Counteractive RL: Rethinking Core Principles for Efficient and Scalable Deep Reinforcement Learning

Ezgi Korkmaz

Abstract

Following the pivotal success of learning strategies to win at tasks, solely by interacting with an environment without any supervision, agents have gained the ability to make sequential decisions in complex MDPs. Yet, reinforcement learning policies face exponentially growing state spaces in high dimensional MDPs resulting in a dichotomy between computational complexity and policy success. In our paper we focus on the agent's interaction with the environment in a high-dimensional MDP during the learning phase and we introduce a theoretically-founded novel paradigm based on experiences obtained through counteractive actions. Our analysis and method provide a theoretical basis for efficient, effective, scalable and accelerated learning, and further comes with zero additional computational complexity while leading to significant acceleration in training. We conduct extensive experiments in the Arcade Learning Environment with high-dimensional state representation MDPs. The experimental results further verify our theoretical analysis, and our method achieves significant performance increase with substantial sample-efficiency in high-dimensional environments.

1 Introduction

Utilization of deep neural networks as function approximators enabled learning functioning policies in high-dimensional state representation MDPs [Mnih et al., 2015]. Following this initial work, the current line of research trains deep reinforcement learning policies to solve highly complex problems from game solving [Hasselt et al., 2016, Schrittwieser et al., 2020] to mathematical and scientific reasoning of large language models [Guo et al., 2025]. Yet, there are still remaining unsolved problems restricting the current capabilities of reinforcement learning in exponentially growing state spaces. One of the main intrinsic open problems in deep reinforcement learning research is sample complexity and experience collection in high-dimensional state representation MDPs. While prior work extensively studied the policy's interaction with the environment in bandits and tabular reinforcement learning, and proposed various algorithms and techniques optimal to the tabular form or the bandit context [Fiechter, 1994, Kearns and Singh, 2002, Brafman and Tennenholtz, 2002, Kakade, 2003, Lu and Roy, 2019], experience collection in deep reinforcement learning remains an open challenging problem while practitioners repeatedly employ quite simple yet effective techniques (i.e. ϵ-greedy) [Whitehead and Ballard, 1991, Flennerhag et al., 2022, Hasselt et al., 2016, Wang et al., 2016, Hamrick et al., 2020, Kapturowski et al., 2023, Korkmaz, 2024, Schmied et al., 2025, Krishnamurthy et al., 2024].

Despite the provable optimality of the techniques designed for the tabular or bandit setting, they generally rely strongly on the assumptions of tabular reinforcement learning, and in particular on the ability to record tables of statistical estimates for every state-action pair which have size growing with the number of states times the number of actions. Hence, these assumptions are far from what is being faced in the deep reinforcement learning setting where states and actions can be parametrized by high-dimensional representations. Thus, in high-dimensional complex MDPs, for which deep

Correspondence to Ezgi Korkmaz: ezgikorkmazmail@gmail.com

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

neural networks are used as function approximators, the efficiency and the optimality of the methods proposed for tabular settings do not transfer well to deep reinforcement learning [Kakade, 2003]. Hence, in deep reinforcement learning research still, naive and standard techniques (e.g. ϵ-greedy) are preferred over both the optimal tabular techniques and over the particular recent experience collection techniques targeting only high scores for particular games [Mnih et al., 2015, Hasselt et al., 2016, Wang et al., 2016, Bellemare et al., 2017, Dabney et al., 2018, Flennerhag et al., 2022, Korkmaz, 2024, Kapturowski et al., 2023].

Sample efficiency still remains to be one of the main challenging problems restricting research progress in reinforcement learning. The magnitude of the number of samples required to learn and adapt continuously is one of the main limiting factors preventing current state-of-the-art deep reinforcement learning algorithms from being deployed in many diverse settings from large language model reasoning to the physical world, but most importantly one of the main challenges that needs to be dealt with on the way to building neural policies that can generalize and adapt continuously in non-stationary environments. Hence, given these limitations in our paper we aim to seek answers for the following questions:

- *How can we construct policies that have the ability to collect novel experiences in high-dimensional complex MDPs without any additional computational complexity?*
- *What is the natural theoretical motivation that can be used to design a zero-cost experience collection strategy while achieving high sample efficiency?*

To be able to answer these questions, in our paper we focus on environment interactions in deep reinforcement learning and make the following contributions:

Contributions. We introduce a fundamental theoretically well-motivated paradigm for reinforcement learning based on state-action value function minimization, which we call counteractive temporal difference learning. Our approach centers on solely reconstituting and conceptually shifting the core principles of learning and as a result increases the information gained from the environment interactions of the policy in a given MDP without adding computational complexity. We first provide the theoretical analysis in Section 3, explaining why and how minimization will result in higher temporal difference. We then as a first step demonstrate the efficacy of counteractive temporal difference learning in a motivating example, i.e. the canonical chain MDP setup, in Section 4. The results in the chain MDP verify the theoretical analysis provided in Section 3 that counteractive temporal difference learning increases temporal difference obtained from the experiences. Furthermore, we conduct an extensive study in the Arcade Learning Environment 100K benchmark with the state-of-the-art algorithms and demonstrate that our temporal difference learning algorithm CoAct TD learning improves performance by 248% across the entire benchmark compared to the baseline algorithm. We demonstrate the efficacy of our proposed CoAct TD Learning algorithm in terms of sample-efficiency. Our method based on maximizing novel experiences via minimizing the state-action value function reaches approximately to the same performance level as model-based deep reinforcement learning algorithms, without building and learning any model of the environment. Finally, we show that CoAct TD learning is a fundamental improvement over canonical methods, it is modular and a plug-and-play method, and any algorithm that uses temporal difference learning can be immediately and simply switched to CoAct TD learning.

2 Background and Preliminaries

The reinforcement learning problem is formalized as a Markov Decision Process (MDP) [Puterman, 1994] $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, r, \gamma, \rho_0, \mathcal{T} \rangle$ that contains a continuous set of states $s \in \mathcal{S}$, a set of actions $a \in \mathcal{A}$, a probability transition function $\mathcal{T}(s, a, s')$ on $\mathcal{S} \times \mathcal{A} \times \mathcal{S}$, discount factor γ, a reward function $r(s, a) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ with initial state distribution ρ_0. A policy $\pi(s, a) : \mathcal{S} \times \mathcal{A} \to [0, 1]$ in an MDP assigns a probability distribution over actions for each state $s \in \mathcal{S}$. The main goal in reinforcement learning is to learn an optimal policy π that maximizes the discounted expected cumulative rewards $\mathcal{R} = \mathbb{E}_{a_t \sim \pi(s_t, \cdot), s_{t+1} \sim \mathcal{T}(s_t, a_t, \cdot)} \sum_t \gamma^t r(s_t, a_t)$. In Q-learning [Watkins, 1989] the learned policy is parameterized by a state-action value function $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, which represents the value of taking action a in state s. The optimal state-action value function is learnt via iterative Bellman update

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r(s_t, a_t) + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$$

where $\max_a Q(s_{t+1}, a) = \mathcal{V}(s_{t+1})$. Let a^* be the action maximizing the state-action value function, $a^*(s) = \arg\max_a Q(s, a)$, in state s. Once the Q-function is learnt the policy is determined via taking action $a^*(s)$. Temporal difference learning [Sutton, 1988] improves the estimates of the state-action values in each iteration via the Bellman Operator [Bellman, 1957]

$$(\Omega^* Q)(s, a) = \mathbb{E}_{s' \sim \mathcal{T}(s,a,\cdot)}[r(s, a) + \gamma \max_{a'} Q(s', a')].$$

For distributional reinforcement learning, QRDQN is an algorithm that is based on quantile regression [Koenker and Hallock, 2001, Koenker, 2005] temporal difference learning

$$\Omega\mathcal{Z}(s, a) = r(s, a) + \gamma\mathcal{Z}(s', \arg\max_{a'} \mathbb{E}_{z \sim \mathcal{Z}(s',a')}[z]) \text{ and } \mathcal{Z}(s, a) \coloneqq \frac{1}{N} \sum_{i=1}^{N} \delta_{\theta_i(s,a)}$$

where $\mathcal{Z}_\theta \in \mathcal{Z}_Q$ maps state-action pairs to a probability distribution over values. In deep reinforcement learning, the state space or the action space is large enough that it is not possible to learn and store the state-action values in a tabular form. Thus, the Q-function is approximated via deep neural networks. In deep double-Q learning, two Q-networks are used to decouple the Q-network deciding which action to take and the Q-network to evaluate the action taken

$$\theta_{t+1} = \theta_t + \alpha(r(s_t, a_t) + \gamma Q(s_{t+1}, \arg\max_a Q(s_{t+1}, a; \theta_t); \hat{\theta}_t) - Q(s_t, a_t; \theta_t))\nabla_{\theta_t} Q(s_t, a_t; \theta_t).$$

Current deep reinforcement learning algorithms use ϵ-greedy during training [Wang et al., 2016, Mnih et al., 2015, Hasselt et al., 2016, Hamrick et al., 2020, Flennerhag et al., 2022, Kapturowski et al., 2023, Krishnamurthy et al., 2024, Schmied et al., 2025]. In particular, the ϵ-greedy [Whitehead and Ballard, 1991] algorithm takes an action $a_k \sim \mathcal{U}(\mathcal{A})$ with probability ϵ in a given state s, i.e. $\pi(s, a_k) = \frac{\epsilon}{|\mathcal{A}|}$, and takes an action $a^* = \arg\max_a Q(s, a)$ with probability $1 - \epsilon$, i.e.

$$\pi(s, \arg\max_a Q(s, a)) = 1 - \epsilon + \frac{\epsilon}{|\mathcal{A}|}$$

While a family of algorithms have been proposed based on counting state visitations (i.e. the number of times action a has been taken in state s by time step t) with provable optimal regret bounds using the principal of optimism in the face of uncertainty in the tabular MDP setting, yet incorporating these count-based methods in high-dimensional state representation MDPs requires substantial complexity including training additional deep neural networks to estimate counts or other uncertainty metrics. As a result, many state-of-the-art deep reinforcement learning algorithms still use simple, randomized experience collection methods based on sampling a uniformly random action with probability ϵ [Mnih et al., 2015, Hasselt et al., 2016, Wang et al., 2016, Hamrick et al., 2020, Flennerhag et al., 2022, Korkmaz, 2023, Kapturowski et al., 2023]. In our experiments, while providing comparison against canonical methods, we also compare our method against computationally complicated and expensive techniques such as noisy-networks that is based on the injection of random noise with additional layers in the deep neural network [Hessel et al., 2018] in Section 5, and count based methods in Section 4 and Section 6. We further highlight that our method is a fundamental theoretically motivated improvement of temporal difference learning. Thus, any algorithm that is based on temporal difference learning can immediately be switched to CoAct TD learning.

3 Maximizing Temporal Difference with Counteractive Actions

Seeking experiences that contain high information has long been the focus of reinforcement learning [Schmidhuber, 1991, 1999, Moore and Atkeson, 1993] and more particularly the experiences that correspond to higher temporal difference [Moore and Atkeson, 1993]. In this section we will provide the theoretical analysis for our proposed algorithm counteractive TD learning. Section 5 further provides the experimental results verifying the theoretical predictions. In deep reinforcement learning the state-action value function is initialized with random weights [Mnih et al., 2015, 2016, Hasselt et al., 2016, Wang et al., 2016, Schaul et al., 2016, Oh et al., 2020, Schrittwieser et al., 2020, Hubert et al., 2021]. During a large portion of the training prior to convergence, the Q-function behaves as a random function rather than providing an accurate representation of the optimal state-action values while interacting with new experiences in high-dimensional MDPs as the learning continues. In particular, in high-dimensional environments in a significant portion of the training the Q-function, on average, assigns approximately similar values to states that are similar, and has little correlation with the immediate rewards. Hence, let us formalize these facts on the state-action value function in the following definitions.

Definition 3.1 (η-*uninformed*)**.** Let $\eta > 0$. A Q-function parameterized by weights $\theta \sim \Theta$ is η-uninformed if for any state $s \in \mathcal{S}$ with $a^{\min} = \arg\min_a Q_\theta(s, a)$ we have

$$|\mathbb{E}_{\theta\sim\Theta}[r(s_t, a^{\min})] - \mathbb{E}_{a\sim\mathcal{U}(\mathcal{A})}[r(s_t, a)]| < \eta.$$

Definition 3.2 (δ-*smooth*)**.** Let $\delta > 0$. A Q-function parameterized by weights $\theta \sim \Theta$ is δ-smooth if for any state $s \in \mathcal{S}$ and action $\hat{a} = \hat{a}(s, \theta)$ with $s' \sim \mathcal{T}(s, \hat{a}, \cdot)$ we have

$$|\mathbb{E}_{\theta\sim\Theta}[\max_a Q_\theta(s, a)] - \mathbb{E}_{s'\sim\mathcal{T}(s,\hat{a},\cdot),\theta\sim\Theta}[\max_a Q_\theta(s', a)]| < \delta$$

where the expectation is over both the random initialization of the Q-function weights, and the random transition to state $s' \sim \mathcal{T}(s, \hat{a}, \cdot)$.

Definition 3.3 (*Disadvantage Gap*)**.** For a state-action value function Q_θ the disadvantage gap in a state $s \in \mathcal{S}$ is given by $\mathcal{D}(s) = \mathbb{E}_{a\sim\mathcal{U}(\mathcal{A}),\theta\sim\Theta}[Q_\theta(s, a) - Q_\theta(s, a^{\min})]$ where $a^{\min} = \arg\min_a Q_\theta(s, a)$.

The following theorem captures the intuition that choosing counteractive actions, i.e. the action minimizing the state-action value function, will achieve an above-average temporal difference.

Theorem 3.4 (*Counteractive Actions Increases Temporal Difference*)**.** *Let $\eta, \delta > 0$ and suppose that $Q_\theta(s, a)$ is η-uninformed and δ-smooth. Let $s_t \in \mathcal{S}$ be a state, and let a^{min} be the action minimizing the state-action value in a given state s_t, $a^{min} = \arg\min_a Q_\theta(s_t, a)$. Let $s^{min}_{t+1} \sim \mathcal{T}(s_t, a^{min}, \cdot)$. Then for an action $a_t \sim \mathcal{U}(\mathcal{A})$ with $s_{t+1} \sim \mathcal{T}(s_t, a_t, \cdot)$ we have*

$$\begin{aligned}
&\mathbb{E}_{s^{min}_{t+1}\sim\mathcal{T}(s_t,a^{min},\cdot),\theta\sim\Theta}[r(s_t, a^{min}) + \gamma \max_a Q_\theta(s^{min}_{t+1}, a) - Q_\theta(s_t, a^{min})] \\
&> \mathbb{E}_{a_t\sim\mathcal{U},(\mathcal{A})s_{t+1}\sim\mathcal{T}(s_t,a_t,\cdot),\theta\sim\Theta}[r(s_t, a_t) + \gamma \max_a Q_\theta(s_{t+1}, a) - Q_\theta(s_t, a_t)] + \mathcal{D}(s_t) - 2\delta - \eta
\end{aligned}$$

Proof. Since $Q_\theta(s, a)$ is δ-smooth we have

$$\begin{aligned}
&\mathbb{E}_{s^{min}_{t+1}\sim\mathcal{T}(s_t,a^{min},\cdot),\theta\sim\Theta}[\gamma \max_a Q_\theta(s^{min}_{t+1}, a) - Q_\theta(s_t, a^{min})] \\
&\qquad > \gamma\mathbb{E}_{\theta\sim\Theta}[\max_a Q_\theta(s_t, a)] - \delta - \mathbb{E}_{\theta\sim\Theta}[Q_\theta(s_t, a^{min})] \\
&\qquad > \gamma\mathbb{E}_{s_{t+1}\sim\mathcal{T}(s_t,a_t,\cdot),\theta\sim\Theta}[\max_a Q_\theta(s_{t+1}, a)] - 2\delta - \mathbb{E}_{\theta\sim\Theta}[Q_\theta(s_t, a^{min})] \\
&\qquad \geq \mathbb{E}_{a_t\sim\mathcal{U}(\mathcal{A}),s_{t+1}\sim\mathcal{T}(s_t,a_t,\cdot),\theta\sim\Theta}[\gamma \max_a Q_\theta(s_{t+1}, a) - Q_\theta(s_t, a_t)] + \mathcal{D}(s_t) - 2\delta
\end{aligned}$$

where the last line follows from Definition 3.3. Further, because $Q_\theta(s, a)$ is η-uninformed, $\mathbb{E}_{\theta\sim\Theta}[r(s_t, a^{\min})] > \mathbb{E}_{a_t\sim\mathcal{U}(\mathcal{A})}[r(s_t, a_t)] - \eta$. Combining with the previous inequality completes the proof. □

Theorem 3.4 shows that counteractive actions, i.e. actions that minimize the state-action value function, in fact increase temporal difference. Now we will prove that counteractive actions achieve an increase in temporal difference further in the case where action selection and evaluation in the temporal difference are computed with two different sets of weights θ and $\hat{\theta}$ as in double Q-learning.

Definition 3.5 (δ-*smoothness for Double-Q*)**.** Let $\delta > 0$. A pair of Q-functions parameterized by weights $\theta \sim \Theta$ and $\hat{\theta} \sim \Theta$ are δ-smooth if for any state $s \in \mathcal{S}$ and action $\hat{a} = \hat{a}(s, \theta) \in \mathcal{A}$ with $s' \sim \mathcal{T}(s, \hat{a}, \cdot)$, we have

$$\left| \mathop{\mathbb{E}}_{\substack{\theta,\hat{\theta}\sim\Theta \\ s'\sim\mathcal{T}(s,\hat{a},\cdot)}} \left[Q_{\hat{\theta}}(s, \arg\max_a Q_\theta(s, a)) \right] - \mathop{\mathbb{E}}_{\substack{\theta,\hat{\theta}\sim\Theta \\ s'\sim\mathcal{T}(s,\hat{a},\cdot)}} \left[Q_{\hat{\theta}}(s', \arg\max_a Q_\theta(s', a)) \right] \right| < \delta$$

where the expectation is over both the random initialization of the Q-function weights θ and $\hat{\theta}$, and the random transition to state $s' \sim \mathcal{T}(s, \hat{a}, \cdot)$.

Now we will prove that counteractive actions, i.e. actions that minimize the state-action value instead of maximizing, will lead to increase in temporal difference in the case of two Q-functions, i.e. Q_θ and $Q_{\hat{\theta}}$, that are alternatively used to take an action and evaluate the value of the action.

Theorem 3.6. *Let $\eta, \delta > 0$ and suppose that Q_θ and $Q_{\hat{\theta}}$ are η-uniformed and δ-smooth. Let $s_t \in \mathcal{S}$ be a state, and let $a^{min} = \arg\min_a Q_\theta(s_t, a)$. Let $s_{t+1}^{min} \sim \mathcal{T}(s_t, a^{min}, \cdot)$. Then for an action $a_t \sim \mathcal{U}(\mathcal{A})$ with $s_{t+1} \sim \mathcal{T}(s_t, a_t, \cdot)$ we have*

$$\begin{aligned}
&\mathbb{E}_{s_{t+1}\sim\mathcal{T}(s,a,\cdot),\theta\sim\Theta,\hat{\theta}\sim\Theta}[r(s_t, a^{min}) + \gamma Q_{\hat{\theta}}(s_{t+1}^{min}, \arg\max_a Q_\theta(s_{t+1}^{min}, a)) - Q_\theta(s_t, a^{min})] \\
&\quad > \mathbb{E}_{a_t\sim\mathcal{U}(\mathcal{A}),s_{t+1}\sim\mathcal{T}(s,a,\cdot),\theta\sim\Theta,\hat{\theta}\sim\Theta}[r(s_t, a_t) + \gamma Q_{\hat{\theta}}(s_{t+1}, \arg\max_a Q_\theta(s_{t+1}, a)) - Q_\theta(s_t, a_t)] \\
&\qquad + \mathcal{D}(s_t) - 2\delta - \eta
\end{aligned}$$

Proof. Since Q_θ and $Q_{\hat{\theta}}$ are δ-smooth we have

$$\begin{aligned}
&\mathbb{E}_{s_{t+1}^{min}\sim\mathcal{T}(s_t,a^{min},\cdot),\theta\sim\Theta,\hat{\theta}\sim\Theta}[+\gamma Q_{\hat{\theta}}(s_{t+1}^{min}, \arg\max_a Q_\theta(s_{t+1}^{min}, a)) - Q_\theta(s_t, a^{min})] \\
&\quad > \mathbb{E}_{s_{t+1}^{min}\sim\mathcal{T}(s_t,a^{min},\cdot),\theta\sim\Theta,\hat{\theta}\sim\Theta}[+\gamma Q_{\hat{\theta}}(s_t, \arg\max_a Q_\theta(s_t, a)) - Q_\theta(s_t, a^{min})] - \delta \\
&\quad > \mathbb{E}_{s_{t+1}\sim\mathcal{T}(s_t,a_t,\cdot),\theta\sim\Theta,\hat{\theta}\sim\Theta}[+\gamma Q_{\hat{\theta}}(s_{t+1}, \arg\max_a Q_\theta(s_{t+1}, a)) - Q_\theta(s_t, a^{min})] - 2\delta \\
&\quad \geq \mathbb{E}_{s_{t+1}\sim\mathcal{T}(s_t,a_t,\cdot),\theta\sim\Theta,\hat{\theta}\sim\Theta}[+\gamma Q_{\hat{\theta}}(s_{t+1}, \arg\max_a Q_\theta(s_{t+1}, a)) - Q_\theta(s_t, a_t)] + \mathcal{D}(s_t) - 2\delta
\end{aligned}$$

where the last line follows from Definition 3.3. Further, because Q_θ and $Q_{\hat{\theta}}$ are η-uniformed, $\mathbb{E}_{\theta\sim\Theta,\hat{\theta}\sim\Theta}[r(s_t, a^{\min})] > \mathbb{E}_{a_t\sim\mathcal{U}(\mathcal{A})}[r(s_t, a_t)] - \eta$. Combining with the previous inequality completes the proof. □

Core Counterintuition:

How could minimizing the state-action value function accelerate learning?

At first, the results in Theorem 3.4 and 3.6 might appear counterintuitive. Yet, understanding this counterintuitive fact relies on first understanding the intrinsic difference between the randomly initialized state-action value function, i.e. Q_θ, and the optimal state-action value function, i.e. Q^*. In particular, from the perspective of the function Q^*, the action $a_{Q_\theta}^{\min}(s) = \arg\min_a Q_\theta(s, a)$ is a uniform random action. However, from the perspective of the function Q_θ, the action $a^{\min}$ is meaningful, in that it will lead to a higher TD-error update than any other action; hence the realization of the intrinsic difference between $a_{Q_\theta}^{\min}(s)$ and $a_{Q^*}^{\min}(s)$ with regard to Q_θ and Q^* provides a valuable insight on how counteractive actions do in fact increase temporal difference. In fact, Theorem 3.4 and 3.6 precisely provide the formalization that the temporal difference achieved by taking the minimum action is larger than that of a random action by an amount equal to the disadvantage gap $\mathcal{D}(s)$. Experimental results reported in Section 5 further verify the theoretical analysis. Now we will formalize this intuition for initialization and prove that the distribution of the minimum value action in a given state is uniform by itself, but is constant once it is conditioned on the weights θ.

Proposition 3.7 (*Marginal and Conditional Distribution of Counteractive Actions*)**.** *Let θ be the random initial weights for the Q-function. For any state $s \in \mathcal{S}$ let $a^{min}(s) = \arg\min_{a'\in\mathcal{A}} Q_\theta(s, a')$. Then for any $a \in \mathcal{A}$, $\mathbb{P}_{\theta\sim\Theta}\left[\arg\min_{a'\in\mathcal{A}} Q_\theta(s, a') = a\right] = \frac{1}{|\mathcal{A}|}$ i.e. the distribution $\mathbb{P}_{\theta\sim\Theta}[a^{min}(s)]$ is uniform. Simultaneously, the conditional distribution $\mathbb{P}_{\theta\sim\Theta}[a^{min}(s) \mid \theta]$ is constant.*

The proof is provided in the supplementary material. Proposition 3.7 shows that in states in which the Q-function has not received sufficient updates, taking the minimum action is almost equivalent to taking a random action with respect to its contribution to the rewards obtained. However, while the action chosen early on in training is almost uniformly random when only considering the current state, it is at the same time still completely determined by the current value of the weights θ, as is the temporal difference. Thus while the marginal distribution on actions taken is uniform, the temporal difference when taking counteractive actions, i.e. the minimum action, is quite different than from the case where an independently random action is chosen. In particular, in expectation over the random initialization $\theta \sim \Theta$, the temporal difference is higher when taking the minimum value action than that of a random action as demonstrated in Section 3.

The main objective of our approach is to increase the information gained from each environment interaction, and we show that this can be achieved via actions that minimize the state-action value

Algorithm 1 CoAct TD Learning: Counteractive Temporal Difference Learning

Input: In MDP $\mathcal{M}$ with $\gamma \in (0, 1]$, $s \in \mathcal{S}$, $a \in \mathcal{A}$ with $Q_\theta(s, a)$ function parametrized by θ, ϵ dithering parameter, $\mathcal{B}$ experience replay buffer, $\mathcal{N}$ is the training learning steps.

Populating Experience Replay Buffer:
for s_t in e **do**
  Sample $\kappa \sim U(0, 1)$
  if $\kappa < \epsilon$ **then**
    $a^{min} = \arg\min_a Q(s_t, a)$
    $s^{min}_{t+1} \sim \mathcal{T}(s_t, a^{min}, \cdot)$
    $\mathcal{B} \leftarrow (r(s_t, a^{min}), s_t, s^{min}_{t+1}, a^{min})$
  else
    $a^{\max} = \arg\max_a Q(s_t, a)$
    $s_{t+1} \sim \mathcal{T}(s_t, a^{\max}, \cdot)$
    $\mathcal{B} \leftarrow (r(s_t, a^{\max}), s_t, s_{t+1}, a^{\max})$
  end if
end for

Learning:
for n in $\mathcal{N}$ **do**
  Sample from replay buffer:
  $\langle s_t, a_t, r(s_t, a_t), s_{t+1} \rangle \sim \mathcal{B}$
  Thus, $\mathcal{TD}$ receives update with probability ϵ
  $\mathcal{TD} = r(s_t, a^{min}) + \gamma \max_a Q(s^{min}_{t+1}, a) - Q(s_t, a^{min})$
  $\mathcal{TD}$ receives update with probability $1 - \epsilon$:
  $\mathcal{TD} = r(s_t, a^{\max}) + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a^{\max})$
  Gradient step with $\nabla \mathcal{L}(\mathcal{TD})$
end for

function. While minimization of the Q-function may initially be regarded as counterintuitive, Section 3 provides the exact theoretical justification on how taking actions that minimize the state-action value function results in higher temporal difference for the corresponding state transitions, and Section 5 provides experimental results that verify the theoretical analysis. Our method is a fundamental theoretically well-motivated improvement on temporal difference learning. Thus, any algorithm in reinforcement learning that is built upon temporal difference learning can be simply switched to CoAct TD learning. Algorithm 1 summarizes our proposed algorithm CoAct TD Learning based on minimizing the state-action value function as described in detail in Section 3. Note that populating the experience replay buffer and learning are happening simultaneously with different rates. TD receives an update with probability ϵ solely due to the experience collection.

CoAct TD is modular: It is a plug-and-play method with any canonical baseline algorithm.

4 Motivating Example

To truly understand the intuition behind our counterintuitive foundational method, we consider a motivating example: the chain MDP. In particular, the chain MDP which consists of a chain of n states $s \in \mathcal{S} = \{1, 2, \cdots n\}$ each with four actions. Each state i has one action that transitions the agent up the chain by one step to state $i + 1$, one action that transitions the agent to state 2, one action that transitions the agent to state 3, and one action which resets the agent to state 1 at the beginning of the chain. All transitions have reward zero, except for the last transition returning the agent to the beginning from the n-th state. Thus, when started from the first state in the chain, the agent must learn a policy that takes $n - 2$ consecutive steps up the chain, and then one final step to reset and get the reward. For the chain MDP, we compare standard approaches in temporal difference learning in tabular Q-learning with our method CoAct TD Learning based on minimization of the state-action values. In particular we compare our method CoAct TD Learning with both the ϵ-greedy action selection method, and the upper confidence bound (UCB) method. In more detail, in the UCB method the number of training steps t, and the number of times $N_t(s, a)$ that each action a has been taken in state s by step t are recorded. Furthermore, the action $a \in \mathcal{A}$ selection is determined as follows:

$$a^{\text{UCB}} = \arg\max_{a \in \mathcal{A}} Q(s, a) + 2\sqrt{\frac{\log t}{N_t(s, a)}}.$$

In a given state s if $N(s, a) = 0$ for any action a, then an action is sampled uniformly at random from the set of actions a' with $N(s, a') = 0$. For the experiments reported in our paper the length of the chain is set to $n = 10$. The Q-function is initialized by independently sampling each state-action value from a normal distribution with $\mu = 0$ and $\sigma = 0.1$. In each iteration we train the agent using Q-learning for 100 steps, and then evaluate the reward obtained by the argmax policy using the current Q-function for 100 steps. Note that the maximum achievable reward in 100 steps is 11.

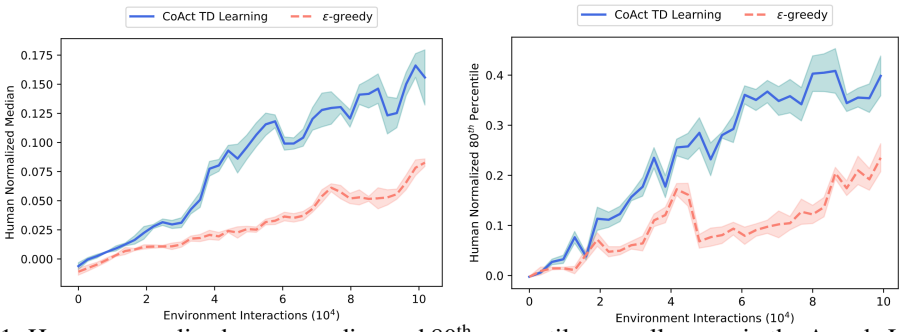

Figure 1: Human normalized scores median and 80$^{\text{th}}$ percentile over all games in the Arcade Learning Environment (ALE) 100K benchmark for CoAct TD Learning and the canonical temporal difference learning with ϵ-greedy for QRDQN. Left: Median. Right: 80$^{\text{th}}$ Percentile.

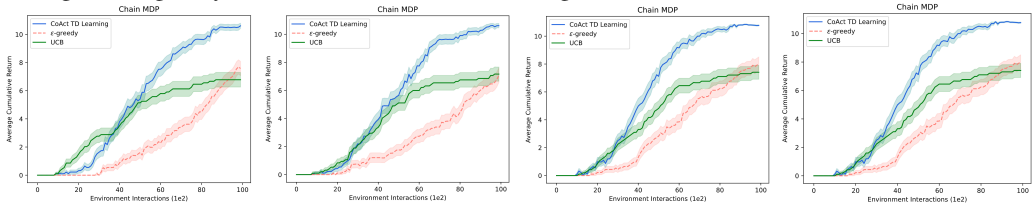

Figure 2: Learning curves in the chain MDP with our proposed algorithm CoAct TD Learning, the canonical algorithm ϵ-greedy and the UCB algorithm with variations in ϵ.

Figure 2 reports the learning curves for each method with varying $\epsilon \in [0.15, 0.25]$ with step size 0.025. The results in Figure 2 demonstrate that our method converges faster to the optimal policy than either of the standard approaches.

5 Experimental Results

The experiments are conducted in the Arcade Learning Environment (ALE). We conduct empirical analysis with multiple baseline algorithms including Deep Double-Q Network [Hasselt et al., 2016] initially proposed in [van Hasselt, 2010] trained with prioritized experience replay [Schaul et al., 2016] without the dueling architecture with its original version [Hasselt et al., 2016], and the QRDQN algorithm that is also described in Section 2. The experiments are conducted both in the 100K Arcade Learning Environment benchmark, and the canonical version with 200 million frame training [Mnih et al., 2015, Wang et al., 2016]. Note that the 100K Arcade Learning Environment benchmark is an established baseline proposed to measure sample efficiency in deep reinforcement learning research, and contains 26 different Arcade Learning Environment games. The policies are evaluated after 100000 environment interactions. All of the polices in the experiments are trained over 5 random seeds. The hyperparameters, the architecture details, and additional experimental results are reported in the supplementary material. All of the results in the paper are reported with the standard error of the mean. The human normalized scores are computed as, $\text{HN} = (\text{Score}_{agent} - \text{Score}_{random})/(\text{Score}_{human} - \text{Score}_{random})$.

Figure 1 reports results of human normalized median scores and 80$^{\text{th}}$ percentile over all of the games of the Arcade Learning Environment (ALE) in the low-data regime for QRDQN, while Figure 5 reports results for double Q-learning. The results reported in Figure 1 once more demonstrate that the performance obtained by the CoAct TD Learning algorithm is approximately double the performance achieved by the canonical experience collection techniques. For completeness we also report several results with 200 million frame training (i.e. 50 million environment interactions). Furthermore, we also compare our proposed CoAct TD Learning algorithm with NoisyNetworks as referred to in Section 2. Table 1 reports results of human normalized median scores, 20$^{\text{th}}$ percentile, and 80$^{\text{th}}$ percentile for the Arcade Learning Environment 100K benchmark. Table 1 further demonstrates that the CoAct TD Learning algorithm achieves significantly better performance results compared to NoisyNetworks. Primarily, note that NoisyNetworks includes adding layers in the Q-network to increase exploration. However, this increases the number of parameters that have been added in the training process; thus, introducing substantial additional cost. Figure 4 demonstrates the learning curves for our proposed algorithm CoAct TD Learning and the original version of the DDQN algorithm with ϵ-greedy training. In the large data regime we observe that while in some MDPs our proposed method CoAct TD Learn-

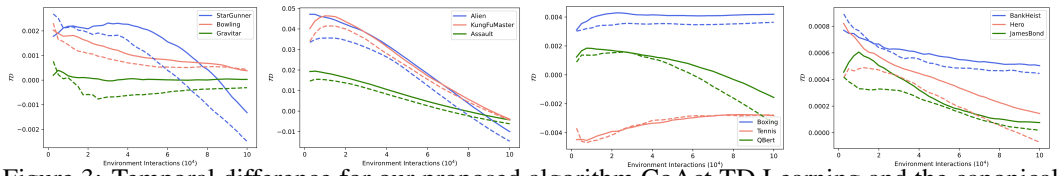

Figure 3: Temporal difference for our proposed algorithm CoAct TD Learning and the canonical ϵ-greedy algorithm in the Arcade Learning Environment 100K benchmark. Dashed lines report the temporal difference for the ϵ-greedy algorithm and solid lines report the temporal difference for the CoAct TD Learning algorithm. Colors indicate games.

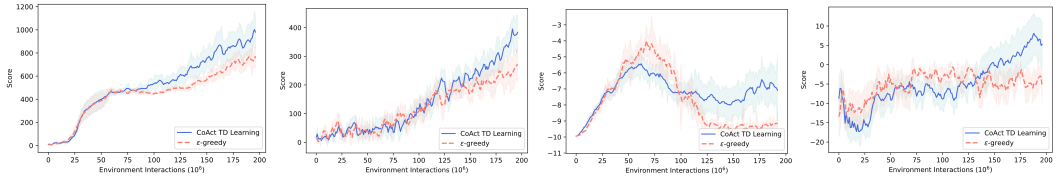

Figure 4: The learning curves for our proposed method CoAct TD Learning algorithm and canonical temporal difference learning in the Arcade Learning Environment with 200 million frame training. Left: JamesBond. MiddleLeft: Gravitar. MiddleRight: Surround. Right: Tennis.

ing that focuses on experience collection with novel temporal difference boosting via counteractive actions converges faster, in other MDPs CoAct TD Learning simply converges to a better policy. Table 1 demonstrates that our proposed CoAct TD Learning algorithm improves on the performance of the canonical algorithm ϵ-greedy by 248% and NoisyNetworks by 204%. The results reported in both of the sample regimes demonstrate that CoAct TD learning results in faster convergence rate and significantly improves sample-efficiency in deep reinforcement learning. The large scale experimental analysis verifies the theoretical predictions of Section 3, and further discovers that the CoAct TD Learning algorithm achieves substantial sample-efficiency with zero-additional cost across many algorithms and different sample-complexity regimes over canonical baseline alternatives.

Table 1: Human normalized scores median, 20$^{\text{th}}$ and 80$^{\text{th}}$ percentile across all of the games in the Arcade Learning Environment 100K benchmark for CoAct TD Learning, ϵ-greedy and NoisyNetworks with DDQN.

Method	CoAct TD	ϵ-greedy	NoisyNetworks
Median	**0.0927±0.0050**	0.0377±0.0031	0.0457±0.0035
20$^{\text{th}}$ Percentile	**0.0145±0.0003**	0.0056±0.0017	0.0102±0.0018
80$^{\text{th}}$ Percentile	**0.3762±0.0137**	0.2942±0.0233	0.1913±0.0144

6 Investigating the Temporal Difference

In Section 3 we provided the theoretical analysis and justification for collecting experiences with counteractive actions, i.e. the minimum Q-value action, in which counteractive actions increase the temporal difference. Increasing temporal difference of the experiences results in novel transitions, and hence accelerates learning [Andre et al., 1997]. The theoretical analysis from Theorem 3.4 and Theorem 3.6 shows that taking the minimum value action results in an increase in the temporal difference. In this section, we further investigate the temporal difference and provide empirical measurements of the TD. To measure the change in the temporal difference when taking the minimum action versus the average action, we compare the temporal difference obtained by CoAct TD Learning with that obtained by several other canonical methods. In more detail, during training, for each batch Λ of transitions of the form (s_t, a_t, s_{t+1}) we record, the temporal difference $\mathcal{TD}$

$$\mathbb{E}_{(s_t,a_t,s_{t+1})\sim\Lambda}\mathcal{TD}(s_t,a_t,s_{t+1}) = \mathbb{E}_{(s_t,a_t,s_{t+1})\sim\Lambda}[r(s_t,a_t) + \gamma \max_a Q_\theta(s_{t+1},a) - Q_\theta(s_t,a_t)].$$

The results reported in Figure 3 and Figure 6 further confirm the theoretical predictions made via Definition 3.2, Theorem 3.6 and Theorem 3.4. In addition to the results for individual games reported in Figure 3, we compute a normalized measure of the gain in temporal difference achieved when using CoAct TD Learning and plot the median across games. We define the normalized $\mathcal{TD}$ gain to be, Normalized $\mathcal{TD}$ Gain $= 1 + (\mathcal{TD}_{\text{method}} - \mathcal{TD}_{\epsilon\text{-greedy}})/(|\mathcal{TD}_{\epsilon\text{-greedy}}|)$, where $\mathcal{TD}_{\text{method}}$ and $\mathcal{TD}_{\epsilon\text{-greedy}}$ are the temporal difference for any given learning method and ϵ-greedy respectively. The leftmost and middle plot of Figure 6 report the median across all games of the normalized $\mathcal{TD}$ gain results for CoAct TD Learning and NoisyNetworks in the Arcade Learning Environment 100K benchmark. Note that, consistent with the predictions of Theorem 3.4, the median normalized

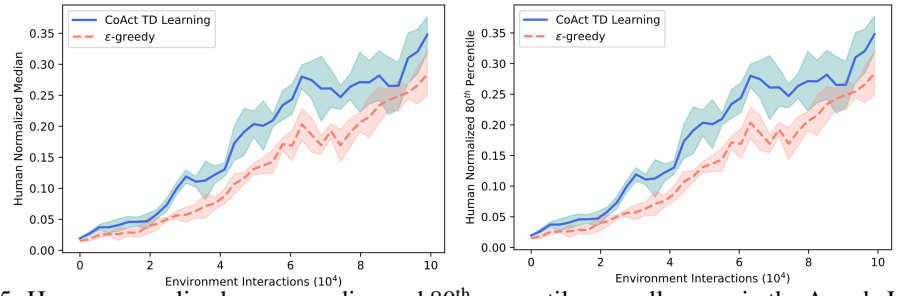

Figure 5: Human normalized scores median and 80$^{\text{th}}$ percentile over all games in the Arcade Learning Environment (ALE) 100K benchmark in DDQN for CoAct TD Learning algorithm and the canonical temporal difference learning with ϵ-greedy. Left:Median. Right: 80$^{\text{th}}$ Percentile.

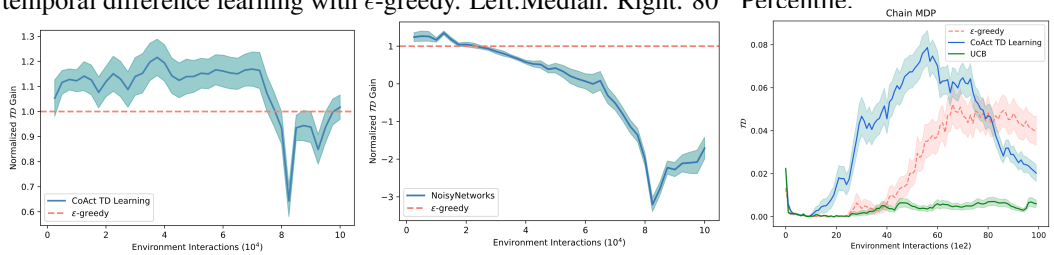

Figure 6: Left and Middle: Normalized temporal difference $\mathcal{TD}$ gain median across all games in the Arcade Learning Environment 100K benchmark for CoAct TD Learning and NoisyNetworks. Right: Temporal difference $\mathcal{TD}$ when exploring chain MDP with Upper Confidence Bound (UCB) method, ϵ-greedy and our proposed algorithm CoAct TD Learning.

temporal difference gain for CoAct TD Learning is up to 25 percent larger than that of ϵ-greedy. The results for NoisyNetworks demonstrate that alternate experience collection methods lack this positive bias relative to the uniform random action. Further note that to guarantee that every action has non-zero probability of being chosen in every possible state for guaranteeing convergence of Q-learning, one can additionally introduce noise, and achieve higher temporal difference by CoAct TD learning. Albeit, across all of the experiments introduction of the noise was not necessary, as there is sufficient noise in the training process that satisfies the property for convergence. Hence, across all the benchmarks CoAct-TD Learning results in consistently and substantially better performance. **CoAct TD Learning** is extremely modular, **only requires two lines of additional code** and can be used as **a drop-in replacement for any baseline algorithm** that uses the canonical methods. The fact that, as demonstrated in Table 1, CoAct TD Learning significantly outperforms noisy networks in the low-data regime is further evidence of the advantage the positive bias in temporal difference confers. The rightmost plot of Figure 6 reports $\mathcal{TD}$ for the motivating example of the chain MDP. As in the large-scale experiments, prior to convergence CoAct TD Learning exhibits a notably larger temporal difference relative to the canonical baseline methods, thus CoAct TD Learning achieves accelerated learning across domains from tabular MDPs to large-scale.

7 Conclusion

In our study we focus on the following questions: *(i) Is it possible to maximize sample efficiency in deep reinforcement learning without additional computational complexity by solely rethinking the core principles of learning?, (ii) What is the foundation and theoretical motivation of the learning paradigm we introduce that results in one of the most computationally efficient ways to explore in deep reinforcement learning?* and, *(iii) How would the theoretically well-motivated approach transfer to high-dimensional complex MDPs?* To be able to answer these questions we propose a novel, theoretically motivated method with zero additional computational cost based on counteractive actions that minimize the state-action value function in deep reinforcement learning. We demonstrate theoretically that our method CoAct TD Learning based on minimization of the state-action value results in higher temporal difference, and thus creates novel transitions with more unique experience collection. Following the theoretical motivation we initially demonstrate in a motivating example in the chain MDP setup that our proposed method CoAct TD Learning results in achieving higher sample-efficiency. Then, we expand this intuition and conduct large scale experiments in the Arcade Learning Environment, and demonstrate that our proposed method CoAct TD Learning increases the performance on the Arcade Learning Environment 100K benchmark by 248%.

References

D. Andre, N. Friedman, and R. Parr. Generalized prioritized sweeping. *Conference in Advances in Neural Information Processing Systems, NeurIPS*, 1997.

M. G. Bellemare, W. Dabney, and R. Munos. A distributional perspective on reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 449–458. PMLR, 2017.

R. E. Bellman. Dynamic programming. In *Princeton, NJ: Princeton University Press*, 1957.

R. I. Brafman and M. Tennenholtz. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 2002.

W. Dabney, M. Rowland, M. G. Bellemare, and R. Munos. Distributional reinforcement learning with quantile regression. In S. A. McIlraith and K. Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 2892–2901. AAAI Press, 2018.

A. de Vries. The growing energy footprint of artificial intelligence. *Joule*, 2023.

C.-N. Fiechter. Efficient reinforcement learning. In *Proceedings of the Seventh Annual ACM Conference on Computational Learning Theory COLT*, 1994.

S. Flennerhag, Y. Schroecker, T. Zahavy, H. van Hasselt, D. Silver, and S. Singh. Bootstrapped meta-learning. *10th International Conference on Learning Representations, ICLR*, 2022.

D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, X. Zhang, X. Yu, Y. Wu, Z. F. Wu, Z. Gou, Z. Shao, Z. Li, Z. Gao, A. Liu, B. Xue, B. Wang, B. Wu, B. Feng, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, D. Dai, D. Chen, D. Ji, E. Li, F. Lin, F. Dai, F. Luo, G. Hao, G. Chen, G. Li, H. Zhang, H. Bao, H. Xu, H. Wang, H. Ding, H. Xin, H. Gao, H. Qu, H. Li, J. Guo, J. Li, J. Wang, J. Chen, J. Yuan, J. Qiu, J. Li, J. L. Cai, J. Ni, J. Liang, J. Chen, K. Dong, K. Hu, K. Gao, K. Guan, K. Huang, K. Yu, L. Wang, L. Zhang, L. Zhao, L. Wang, L. Zhang, L. Xu, L. Xia, M. Zhang, M. Zhang, M. Tang, M. Li, M. Wang, M. Li, N. Tian, P. Huang, P. Zhang, Q. Wang, Q. Chen, Q. Du, R. Ge, R. Zhang, R. Pan, R. Wang, R. J. Chen, R. L. Jin, R. Chen, S. Lu, S. Zhou, S. Chen, S. Ye, S. Wang, S. Yu, S. Zhou, S. Pan, and S. S. Li. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 2025.

J. Hamrick, V. Bapst, A. SanchezGonzalez, T. Pfaff, T. Weber, L. Buesing, and P. Battaglia. Combining q-learning and search with amortized value estimates. In *8th International Conference on Learning Representations, ICLR*, 2020.

H. v. Hasselt, A. Guez, and D. Silver. Deep reinforcement learning with double q-learning. *Association for the Advancement of Artificial Intelligence (AAAI)*, 2016.

M. Hessel, J. Modayil, H. Van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

T. Hubert, J. Schrittwieser, I. Antonoglou, M. Barekatain, S. Schmitt, and D. Silver. Learning and planning in complex action spaces. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 4476–4486. PMLR, 2021.

S. Kakade. On the sample complexity of reinforcement learning. In *PhD Thesis: University College London*, 2003.

S. Kapturowski, V. Campos, R. Jiang, N. Rakicevic, H. van Hasselt, C. Blundell, and A. P. Badia. Human-level atari 200x faster. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL `https://openreview.net/pdf?id=JtC6yOHRoJJ`.

M. Kearns and S. Singh. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 2002.

R. Koenker. Quantile regression. *Cambridge University Press*, 2005.

R. Koenker and K. F. Hallock. Quantile regression. *Journal of Economic Perspectives*, 2001.

E. Korkmaz. Adversarial robust deep reinforcement learning requires redefining robustness. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI*, pages 8369–8377. AAAI Press, 2023.

E. Korkmaz. Understanding and Diagnosing Deep Reinforcement Learning. In *International Conference on Machine Learning, ICML 2024*, 2024.

A. Krishnamurthy, K. Harris, D. J. Foster, C. Zhang, and A. Slivkins. Can large language models explore in-context? In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

X. Lu and B. V. Roy. Information-theoretic confidence bounds for reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2458–2466, 2019.

V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, a. G. Bellemare, A. Graves, M. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, volume 48, pages 1928–1937, 2016.

A. W. Moore and C. G. Atkeson. Prioritized sweeping: Reinforcement learning with less data and less time. *Machine Learning*, 13:103–130, 1993.

U. Nations. Ai has an environmental problem. here's what the world can do about that. *United Nations Environment Programme*, 2024.

J. Oh, M. Hessel, W. M. Czarnecki, Z. Xu, H. van Hasselt, S. Singh, and D. Silver. Discovering reinforcement learning algorithms. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

M. L. Puterman. Markov decision processes: Discrete stochastic dynamic programming. *John Wiley and Sons, Inc*, 1994.

T. Schaul, J. Quan, I. Antonogloua, and D. Silver. Prioritized experience replay. *International Conference on Learning Representations (ICLR)*, 2016.

J. Schmidhuber. A possibility for implementing curiosity and boredom in model-building neural controllers. 1991.

J. Schmidhuber. Artificial curiosity based on discovering novel algorithmic predictability through coevolution. In *Proceedings of the 1999 Congress on Evolutionary Computation, CEC 1999, Washington, DC, USA July 6-9, 1999*, pages 1612–1618. IEEE, 1999. doi: 10.1109/CEC.1999.785467. URL `https://doi.org/10.1109/CEC.1999.785467`.

T. Schmied, J. Bornschein, J. Grau-Moya, M. Wulfmeier, and R. Pascan. Llms are greedy agents: Effects of rl fine-tuning on decision-making abilities. 2025.

J. Schrittwieser, I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, T. Graepel, T. P. Lillicrap, and D. Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588, 2020.

R. Sutton. Learning to predict by the methods of temporal differences. In *Machine Learning*, 1988.

H. van Hasselt. Double q-learning. In *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010.*, pages 2613–2621. Curran Associates, Inc., 2010.

Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Freitas. Dueling network architectures for deep reinforcement learning. *Internation Conference on Machine Learning ICML.*, page 1995–2003, 2016.

C. Watkins. Learning from delayed rewards. In *PhD thesis, Cambridge*. King's College, 1989.

S. Whitehead and D. Ballard. Learning to percieve and act by trial and error. In *Machine Learning*, 1991.

A. Zewe. Explained: Generative ai's environmental impact. *MIT Technological Review*, 2025.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The introduction and abstract of our paper describes the contributions of our paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Our paper provides a theoretical analysis that introduces the basis on how the minimization of the state-action value function will result in higher temporal difference. Furthermore, we provide experimental results in the canonical training benchmarks, and the results verify the theoretical analysis provided in Section 3. Furthermore, we specifically report the temporal difference obtained by the agents in Section 6 in Figure 3 and Figure 6. These results further verify the theoretical analysis provided in Section 3.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The theoretical results are stated with full set of assumptions and a complete proof.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All the necessary details are explained in the supplementary material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The main paper references the algorithms used and describes our algorithm with the theoretical analysis. All the necessary details regarding implementation are explained in the supplementary material.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All the necessary details regarding implementation are provided in the supplementary material. The standard settings were implemented.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All the results reported in the paper contain appropriate information about the statistical significance of the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We test our method even against some of the compute exhaustive methods which was also explained between Line 261 and 265. Furthermore, we state in Section 6 that CoAct TD Learning only requires two lines of additional code, is extremely modular, and can be used as a drop-in replacement for any baseline algorithm that uses the canonical methods between Line 310 and 311.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research does not violate the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: Our work is foundational research. Yet, we believe it might still have potential societal impacts. Particularly, our paper introduces a foundational drop-in replacement for canonical algorithms that results in more efficient learning without additional cost. With the rise of progress in artificial intelligence and the objectives to scale base algorithms to benefit humanity, the energy consumption required by the hardware regarding AI training is estimated to be at the level of an entire country's electricity consumption, i.e. 85 to 134 terawatt hours (Twh) annually, [de Vries, 2023, Nations, 2024, Zewe, 2025]. Regarding the carbon footprint of artificial intelligence and the climate crisis, our paper poses a line of contribution on trying to minimize the environmental damage and impacts of the algorithms we built by achieving significant improvement on the samples that are required to train them.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We reference the algorithms, training environments and the metrics that evaluate algorithms throughout our paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: Our paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Our paper does not involve crowdsourcing nor research with human subjects

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Our paper does not involve crowdsourcing nor research with human subjects

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: Our paper does not include any LLM usage in any form.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.