\theta_t}}(\mu)}{\partial \theta_t} \right\|_2^2 (\text{Because } \eta = \frac{(1-\gamma)^3}{8M + \tau(4+8\log A)}) \\
&\leq -\frac{(1-\gamma)^3}{16M + \tau(8 + 16\log A)} \cdot \frac{2\tau}{S} \cdot \min_s \mu(s) \cdot \left[\min_{s,a} \pi_{\theta_t}(a|s) \right]^2 \cdot \left\| \frac{d_\mu^{\pi^*_\tau}}{d_\mu^{\pi_{\theta_t}}} \right\|_\infty^{-1} \cdot \left[\tilde{V}^{\pi^*_\tau}(\mu) - \tilde{V}^{\pi_{\theta_t}}(\mu) \right] \\
&\leq -\frac{(1-\gamma)^3}{(8M/\tau + 4 + 8\log A) \cdot S} \cdot \min_s \mu(s) \left[\min_{s,a} \pi_{\theta_t}(a|s) \right]^2 \cdot \left\| \frac{d_\mu^{\pi^*_\tau}}{(1-\gamma)\mu} \right\|_\infty^{-1} \cdot \tilde{\delta}_t (\text{Because } d_\mu^{\pi_{\theta_t}}(s) \geq (1-\gamma)\mu(s)) \\
&\leq -\frac{(1-\gamma)^4}{(8M/\tau + 4 + 8\log A) \cdot S} \cdot \min_s \mu(s) \cdot \left[\inf_{t\geq 1} \min_{s,a} \pi_{\theta_t}(a|s) \right]^2 \cdot \left\| \frac{d_\mu^{\pi^*_\tau}}{\mu} \right\|_\infty^{-1} \cdot \tilde{\delta}_t \qquad (36)
\end{aligned}$$

$\inf_{t\geq 1}\min_{s,a}\pi_{\theta_t}(a|s)\in\Omega(1)$ is independent with t.

$$\begin{aligned}
\tilde{\delta}_t &\leq \left[1-\frac{(1-\gamma)^4}{(8M/\tau+4+8\log A)\cdot S}\cdot\min_s\mu(s)\cdot\left[\inf_{t\geq 1}\min_{s,a}\pi_{\theta_t}(a|s)\right]^2\cdot\left\|\frac{d_\mu^{\pi_\tau^*}}{\mu}\right\|_\infty^{-1}\right]\tilde{\delta}_{t-1}\\
&\leq \exp\left\{-\frac{(1-\gamma)^4}{(8M/\tau+4+8\log A)\cdot S}\cdot\min_s\mu(s)\cdot\left[\inf_{t\geq 1}\min_{s,a}\pi_{\theta_t}(a|s)\right]^2\cdot\left\|\frac{d_\mu^{\pi_\tau^*}}{\mu}\right\|_\infty^{-1}\right\}\tilde{\delta}_{t-1}\\
&\leq \exp\left\{-\frac{(1-\gamma)^4}{(8M/\tau+4+8\log A)\cdot S}\cdot\min_s\mu(s)\cdot\left[\inf_{t\geq 1}\min_{s,a}\pi_{\theta_t}(a|s)\right]^2\cdot\left\|\frac{d_\mu^{\pi_\tau^*}}{\mu}\right\|_\infty^{-1}(t-1)\right\}\tilde{\delta}_1\\
&\leq \exp\left\{-\frac{(1-\gamma)^4}{(8M/\tau+4+8\log A)\cdot S}\cdot\min_s\mu(s)\cdot\left[\inf_{t\geq 1}\min_{s,a}\pi_{\theta_t}(a|s)\right]^2\cdot\left\|\frac{d_\mu^{\pi_\tau^*}}{\mu}\right\|_\infty^{-1}(t-1)\right\}\frac{M+\tau\log A}{1-\gamma}
\end{aligned}\tag{37}$$

Thus

$$\tilde{V}^{\pi_\tau^*}(\mu)-\tilde{V}^{\pi_{\theta_t}}(\mu)\leq \exp\left\{-\frac{(1-\gamma)^4}{(8M/\tau+4+8\log A)\cdot S}\cdot\min_s\mu(s)\cdot\left[\inf_{t\geq 1}\min_{s,a}\pi_{\theta_t}(a|s)\right]^2\cdot\left\|\frac{d_\mu^{\pi_\tau^*}}{\mu}\right\|_\infty^{-1}(t-1)\right\}\frac{M+\tau\log A}{1-\gamma}$$

$$\begin{aligned}
\tilde{V}^{\pi_\tau^*}(\rho)-\tilde{V}^{\pi_{\theta_t}}(\rho) &\leq \frac{1}{1-\gamma}\|\frac{1}{\mu}\|_\infty\left[\tilde{V}^{\pi_\tau^*}(\mu)-\tilde{V}^{\pi_{\theta_t}}(\mu)\right]\\
&\leq \frac{\|1/\mu\|_\infty}{\exp\{C_\tau\cdot[\inf_{t\geq 1}\min_{s,a}\pi_{\theta_t}(a|s)]^2\cdot t\}}\cdot\frac{M+\tau\log A}{(1-\gamma)^2}
\end{aligned}\tag{38}$$

where $C_\tau=\frac{(1-\gamma)^4}{(8M/\tau+4+8\log A)\cdot S}\cdot\min_s\mu(s)\cdot\left\|\frac{d_\mu^{\pi_\tau^*}}{\mu}\right\|_\infty^{-1}$

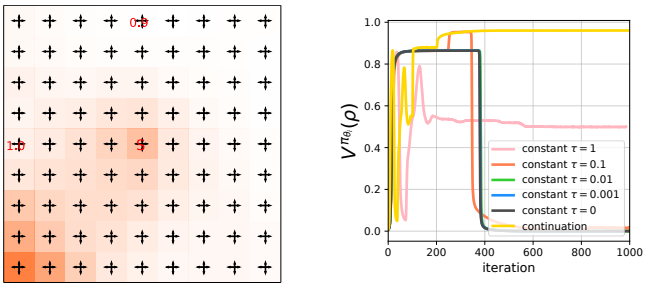

Figure 5: **Left:** visualization of the behavior policy and the dataset collected from the behavior policy. **Right:** learning curve of the value of the parameterized policy π_{θ_i} at each iteration i.

C Experimental Details

C.1 Grid World

We design the grid world of size 9×9. The action space includes actions: up, down, left, right. The reward is 0 at most states, while there are positive rewards 0.9 and 1 at two terminal states. In each episode, the agent starts from the state in the center annotated with 'S', walks around the environment, and terminates the episode only when the agent visits any of the four terminal states (yellow squares at the edge of the grid work). As explained in Section 3.3, we study the performance of soft policy iteration algorithm with different value of τ. The behavior policy is mediocre and tends to move left and down with higher probability at each state. We collect 10000 transitions in the domain according to the behavior policy . We visualize the behavior policy in Fig. 5. In each state, the length of the arrow is proportional to the probability of taking the actions in four directions. We also visualize the visitation count of each state in the collected dataset. The darker color means more visitation. We can see the behavior policy mostly move around the left bottom corner.

We consider optimizing $\tilde{V}^{\pi,\tau}(\rho)$ with different value of τ. It could be constant value in $\{1, 0.1, 0.01, 0.001, 0\}$. For the continuation method, we initially set the value of τ as 1, and decay it with $\tau \leftarrow 0.1\tau$ at every 100 iteration. We search the hyper-parameter of learning rate in $\{5, 1, 0.5, 0.1, 0.05, 0.01, 0.005, 0.001\}$. The learning curves in Fig. 5 verify that the continuation method has better performance than the baselines with a fixed constraint. Visualization of the learned policies for different methods is in Fig. 2. With a constant $\tau = 1$, the learned policy performs similarly to the behavior policy. With a constant $\tau = 0.001$, the learned policy crashes because the value estimate is noisy. With decaying value of τ, the error in Q value estimate is reduced and the relaxing constraint allows the agent to deviate much from the behavior policy.

C.2 Mujoco

As mentioned in the main paper, the architecture of the critic network and the policy network is kept the same in the baselines and our method. The critic network consists of 4 independent Q–networks, which share the same feedforward network architecture. The state and action are concatenated and then passed through 2 hidden layers of size 400 and 300. The policy network is a simple linear network, with the input and output corresponding to the state and the action respectively. ReLU is used as the activation function in all hidden layers. We use 2 separate Adam optimizers for policy and critic network learning and the learning rates are set to 0.001. For all the baselines, we do a grid search of their algorithm–specific hyper–parameters for each game and each dataset. This parameter is fixed in 5 independent runs.

C.2.1 Continuous BCQ

For continuous BCQ, we mostly follow their official implementation[2] and hyper–parameter settings. BCQ uses a perturbation model to adjust the action output with an added residual, which is in the range of $[-\Phi, \Phi]$. We do a grid search of the max perturbation Φ over $\{0.1, 0.2, 0.3\}$.

[2] https://github.com/sfujim/BCQ

C.2.2 BEAR

For BEAR, we also follow the official implementation[3]. In BEAR, a threshold ϵ is used to constrain the MMD distance between the unknown behaviour policy β and the target policy π. It is set to be 0.05 in the original experiments. We instead do a grid search of ϵ over $\{0.05, 0.1, 0.2\}$. BEAR also uses a VAE network to generate actor with distribution similar as the unknown behavior policy. We keep the same architecture as their implementation for VAE and use an Adam optimizer with learning rate 0.001

C.2.3 ABM+SVG

For ABM, we use the batch size of 100, target network update period 1. The KL divergence between the learned policy and the prior policy is constrained by ϵ in ABM, which is set to 0.2 for SVG in the original work. We do a grid search of ϵ over $\{0.05, 0.1, 0.2\}$. For the additional prior policy network, we also use an Adam optimizer with learning rate 0.001.

C.2.4 CRR

In CRR training, we use the batch size of 128, target network update period 1. We use 4 samples to compute the advantage as the original paper does. For fair comparison, we keep the critic architecture the same as others, instead of the distributional one. We focus on the 'mean exp' variant of CRR, because it performs well in CRR paper, while another variant 'max binary' is roughly equivalent to ABM+SVG, which we have run as a separate baseline. To decide on the appropriate temperature in the scalar function f, we swept β over $\{0.01, 0.1, 1, 10\}$, where β is fixed as 1 in the original paper.

C.2.5 CQL

For CQL, we mostly follow the official implementation[4] but make the network architecture same as the other methods. In CQL, α is the coefficient to control the conservative, lower bound Q–function and there are two variants, $CQL(\rho)$ and $CQL(\mathcal{H})$. We search the hyper-parameter for both variants on our datasets. For $CQL(\rho)$, we search the Lagrange threshold τ over $\{0.1, 0.5, 2, 5, 10\}$. For $CQL(\mathcal{H})$, we search the fixed tradeoff factor α over $\{0.1, 1, 10, 100, 1000, 10000\}$. Besides, since our buffer size is 1 million, a half of the buffer size in the original D4RL experiments, we also reduce the batch size from 256 to 100 to keep it consistent with the other methods. The remaining hyper parameters are kept the same as the default value in CQL.

C.2.6 CONSTANT

To compare with the cases that there's no decay of the KL weight in our method, we set KL weight to be 10, 1, 0.1 separately and fix the decay to be 1. We run this method on the dataset with $\alpha = 0.6$ on all three games.

C.2.7 BRAC

For BRAC, we adapt the code of the official implementation[5] to use the same network architecture and datasets as the other methods. BRAC has different design choices (value penalty and/or policy regularization). We use both value penalty in the Q function objective and policy update with KL regularization, which is reported to have better performance than policy regularization only. We use the KL divergence in the primal form to keep the setting the same as our method. We search the hyper-parameter for both fixed and adaptive version to set the regularization coefficient α. For the fixed version, we search α over the range $\{0.1, 1, 10\}$. For the adaptive version, we search the threshold ϵ, which is the same as the threshold used in BEAR, over the range $\{0.05, 0.5, 2\}$. We find the fixed version has a better performance, which is consistent with the conclusion from BRAC paper. Besides, we also change the learning rate of the policy network to 0.001, which is the same as ours. Other hyper-parameters are kept the same as the default value.

[3] https://github.com/aviralkumar2907/BEAR

[4] https://github.com/aviralkumar2907/CQL

[5] https://github.com/google-research/google-research/tree/master/behavior_regularized_offline_rl

The fixed version of BRAC is equivalent to our "Constant" baseline, though there are many differences in implementation details. For example, following the official implementation of BRAC, there is a regularization for the weight of network parameters in BRAC baseline but "Constant KL" baseline does not have this term. BRAC has two training phases, the training of the behavior policy for 30K updates, and the training of BRAC with the fixed behavior policy trained before. However, "Constant" trains the behavior policy simultaneously throughout the whole procedure. In the computation of KL divergence, BRAC samples 4 actions for one state to approximate the KL divergence but "Constant" uses the closed–form solution to calculate KL divergence between two normal distributions.

C.2.8 OURS

Initially in the first J iterations we learn behavior policy β_ψ from the data and train the target policy π_θ only minimizing KL divergence between π_θ and β_ψ. To find the well-trained policy with good performance before the value estimate becomes quite noisy, we record the variance in Q estimate $var(Q^{\pi_{\theta_i}})$ for each update i. In each run, we calculate the average of the variance in 1000 iterations at the end of behavior cloning, i.e. $x = \frac{1}{1000}\sum_{i=J-1000}^{J-1} var^{\pi_{\theta_i}}(Q)$. This is a reference point of the Q variance for this run. As training continues, at the iteration j, we monitor the average of Q variance in the most recent 1000 iterations $y = \frac{1}{1000}\sum_{i=j-1000}^{j-1} var(Q^{\pi_{\theta_i}})$. If the average of Q variance is larger than 1.5 times the reference point, i.e. $\frac{y}{x} > 1.5$, the current value of τ may be not reliable and we report the score of policy checkpointed with the previous value of τ. Until the end of training, if we always have $\frac{y}{x} \leq 1.5$, then we find the policy iteration is stable and we report the final performance of the trained agent.

For our continuation method, accross the different datasets, we set most hyper-parameters the same. For the first $J = 500K$ updates, we train the policy network with only KL divergence loss to conduct behavior cloning, which is equivalent to the case that $\tau \to \infty$. We set the decay rate $\lambda = 0.9$ and decay the weight KL term with $\tau \leftarrow \lambda * \tau$ for every $I = 10000$ updates. We conduct hyper-parameter search over the initial value of τ in the set $\{50000, 10000, 1000, 100\}$.

C.3 ATARI

To generate the dataset $\mathcal{D}$ for the Atari games, we train the DQN agents using the standard online procedure to 10 million timesteps on the environment with sticky action. To generate the trajectory, with probability of 0.8, we uses ϵ-greedy with noise $\epsilon = 0.2$ for the whole episode to take actions. With probability of 0.2, we use the noise $\epsilon = 0.001$ for the whole trajectory. As explained in [13], we can ensure the dataset includes trajectories reaching the good performance of the DQN agent as well as trajectories with exploratory behavior.

C.3.1 DISCRETE BCQ

We refer to the official implementation of discrete BCQ[13], training the the behavioral policies, generating datasets and training the BCQ model. The $84 \times 84 \times 3$ RGB image is fed into a convolutional neural network. The input image first goes through an 8×8 convolution with 32 filters and stride 4, then a 4×4 convolution with 64 filters and 2 stride, followed by a 3×3 convolution with 64 filters and 1 stride, with ReLU activations. There are 2 heads, each being a fully connected layer, for Q–network output and generative model output after the convolutional neural network. Both fully connected layers have hidden size 512 and use the ReLU activation. The convolutional neural network is shared between the Q–network and the generative model. A final softmax layer is used after the output of the generative model to recover the probability for each action. We search of BCQ threshold over $\{0.1, 0.3, 0.5\}$ to find the best parameter for each game. We use the same hyper–parameter as the original implementation for training and only change the evaluation ϵ in the testing time to keep consistent with other baselines and our method.

C.3.2 REM

For REM training, we completely follow the implementation from the official codebase[6] released by [2]. The architecture is exactly the same as [2] while the dataset is from a single behavior policy. shared with discrete BCQ and our method. We use the multi-head architecture with 200 heads and do not change any hyper parameter.

C.3.3 CQL

The official implementation of CQL[7] is based on REM, so we use the same way to adapt the code and run experiments on our dataset. We follow CQL experiments on Atari, using $CQL(H)$ with a fixed tradeoff factor α. We search α over $\{0.5, 1, 4\}$ for the best hyper-parameter for each game. Other hyper-parameters are kept the same as the default setting in the official code.

C.3.4 OURS

In our method, we also use an ensemble critic network as we do in Mujoco. The architecture of each Q-network in the critic network is the same as the Q-network architecture used by discrete BCQ, with a 3–layer convolution and a fully connect layer. The policy network also uses the same architecture.

For the first $J = 500K$ updates, we train the policy network with only KL divergence loss to conduct behavior cloning, which is equivalent to the case that $\tau \to \infty$. We set initial weight of the KL term as 1, the decay rate $\lambda = 0.9$ and decay the weight KL term with $\tau \leftarrow \lambda * \tau$ for every $I = 100000$ updates. We pick the learned policy with a reasonable value of τ using the same way as in Mujoco experiment.

C.4 RECOMMENDER

Here we explain more details about the experiments for a softmax recommender. In the MovieLen-10M, the rating is of 5-score in the set of $1, 2, 3, 4, 5$. Similarly to [25], we view the scores 4 or 5 as positive feedback from users but the ratings less than 4 as negative feedback. In the bandit setting, the expected return of the policy π can be expressed as $V^{\pi}(\rho) = \mathbb{E}_{s\sim\rho, a\sim\beta(\cdot|s)}(\frac{\pi(a|s)}{\beta(a|s)} r(s,a))$ with importance sampling. So we do not use the critic network to estimate the soft Q value. We convert the objective $\tilde{V}^{\pi,\tau}(\rho)$ to $\mathbb{E}_{s\sim\rho, a\sim\beta(\cdot|s)}(\frac{\pi(a|s)}{\beta(a|s)} r(s,a)) - \tau\mathbb{E}_{s\sim\rho} KL(\pi(\cdot|s)|\beta(\cdot|s))$ in the bandit setting. Here the state distribution should be randomly sampling the users from the training dataset, and our objective is the KL regularized loss $J_{Ours}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\left[\frac{\pi_\theta(a_i|s_i)}{\beta(a_i|s_i)} r_i - \tau KL(\pi_\theta(\cdot|s_i)|\beta(\cdot|s_i))\right]$.

The architecture of the simulator, the behavior policies and target policy are the same as introduced in [25]. The simulator is trained with all 1 million records of (user, movie, binary feedback) in the MovieLen-1M dataset. We train the behavior policies with 500, 5000, 20000 samples respectively to get the behavior policies with varying performance. Then for a total of 6040 users in MovieLens-1M dataset, the movie can be selected based on the pre-trained behavior policy, and the simulator provides the binary feedback for each (user, movie) pair. If the behavior policy recommends 5 movies out of the 3900 choices for each user, we have a dataset $\mathcal{D}$ with around 30,000 samples. If the behavior policy recommend 10 movies for each user, we have a dataset $\mathcal{D}$ with around 60,000 samples. In this way, we collect a total of 6 datasets with 3 different policies and 2 different numbers of movie recommended to each user.

With the generated datasets of different sizes and different qualities, we train the target policy π_θ using three methods: 'Cross-Entropy', 'IPS' and 'Ours'. The objective functions are optimized for 400 epochs, with Adagrad optimizer and the learning rate 0.05 and the batch size 256. For our continuation method, the value of τ is initialized as 10 and decay with the rate $\lambda = 0.9$ in every epoch. The users in MovieLens-1M dataset are split into validation set (2000 users) and test set (4040 users). In each epoch, we measure the precision at 10 for the learned policy on the validation set. We take the policy achieving best performance on the validataion set during training, and test it for the held out users to report the performances in Table 3.

[6] https://github.com/google-research/batch_rl

[7] https://github.com/aviralkumar2907/CQL

D VARIANCE OF ENSEMBLE CRITIC NETWORKS

We found the error in Q estimation is highly correlated with the variance of ensemble critic network. If the learned policy π tends to select the state-action pairs with high variance in ensemble networks, the error in the Q estimation on these state-action pairs will propagate to the other state-action pairs and mislead the policy update. So we measure $\mathbb{E}_{a\sim\pi}\left[var(Q_{\phi^{(1)}}(s,a), Q_{\phi^{(2)}}(s,a), \cdots, Q_{\phi^{(K)}}(s,a))\right]$ as the criteria to determine proper value of τ. When this statistics is quite large, the policy tends to select actions with highly erroneous value estimation and we no longer trust the policy with smaller value of τ.

D.1 TRAIN THE CRITIC NETWORKS WITH THE SAME DATA

When we train the critic networks with the same data, the discrepancy in value estimation comes from the initialization of different critic networks. For state-action pairs frequently visited by the behavior policy, these data are frequently sampled to update the Q network. Therefore, the error in Q estimation and variance in Q ensemble on these state-action pairs diminishes as training goes on. But for state-action pairs less frequently visited by the behavior data, they are rarely used to update the Q network. The variance in Q ensemble are larger compared to the other state-action pairs. In Figure 6, for each state-action pair in the grid world environment, we visualize the error $|\frac{1}{K}\sum_{k=1}^{K} Q_{\Phi^{(k)}}(s,a) - \tilde{Q}^{\pi,\tau}(s,a)|$ and the variance $var(Q_{\phi^{(1)}}(s,a), Q_{\phi^{(2)}}(s,a), \cdots, Q_{\phi^{(K)}}(s,a))$ in the grid world for each state-action pair. It shows that the error becomes highly correlated with the variance as training goes on. Obviously, the frequently visited state-action pairs in the dataset has lower error and variance during training.

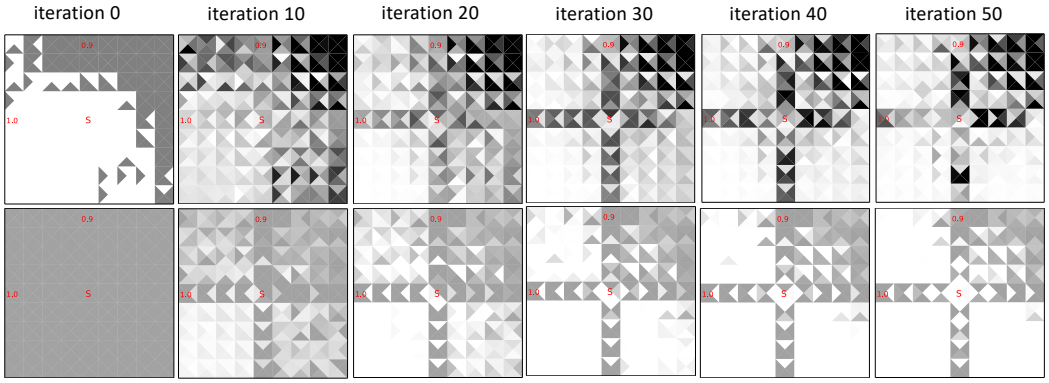

Figure 6: Visualization of the error in soft Q value estimation (the first row) and variance in the ensemble of critic networks (the second row). Triangles represent the error/variance for actions that move in different directions. Darker color indicates higher error/variance.

D.2 TRAIN THE CRITIC NETWORKS WITH THE DIFFERENT DATA

We have an alternative way to train the critic network with delete-d jackknife resampling. It is a more standard approach to estimate the variance through data perturbation. To train each of 4 critic networks, we leave out $\frac{1}{4}$ data samples in the dataset.

We assume $\hat{Q}(s,a)$ is an estimator of $\tilde{Q}^{\pi,\tau}(s,a)$. We consider the expected squared estimation error.

$$
\begin{aligned}
\mathbb{E}\left[(\hat{Q}(s,a) - \tilde{Q}^{\pi,\tau}(s,a))^2\right] &= \mathbb{E}\left[\hat{Q}(s,a)^2 + \tilde{Q}^{\pi,\tau}(s,a)^2 - 2\hat{Q}(s,a)\tilde{Q}^{\pi,\tau}(s,a)\right] \\
&= \mathbb{E}\left[\hat{Q}(s,a)^2\right] + \tilde{Q}^{\pi,\tau}(s,a)^2 - 2\mathbb{E}\left[\hat{Q}(s,a)\right]\tilde{Q}^{\pi,\tau}(s,a) \\
&= \mathbb{E}\left[\hat{Q}(s,a)\right]^2 + \tilde{Q}^{\pi,\tau}(s,a)^2 - 2\mathbb{E}\left[\hat{Q}(s,a)\right]\tilde{Q}^{\pi,\tau}(s,a) + \mathbb{E}\left[\hat{Q}(s,a)^2\right] - \mathbb{E}\left[\hat{Q}(s,a)\right]^2 \\
&= \left[\mathbb{E}\left[\hat{Q}(s,a)\right] - \tilde{Q}^{\pi,\tau}(s,a)\right]^2 + \left[\mathbb{E}\left[\hat{Q}(s,a)^2\right] - \mathbb{E}\left[\hat{Q}(s,a)\right]^2\right]
\end{aligned}
$$

The first term is the bias of the estimator and the second term is the variance of the estimator. In our case, we use the average of the ensemble Q networks $\hat{Q}(s,a) = \frac{1}{K}\sum_{k=1}^{K} Q_{\phi^{(k)}}(s,a)$ to estimate the soft Q value. With jackknife re-sampling, the variance of the estimator $\hat{Q}(s,a) = \frac{1}{K}\sum_{k=1}^{K} Q_{\phi^{(k)}}(s,a)$ can be approximated by $var(Q_{\phi^{(1)}}(s,a), Q_{\phi^{(2)}}(s,a), \cdots, Q_{\phi^{(K)}}(s,a)) = \frac{1}{K(K-1)}\sum_{k=1}^{K}\left[Q_{\phi^{(k)}}(s,a) - \frac{1}{K}\sum_{j=1}^{K} Q_{\phi^{(j)}}(s,a)\right]^2$. Thus, the squared estimation error is closely related with the variance in the ensemble $Q_{\phi^{(1)}}(s,a), Q_{\phi^{(2)}}(s,a), \cdots, Q_{\phi^{(K)}}(s,a)$.

With this method, the empirical results on Mujoco and Atari are similar to the scores we achieved when training critic networks with the same data (Table 4& 5). Therefore, we validate the use of variance of the ensemble network to detect the Q estimation error and then stop annealing τ on Mujoco and Atari as well.

	Hopper				HalfCheetah				Walker			
α	0.2	0.4	0.6	0.8	0.2	0.4	0.6	0.8	0.2	0.4	0.6	0.8
Same	2097	1569	1648	226	2145	1824	1487	547	1441	1223	853	7
Jackknife	2072	1718	1404	230	2206	1840	1529	589	1462	1078	920	8

Table 4: Performance of our method on Mujoco datasets. "same" is the variant to train each critic network with the same batch of data, which is reported in the main text. "jackknife" is the variant to train each critic network with data from jackknife resampling. The score is averaged over 5 independent runs for each method on each dataset, except that we only have one run on HalfCheetah and Walker with $\alpha = 0.2$ or $\alpha = 0.2$. We will complete the experiments with more runs.

	Amidar	Asterix	Breakout	Enduro	MsPacman	Qbert	Seaquest	SpaceInvaders
Same	175	3477	199	923	2494	4733	9935	1070
Jackknife	171	3890	217	908	2305	5389	9636	914

Table 5: Performance of our method on Atari datasets. "same" is the variant to train each critic network with the same batch of data, which is reported in the main text. "jackknife" is the variant to train each critic network with data from jackknife resampling. Due to time limit, we only have one run for the variant with jackknife resampling. We will complete the experiment with 3 independent runs on each dataset.

E SOFT POLICY ITERATION VIA LAGRANGIAN DUALITY

The discounted state-action visitations d^π of π can be defined as:

$$d^\pi(s,a) = \sum_{t=0}^{\infty} \gamma^t \mathcal{P}(s_t = s, a_t = a | s_0 \sim \rho, \forall t, a_t \sim \pi(\cdot|s_t), s_{t+1} \sim \mathcal{P}(s_t, a_t))$$

The visitation satisfies the single-step transpose Bellman recurrence:

$$d^\pi(s,a) = \rho(s)\pi(a|s) + \gamma \mathcal{P}_*^\pi d^\pi(s,a)$$

where $\mathcal{P}_*^\pi$ is the transpose policy transition operator:

$$\mathcal{P}_*^\pi d(s,a) = \pi(a|s) \sum_{\tilde{s},\tilde{a}} \mathcal{P}(s|\tilde{s},\tilde{a}) d(\tilde{s},\tilde{a})$$

Note that if we define the policy transition operator $\mathcal{P}^\pi y(s,a) = \mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \mathbb{E}_{a' \sim \pi(\cdot|s')} y(s',a')$. We have $\sum_{s,a} y(s,a) \mathcal{P}_*^\pi x(s,a) = \sum_{s,a} x(s,a) \mathcal{P}^\pi y(s,a)$.

The objective function $\tilde{V}^{\pi,\tau}(\rho)$ can be expressed as $\tilde{V}^{\pi,\tau}(\rho) = \sum_{s,a} d^\pi(s,a) r(s,a) - \tau \sum_{s,a} d^\pi(s,a) \log \frac{\pi(a|s)}{\beta(a|s)}$.

Consider the optimization problem:

$$\max_\pi \sum_{s,a} d^\pi(s,a) r(s,a) - \tau \sum_{s,a} d^\pi(s,a) \log \frac{\pi(a|s)}{\beta(a|s)}$$
$$s.t \quad d^\pi(s,a) = \rho(s)\pi(a|s) + \gamma \mathcal{P}_*^\pi d^\pi(s,a), \forall s, a$$

Application of Lagrange duality to the above problem yields $\max_\pi \min_Q L(Q,\pi)$ where

$$L(Q,\pi) = \sum_{s,a} d^\pi(s,a) r(s,a) - \tau \sum_{s,a} d^\pi(s,a) \log \frac{\pi(a|s)}{\beta(a|s)} + \sum_{s,a} Q(s,a) \left[\rho(s)\pi(a|s) + \gamma \mathcal{P}_*^\pi d^\pi(s,a) - d^\pi(s,a)\right]$$

and $Q(s,a)$ are dual variables.

We can rewrite $L(Q,\pi)$:

$$\begin{aligned}
L(Q,\pi) &= \sum_{s,a} d^\pi(s,a) r(s,a) - \tau \sum_{s,a} d^\pi(s,a) \log \frac{\pi(a|s)}{\beta(a|s)} \\
&\quad + \sum_{s,a} Q(s,a)\rho(s)\pi(a|s) + \gamma \sum_{s,a} Q(s,a) \mathcal{P}_*^\pi d^\pi(s,a) - \sum_{s,a} Q(s,a) d^\pi(s,a) \\
&= \sum_{s,a} d^\pi(s,a) r(s,a) - \tau \sum_{s,a} \rho(s)\pi(a|s) \log \frac{\pi(a|s)}{\beta(a|s)} - \tau \sum_{s,a} \gamma \mathcal{P}_*^\pi d^\pi(s,a) \log \frac{\pi(a|s)}{\beta(a|s)} \\
&\quad + \sum_{s,a} Q(s,a)\rho(s)\pi(a|s) + \gamma \sum_{s,a} Q(s,a) \mathcal{P}_*^\pi d^\pi(s,a) - \sum_{s,a} Q(s,a) d^\pi(s,a) \\
&= \sum_{s,a} Q(s,a)\rho(s)\pi(a|s) - \tau \sum_{s,a} \rho(s)\pi(a|s) \log \frac{\pi(a|s)}{\beta(a|s)} \\
&\quad + \sum_{s,a} d^\pi(s,a) r(s,a) + \gamma \sum_{s,a} (Q(s,a) - \tau \log \frac{\pi(a|s)}{\beta(a|s)}) \mathcal{P}_*^\pi d^\pi(s,a) - \sum_{s,a} d^\pi(s,a) Q(s,a) \\
&= \sum_{s,a} Q(s,a)\rho(s)\pi(a|s) - \tau \sum_{s,a} \rho(s)\pi(a|s) \log \frac{\pi(a|s)}{\beta(a|s)} \\
&\quad + \sum_{s,a} d^\pi(s,a) r(s,a) + \gamma \sum_{s,a} d^\pi(s,a) \mathcal{P}^\pi (Q(s,a) - \tau \log \frac{\pi(a|s)}{\beta(a|s)}) - \sum_{s,a} d^\pi(s,a) Q(s,a) \\
&= \mathbb{E}_{s \sim \rho} \left[\mathbb{E}_{a \sim \pi(\cdot|s)} Q(s,a) - KL(\pi(\cdot|s) \| \beta(\cdot|s)) \right] \\
&\quad + \sum_{s,a} d^\pi(s,a) \left[r(s,a) + \gamma \mathcal{P}^\pi (Q(s,a) - \tau \log \frac{\pi(a|s)}{\beta(a|s)}) - Q(s,a) \right]
\end{aligned}$$

In our soft policy evaluation step, we repeatedly apply the soft Bellman operator. At the convergence, the Q value satisfies $\mathcal{T}^{\pi,\tau}Q(s,a) = Q(s,a)$, i.e. $r(s,a)+\gamma\mathcal{P}^{\pi}(Q(s,a)-\tau\log\frac{\pi(a|s)}{\beta(a|s)})-Q(s,a) = 0$. Thus, the second part of $L(Q,\pi)$ is 0.

In our soft policy improvement step, for each state s, we optimize $\pi(\cdot|s)$ to maximize $\mathbb{E}_{a\sim\pi(\cdot|s)}Q(s,a) - KL(\pi(\cdot|s)\|\beta(\cdot|s))$. Thus, the first part of $L(Q,\pi)$ is maximized.

Therefore, the dual objective function bears some resemblance to the our soft policy iteration.

F PERFORMANCE ON DATASETS OF DIFFERENT QUALITY

We study the performance of our method on datasets of different quality. As shown in Figure 7, our method is able to learn a policy superior to the behavior policy and the gap is obvious especially when $\alpha \in [0.4, 0.6]$. In other words, when the behavior policy is mediocre, our method has a good chance to learn from the good trajectories in the dataset and achieve a better score than the average score of the dataset.

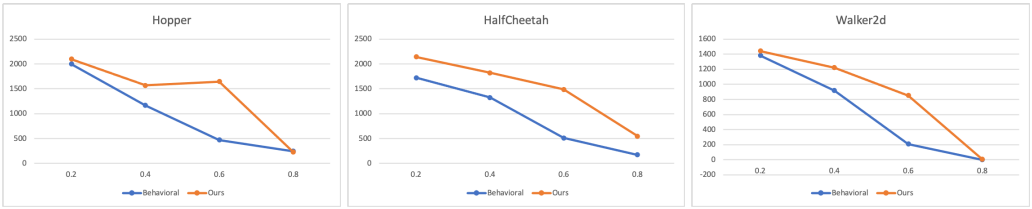

Figure 7: Performance of our learned policy and the behavior policy for different datasets. The x-axis, y-axis represents α and average reward separately. For $\alpha \in [0, 1]$, the behavior policy $\mathcal{N}(((1-\alpha)\theta_{opt} + \alpha\theta_0)^T s, 0.5\mathbb{I})$. If $\alpha = 0$, the behavior policy is the optimal. If $\alpha = 1$, the behavior policy is a random policy.

We further investigate the distribution of the trajectory rewards in the datasets. In Figure 7, we observe that our method does not perform so well in comparison with the behavior policy on the datasets (Hopper, $\alpha = 0.2$), (Hopper, $\alpha = 0.8$), (Walker, $\alpha = 0.2$), (Walker, $\alpha = 0.8$). In Figure 8, we find the problem of these four dataset is that there is only a small fraction of trajectories with cumulative rewards better than the average trajectory reward plus the standard deviation in the dataset. We study the percentage of trajectories with reward better than one standard deviation of the mean (i.e. $\%1\sigma$ trajectory). In Figure 9, we see the percentage of 1σ trajectory highly correlates with the improvement, where the improvement is the average trajectory reward of our learned policy minus the average trajectory reward of the dataset. If there is a reasonable proportion (i.e. $> 10\%$) of $\%1\sigma$ trajectory in the dataset, our method can learn from these good trajectories and perform significantly better than the average trajectory reward in the dataset.

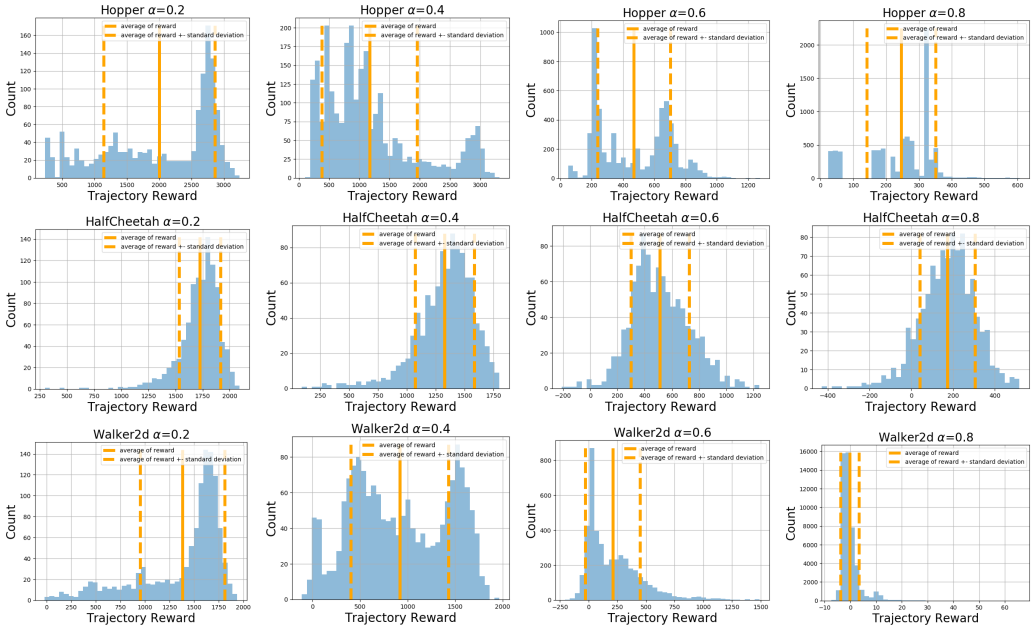

Figure 8: Histogram of the trajectory rewards in our Mujoco datasets.

Figure 9: On 12 datasets in Mujoco environment, we create the scatter plot of good trajectory percentage versus the improvement, where X-axis represents the percentage of 1σ trajectory and Y-axis represents the improvement of our method over the average trajectory reward in the dataset.