Continual Optimistic Initialization for Value-Based Reinforcement Learning

Sheelabhadra Dey
Texas A&M University
College Station, USA
sheelabhadra@tamu.edu

James Ault
Texas A&M University
College Station, USA
jault@tamu.edu

Guni Sharon
Texas A&M University
College Station, USA
guni@tamu.edu

ABSTRACT

Comprehensive state-action exploration is essential for reinforcement learning (RL) algorithms. It enables them to find optimal solutions and avoid premature convergence. In value-based RL, *optimistic initialization* of the value function ensures sufficient exploration for finding the optimal solution. Optimistic values lead to *curiosity*-driven exploration enabling visitation of under-explored regions. However, optimistic initialization has limitations in stochastic and non-stationary environments due to its inability to explore "infinitely-often". To address this limitation, we propose a novel exploration strategy for value-based RL, denoted *COIN*, based on recurring optimistic initialization. By injecting a continual exploration bonus, we overcome the shortcoming of optimistic initialization (sensitivity to environment noise). We provide a rigorous theoretical comparison of *COIN* versus existing popular exploration strategies and prove it provides a unique set of attributes (coverage, infinite-often, no visitation tracking, and curiosity). We demonstrate the superiority of *COIN* over popular existing strategies on a designed toy domain as well as present results on common benchmark tasks. We observe that *COIN* outperforms existing exploration strategies in four out of six benchmark tasks while performing on par with the best baseline on the other two tasks.

KEYWORDS

Reinforcement Learning; Exploration Strategies; Optimistic Initialization

ACM Reference Format:
Sheelabhadra Dey, James Ault, and Guni Sharon. 2024. Continual Optimistic Initialization for Value-Based Reinforcement Learning. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 10 pages.

1 INTRODUCTION

Sequential decision problems are commonly modeled as Markov decision processes (MDP) [49]. A solution to an MDP is a policy that maps states to actions such that the resulting sequential behavior is optimal with respect to a given utility function. Reinforcement learning (RL) algorithms [61] are designed to solve MDPs via interactions with the underlying environment. RL algorithms were shown to be effective on a variety of applications such as robotics [2, 37], autonomous driving [25, 29, 36], traffic signal control [9, 10, 72], and drilling operations [5, 35, 68]. An RL agent commonly explores the environment by executing actions and observing outcomes. The acquired knowledge enables the RL agent to reason about the optimal policy. RL algorithms commonly require that the Cartesian product of the state and action spaces is sufficiently explored. Consequently, *Exploration strategies* are usually designed to provide some theoretical assurances regarding the state-action visitation distribution. These assurances result in desirable learning guarantees [7, 8, 26, 60] relating to, policy convergence, policy optimality, speed of convergence (sample efficiency), memory complexity, and regret bounds. Existing exploration strategies can be divided into three categories (following taxonomy by Amin et al. [4]) (C1) sampling-based, (C2) visitation-tracking, and (C3) optimistic initialization-based. Each of these classes has unique benefits and drawbacks with respect to the following properties; (P1) state-action space coverage, (P2) infinite-often visitation, (P3) visitation tracking, and (P4) curiosity.

In this paper, we first study the effectiveness of exploration strategies in a specially designed MDPs denoted *bridge crossing problems* (explained later in Section 3.1). These are a class of grid-world problems, where popular exploration strategies struggle to find the optimal policy efficiently. We then provide empirical evidence that curiosity is a key property that helps overcome the challenges in this problem and is inherently present in optimistic initialization-based exploration strategies. In essence, curiosity encourages the agent to visit unseen states and speed up learning in many cases.

We then present a novel exploration strategy, denoted *COIN*, for Q-learning algorithms [69, 70]. *COIN* performs **c**ontinual **o**ptimistic **in**itialization of the Q-values through scaled optimistic initializations at "appropriate" steps. *COIN* is designed to overcome a major shortcoming in standard optimistic initialization [40, 42, 61] while preserving its beneficial traits. To the best of our knowledge, *COIN* is the first general exploration strategy that satisfies properties P1, P2, and P4 while obviating P3. We demonstrate the implications of satisfying these properties using the designed bridge crossing problem. We then present two variants of *COIN*, (V1) vanilla that is applicable to tabular Q-learning and (V2) dual-*COIN* that is appropriate for Q-learning with approximation. Lastly, we provide an empirical study comparing *COIN* against popular existing exploration strategies on benchmark tasks. We demonstrate that *COIN* is a general exploration strategy that scales to a range of tasks, both with dense and sparse reward functions. In particular, empirical results show that *COIN* performs consistently better or at par with existing strategies on domains with sparse feedback.

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

2 PRELIMINARIES

2.0.1 Markov decision processes: Stochastic control problems are commonly formulated as Markov decision processes (MDPs) [49] represented by $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, R, \gamma \rangle$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ is the transition function, $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ is the reward function, and γ is the discount factor. An agent is assumed to follow an internal policy, $\pi : \mathcal{S} \rightarrow \mathcal{A}$, which maps states to actions. At each timestep, t, executing a visited action, a_t, at the current state, s_t, leads the agent to a new state, s_{t+1}, and results in a feedback signal, $r_t = R(s_t, a_t)$, representing the immediate utility gained from executing a_t at s_t. A finite sequence of actions starting from an initial state s_0,[1] results in an *episode*, $\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_T)$. The cumulative discounted rewards thus obtained over τ, also known as the episodic *return*, is defined as $\mathcal{R}(\tau) = \sum_{t=0}^{T} \gamma^t R(s_t, a_t)$. The expected return for a policy π is $J(\pi) = \mathbb{E}_{\tau \sim \pi}[\mathcal{R}(\tau)]$. The *optimal policy*, π^*, is the policy that maximizes J, i.e., $\pi^* = \arg\max_\pi J(\pi)$.

2.0.2 Q-learning: The *action-value function* of π, denoted $Q^\pi(s_t, a_t)$, is the expected return over trajectories generated by executing a_t at s_t and then following π from s_{t+1} onward. The optimal Q-function, $Q^{\pi^*}(s_t, a_t)$ is concisely denoted by $Q^*(s_t, a_t)$. Off-policy Q-learning algorithms update Q^* at each timestep via *temporal difference* (TD) learning [67, 69, 70], given by

$$Q^*_{(k+1)}(s_t, a_t) := Q^*_{(k)}(s_t, a_t) + \alpha_{(k)} \left(r_t + \gamma \max_{a'} Q^*_{(k)}(s_{t+1}, a') - Q^*_{(k)}(s_t, a_t) \right) \quad (1)$$

where α denotes the learning rate and k denotes the Q-update iteration for (s_t, a_t) pairs. The term, $r_t + \gamma \max_{a'} Q^*_{(k)}(s_{t+1}, a') - Q^*_{(k)}(s_t, a_t)$, is also known as the TD-error. Q-learning converges to π^* with probability 1 provided that the following conditions hold, (1) the learning rate is *well-behaved*, i.e., $\sum_k \alpha_{(k)} = \infty, \sum_k \alpha^2_{(k)} < \infty$, and, (2) every state-action pair is visited *infinitely often*, i.e., every (s, a) pair is visited an infinite number of times at the limit [13, 32]. While addressing Condition (1) is trivial, a plethora of effective exploration strategies [34, 60, 67] was previously presented for addressing Condition (2).

2.0.3 Off-policy learning: RL algorithms are designed to optimize a target policy, π. Nonetheless, they might use a different policy, denoted η, to explore the environment. We refer to η as the *exploration strategy* which is a mapping from states to actions. RL algorithms which provide policy convergence guarantees only for the case where $\eta \equiv \pi$ are known as *on-policy* algorithms [43, 54, 55], else (allowing η to differ from π), they are denoted *off-policy* [30, 38, 44]. Q-learning is a prominent example of an *off-policy* algorithm. While the target policy is defined by $\pi(s) := \arg\max_a Q(s, a)$, in many cases the exploration strategy is different. For example, Q-learning is often paired with an ϵ-greedy exploration strategy [60].

2.1 Related work

We follow the taxonomy presented by Amin et al. [4] to briefly discuss the major advantages and shortcomings of common exploration strategy classes for Q-learning.

[1] In general, s_0 might be sampled from an initial state distribution.

2.1.1 `C1`: *Sampling-based exploration:* $\eta(s) = a \sim f(Q(s, \cdot))$*, for a probability density function,* f*.*

In sampling-based exploration strategies, actions are randomly visited, e.g., uniformly (ϵ-greedy) [60], based on a distribution derived from TD-errors [61, 65, 66], or from a *Boltzmann* distribution over the Q-values [11, 39, 70, 71]. These strategies guarantee that all state-action pairs will be visited infinitely often. However, prior work has shown that such exploration strategies, in practice, are sample inefficient in finding the optimal solution in long horizon tasks, possibly hampering convergence to π^* [24, 58].

2.1.2 `C2`: *Uncertainty-based exploration:* $\eta(s) = \arg\max_a [Q(s, a) + U(s, a)]$*, for an uncertainty estimator,* U*.*

Adding an exploration bonus to the Q-values based on the *Optimism in the Face of Uncertainty* (OFU) principle [33, 64] is a prominent example of this category of exploration strategies. *Upper Confidence Bounds* (UCBs) [6] keeps a count of the state-action visitation [33, 50, 59] or empirical estimates of reward and transition probability [7] among many others as a proxy for exploration bonus. Many curiosity-driven exploration [12, 17, 47] methods also fall under this category. While providing desired performance guarantees, such approaches come at the cost of additional memory requirements for storing state-action visitation counts [6, 7]. Alternatively, they are heavily dependent on the accuracy of an estimator approximating state-action visitation counts [28, 63].

2.1.3 `C3`: *Optimistic initialization-based exploration:* $\eta(s) = \arg\max_a Q(s, a)$.

This is a unique class of exploration strategies as $\eta(s)$ equals the target policy of Q-learning. Exploration strategies belonging to this class induce effective exploration through perturbation(s) to the Q-values. They promote exploration via *optimistic initialization* in which unvisited (s, a) pairs are assumed to lead to the best possible return [15, 61]. A few optimistic initialization methods use estimates of the environment dynamics and reward function to assign optimistic values uniformly to all states [15] or only to unknown states [26, 62]. Alternatively, adding a fixed "bonus" to the Q-function once at the start of training via reward shaping has been shown to induce optimistic initialization [58]. Sun et al. [58] demonstrate, through empirical results, that when used in conjunction with a sampling-based strategy, it results in improved exploration leading to better sample efficiency in Q-learning-based algorithms. However, it is unclear whether their proposed approach would perform similarly in the absence of a sampling-based strategy (they use ϵ-greedy in conjunction with their proposed approach). Using any sampling-based strategy, in theory, introduces limitations (as mentioned previously in this section).

3 PROPERTIES OF EXPLORATION STRATEGIES

We consider 4 properties, `P1-4`, of exploration strategies. We use $Pr_i(s, a)$ to denote the probability of visiting a state-action pair (s, a) during episode i, i.e., $Pr[(s, a) \in \tau_i]$, under a given exploration strategy. Note that exploration strategies might evolve over time. Consequently, for a given (s, a) pair, $Pr_i(s, a)$ might be different from $Pr_j(s, a)$ where $i \neq j$.

Table 1: Comparison of exploration strategies. We specify the category under which an exploration strategy ('Exp. strategy') falls as described in Section 2.1. We highlight which of the properties among coverage, infinite often visitation ('∞-often'), and curiosity is satisfied by each of the exploration strategies. We indicate whether each of them requires visitation tracking ('No vis. track.'). 'Hyperparam.' is the hyperparameter essential to the exploration strategy.

Exp. strategy	*Category*	*Coverage*	*∞-often*	*No vis. track.*	*Curiosity*	*Hyperparam.*
ϵ-greedy	C1	✓	✓	✓	×	ϵ
Boltzmann	C1	✓	✓	✓	×	Temperature
UCB	C2	✓	✓	×	✓	Bonus coeff. (C_p)
Optimistic init.	C3	✓	×	✓	✓	Initial Q-values
COIN	C3	✓	✓	✓	✓	Bonus (b)

Definition 1 (P1: Coverage at limit)**.** An exploration strategy is said to provide *coverage at limit* if $\forall(s,a) \in \mathcal{S}\times\mathcal{A}$, $\prod_{i=0}^{\infty}(1 - Pr_i(s,a)) = 0$. [2]

Any exploration strategy that does not provide coverage at the limit cannot guarantee convergence to the optimal policy as some actions along the optimal sequence of actions might never have been visited [61].

Definition 2 (P2: Infinite-often)**.** Following the Borel-Cantelli lemma [19, 22], an (s,a) pair is said to be visited *infinitely often* if $\sum_{i=0}^{\infty} Pr_i(s,a) = \infty$. An exploration strategy is said to be 'Infinite-often' if every (s,a) pair is visited infinitely often.

Note that an exploration strategy with the infinite-often property implies that it also has the property of coverage at the limit but not vice-versa. Various RL algorithms based on Watkins [70] and Singh et al. [56] guarantee convergence to the optimal policy if infinite-often visitation is satisfied [51]. Note that there is a difference between the convergence of η and π. While η is not required to, and usually does not converge to π^*, π can still converge to π^*.

Definition 3 (P3: Visitation tracking)**.** An exploration strategy using *visitation tracking* continually tracks (or estimates) the number of times each (s,a) pair was visited.

See C2 in Section 2.1.2 for examples of exploration strategies that track the number of (s,a) visits either explicitly in tabular MDPs or implicitly using an estimator in continuous state spaces.

Definition 4 (P4: Curiosity)**.** An exploration strategy possesses *curiosity* if action visitation precedence at a given state is inversely correlated, in expectation, with the number of times that action was visited. That is, for any state s and action a, the probability (sampling-based exploration) or score (hardmax exploration) for visiting a at s monotonically decreases, in expectation, with the number of visits to a. In general, this may also occur after a bounded number of initial visits to a at s.

Exploration strategies possessing the property of curiosity have been shown to outperform vanilla sampling-based methods on many hard-exploration tasks [18]. However, previous work provide vague definitions for curiosity. For example, the seminal curiosity work [47] defines this property as "intrinsic motivation/reward". Such reward is further defined as "Most formulations of intrinsic reward can be grouped into two broad classes: 1) encourage the agent to explore 'novel' states [12, 41, 48] or, 2) encourage the agent to perform actions that reduce the error/uncertainty in the agent's ability to predict the consequence of its own actions (i.e. its knowledge about the environment) [20, 31, 45, 52, 53, 57]". As a result, we provide a precise definition of curiosity (P4) that is general (applying to known curiosity-based algorithms) while being unambiguous. In common sampling-based exploration strategies (ϵ-greedy and Boltzmann), Q-values for specific (s,a) pairs may increase over visitations. This would lead to a violation of P4. On the other hand, exploration in UCB-based algorithms, for instance, UCB1 [6] defines $\eta(s) = \arg\max_a \left[Q(s,a) + C_p \sqrt{\frac{\ln n}{N(s,a)}} \right]$, where n is the total number of learning timesteps taken by the RL agent and $N(s,a)$ is the number of times an (s,a) pair has been visited. If C_p is set to a large enough value, the score for visiting a at s monotonically decreases with the number of visits to a, hence satisfying P4. In optimistic initialization, with sufficiently large initialization of Q-values, the expected TD-error is non-positive. [3] Since action visitation in optimistic initialization relies only on a hardmax over the Q-values, $Q(s,a)$ for an action a visited at s with a small α, in expectation, decreases monotonically following Equation 1. Hence, optimistic initialization satisfies P4.

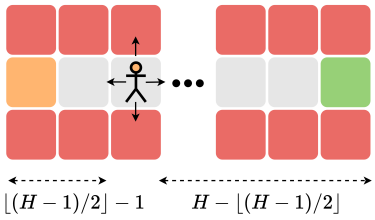
Table 1 presents a summary of exploration strategies and their associated properties.

3.1 Motivating example: the bridge problem

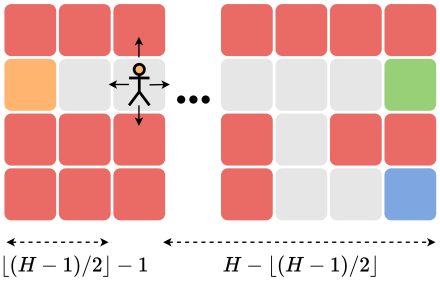
We further demonstrate the significance of curiosity on a toy example containing a high proportion of terminal states with undesirable outcomes. Consider the gridworld shown in Figure 1, which we refer to as the *bridge crossing* problem. We model it as an MDP. In the stationary bridge problem illustrated in Figure 1a, there are two terminating goals; one optimal (green cell at the right extreme) and another suboptimal (orange cell at the left extreme) along with many terminating negative outcomes for falling off the bridge (red cell). Assuming that the start state is fixed, an issue in this domain is that it is easy for the agent to converge on a suboptimal policy leading to the suboptimal goal, or take a long time to learn the optimal policy. This issue is particularly exacerbated when the horizon of the bridge, H, is increased and sampling-based exploration strategies are deployed.

[2] This paper assumes that all states in $\mathcal{S}$ can be reached with non-zero probability.

[3] See Even-Dar and Mansour [26] for the sufficiently large Q-initialization required to guarantee convergence.



(a) Stationary bridge (H) problem.



(b) Non-stationary bridge (H) problem.

Figure 1: The bridge (H) problem, where H corresponds to the number of columns in the grid. The red, orange, and green cells are terminal states with rewards of -10, 10, and 30 respectively. Non-terminal cells furnish an immediate reward of 0. The start state is fixed at the cell marked with the stick figure (agent) and the column number is decided using the rule $\lfloor (H-1)/2 \rfloor$. The arrows represent the 4 actions, 'up', 'down', 'left', and 'right', that an agent can take at each cell. (b) At some point in time, the stationary bridge (H) changes to a non-stationary bridge (H). The blue cell is a new terminal state with a reward of 40.

We demonstrate the shortcomings of popular exploration strategies listed in Table 1 using the bridge problem and explain how optimistic initialization addresses them.

3.1.1 Sampling-based: In this family of exploration strategies (Section 2.1.1) the probability of the agent following the greedy action over a sequence of states diminishes exponentially as T increases. Reaching the optimal goal in this example clearly requires $T \geq H - \lfloor (H-1)/2 \rfloor$ steps. The probability of the agent reaching the optimal goal state by following the optimal greedy policy in an episode is $\lim_{T\to\infty} \prod_{t=0}^{T} Pr(a_t|s_t, \pi^*) = \lim_{T\to\infty}(1-\epsilon)^T = 0$, where ϵ is the non-zero probability of selecting a non-greedy action at each step. That is, the probability of reaching the optimal goal state, diminishes exponentially with the bridge length even when π^* is known. This results in sampling-based exploration strategies becoming sample-inefficient in similar settings [27, 58].

3.1.2 Optimistic initialization-based: Since optimistic initialization possesses curiosity, it encourages the RL agent to explore unvisited (s, a) pairs. This, in practice, results in optimistic initialization being sample-efficient on the bridge problems.

3.1.3 Limitations of optimistic initialization: Optimistic initialization is known to potentially converge to suboptimal solutions in stochastic environments since it does not hold the infinite-often property [26, 27]. In such scenarios, an appropriate Q-initialization may be a tedious hyperparameter to tune. Furthermore, optimistic initialization has limited efficacy in non-stationary MDPs as exploration ceases when Q-values converge.

Algorithm 1: *COIN*

Input: discount factor γ, learning rate α, maximum learning steps T
Initialize: action-value function Q

for $t \leftarrow 1$ **to** T **do**
  $a_t \leftarrow \arg\max_a Q(s_t, a)$;
  Execute a_t and observe reward r_t and state s_{t+1};
  $y_t \leftarrow \begin{cases} r_t, & s_{t+1} = s_T \\ r_t + \gamma \max_{a'} Q(s_{t+1}, a'), & \text{else} \end{cases}$;
  $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(y_t - Q(s_t, a_t))$;
  if *t = appropriate b addition step* **then**
    ; `/* COIN iteration */`
    $b \leftarrow \max_s \{\max_a Q(s, a) - \min_a Q(s, a)\} + \epsilon_b$;
    $Q \leftarrow Q + b$; `/* Bonus (b) update */`
  end
end

4 CONTINUAL OPTIMISTIC INITIALIZATION

We turn to present our proposed **c**ontinual **o**ptimistic **in**itialization exploration strategy, denoted *COIN*, an optimistic initialization-based exploration strategy, to address the shortcomings of "vanilla" optimistic initialization. As such, *COIN* follows $\eta(s) = \arg\max_a Q(s, a)$ while periodically augmenting the Q-values in a way that achieves effective exploration.

Based in Q-learning, *COIN* guarantees that the Q-values for each (s, a) pair along all the *greedy trajectories* will converge after a finite number of Q-value updates [69]. We define a greedy trajectory as a sequence of (s, a) pairs obtained when following the greedy policy, i.e., $\pi^g(s) = \arg\max_a Q(s, a)$. Once Q-values converge along all greedy trajectories, we add a positive bonus, b, to the Q-function across all $(s, a) \in \mathcal{S} \times \mathcal{A}$. We term the step at which such convergence occurs as an "appropriate b addition step". Intuitively, adding b is analogous to initializing the Q-function with optimistic values, albeit continually. A pseudocode of *COIN* has been provided in Algorithm 1 respectively.

Definition 5 (Appropriate b addition step)**.** We define an appropriate b addition step as the training step at which the maximum TD-error along the greedy trajectories is within an ϵ threshold.

We also call an appropriate b addition as a *COIN iteration* to reduce verbosity. Note there may be multiple episodes between two consecutive *COIN* iterations.

4.0.1 b setting in COIN:. Although any positive b is sufficient for the theoretical properties of *COIN* to hold, empirically we observe that setting b such that

$$b = \max_{s \in \mathcal{S}^\eta} \left\{ \max_{(s,a \in \mathcal{A})} Q(s, a) - \min_{(s,a \in \mathcal{A})} Q(s, a) \right\} + \epsilon_b,$$

where $\mathcal{S}^{\eta}$ is the set of states reachable under η and ϵ_b is a small positive value, results in sample-efficient learning. This heuristic makes all actions along the greedy trajectory to be optimistic with respect to the current π^g.

More formally, *COIN* follows: every *COIN* iteration, set $\forall(s,a) \in \mathcal{S}\times\mathcal{A}, Q(s,a) = Q(s,a)+b$. Adding b to the Q-function continually at an infinite number of *COIN* iterations ensures infinite-often visitation (P2) (see Lemma 1).

4.1 Theoretical properties of *COIN*

The following theoretical properties and analysis apply to tabular Q-learning. While *COIN* can be coupled with function approximation as we show later in section 5.2, our theoretical claims might no longer be guaranteed.

Lemma 1 (*COIN* ensures infinite often visitation (P2)). *Assuming that the Q-function is updated at each COIN iteration, any* $(s,a) \in \mathcal{S}\times\mathcal{A}$ *must be visited infinitely often under COIN, i.e.,* $\sum_{i=0}^{\infty} Pr_i(s,a) = \infty$.

Proof. We prove via induction that any (s,a) pair must be visited within a bounded number of *COIN* iterations. As a result, an infinite number of iterations would result in infinite-often visitation.

Base case: *(every action,* $a \in \mathcal{A}$*, will be visited at s after a bounded number of visitations to s, and specifically for* $s = s_0$*)*: By contradiction, we assume some action, a', at s will not be visited within a finite number of *COIN* iterations.

Notation:

- $Q^{(old)}$: Q-function at the end of the *COIN* iteration when a' was last visited at s or iteration 0 if it was never visited.
- $Q^{(m)}$: Q-function after m *COIN* iterations since a' was last visited.
- a^g: $\arg\max_a Q^{(m)}(s,a)$.

Since a' hasn't been visited by the greedy policy in m *COIN* iterations,

$$Q^{(m)}(s_0,a^g) > Q^{(m)}(s_0,a'),\ \forall m \tag{2}$$

$$Q^{(m)}(s_0,a') \geq Q^{(old)}(s_0,a') + m\epsilon_b \tag{3}$$

As $Q^{(m)}$ is updated following Equation 1, for a low enough α,

$$Q^{(m)}(s_0,a^g) \leq Q^*(s_0,a^*) \leq C \tag{4}$$

Since, $\mathcal{R}$ is bounded, C is finite. From Equations (2) and (3) it must be that $\exists\ m$ such that,

$$Q^{(old)}(s_0,a') + m\epsilon_b > C \tag{5}$$

Equations (2) and (3), in conjunction, contradict (5) as they imply $C < C$. Hence, a' must be visited within a finite number *COIN* iterations. Further, since $b > 0$, an infinite number of *COIN* iterations will result in a' being visited infinitely often.

Induction assumption: *(every action,* $a \in \mathcal{A}$*, at* s_{n-1} *will be visited after a bounded number of visitations to* s_{n-1}*, where* s_{n-1} *is any state reachable from* s_0 *in n steps.)*

Induction step: Following the (general claim) base case, every action at s_{n-1} will be visited infinitely often. Hence, for any state, s_n, if $\mathcal{P}(s_n|s_{n-1},a) > 0$, it is reachable in $(n+1)$ steps from s_0 with a non-zero probability. As such, the general base case can be invoked again. That is, any a' will be visited at s_n an infinite number of times. □

Corollary 1 (*COIN* satisfies coverage at limit (P1)). *Assuming that the Q-function is updated at each COIN iteration, any* $(s,a) \in \mathcal{S}\times\mathcal{A}$ *must be selected under COIN, i.e.,* $\prod_{i=0}^{\infty}(1 - Pr_i(s,a)) = 0$.

Corollary 1 follows from Lemma 1 since infinite-often visitation implies coverage at limit.

Remark 1. *COIN* has the property of curiosity.

After a finite number of *COIN* iterations, $Q(s,a)$ for any unvisited (s,a) pair must be overestimated as its Q-values are continually inflated. For a small enough α, $Q(s,a)$ for visited (s,a) pairs are also overestimated after a finite number of *COIN* iterations as any reduction in their Q-values will be overcome by the addition of b to the Q-function. Once, the Q-values for all (s,a) pairs are overestimated, they are updated following Equation 1 with a negative TD-error, in expectation. This leads to a monotonic decrease in the Q-values, in expectation, with each visitation to a.

4.2 Extension of *COIN* to non-stationary MDPs

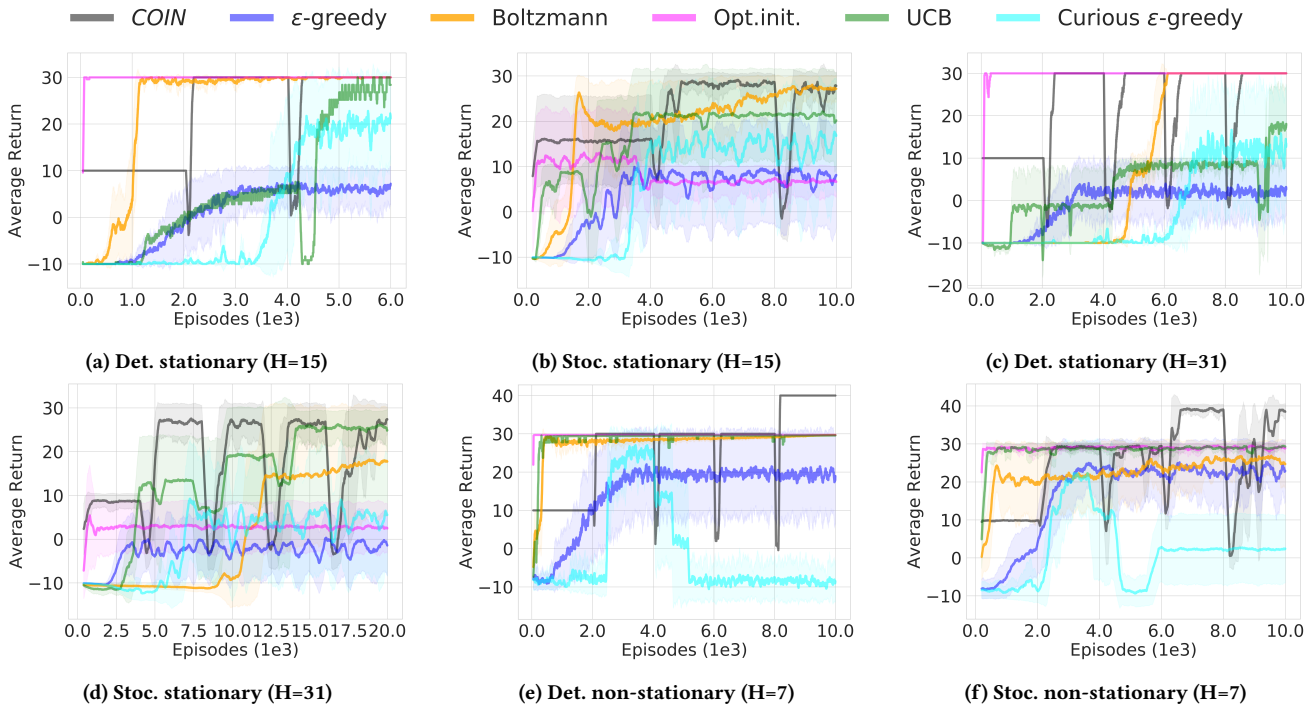
COIN can be interpreted as a model-free restarting strategy for non-stationary MDPs (if b were held fixed and equivalent to optimistic initialization). Such strategies have been suggested previously for non-stationary episodic MDPs and multi-armed bandit problems [3, 14]. *COIN* may be a suitable candidate for non-stationary MDPs since it has infinite-often property. However, convergence of RL algorithms in non-stationary MDPs is an active branch of research as optimal convergence proofs of RL algorithms often assume stationary transition and reward functions [46]. Notably, prior work has shown value function-based RL to produce policies "close" to optimal in a special class of non-stationary MDPs [23]. Section 4.3.2 briefly discusses the effectiveness of *COIN* on a toy non-stationary MDP. However, an in-depth analysis is left for future work.

4.3 *COIN* on the bridge problem

We provide empirical results for *COIN* and the exploration strategies presented in Table 1 when coupled with tabular Q-learning on a few variants of the bridge problem. Later in Section 5.2, we present results on domains with continuous state spaces.

Since *COIN* has a deterministic policy, it avoids issues faced by sampling-based exploration strategies. *COIN*'s infinite-often visitation property overcomes the limitation of optimistic initialization. Note that it is expected to observe "dips" in the learning curves with *COIN* each time a b addition is performed as it perturbs the Q-values and encourages the greedy policy to explore. In order to clearly present these trends, we perform a b addition after a fixed number of episodes instead of our proposed b addition step from Definition 5. ϵ_b and ϵ are set to 0.1 and 0.05 respectively.

4.3.1 Results for stationary bridge: Figures 2a and 2b show results for the case where H=15 with deterministic and stochastic transitions respectively. In the deterministic case, it can be observed that all strategies except ϵ-greedy find the optimal solution within 6,000 episodes of learning. While in the stochastic case, Optimistic init. fails as learning progresses since it suffers from the issue explained



(a) Det. stationary (H=15)

(b) Stoc. stationary (H=15)

(c) Det. stationary (H=31)

(d) Stoc. stationary (H=31)

(e) Det. non-stationary (H=7)

(f) Stoc. non-stationary (H=7)

Figure 2: Learning curves of exploration strategies listed in Table 1 on the bridge problem. The shaded region represents 2 standard deviations of the return over 5 trials. The curves have been smoothed for visual clarity. 'Det.' and 'Stoc.' refer to a deterministic and stochastic MDP respectively. The probability of the agent 'slipping' is set to 0.01 in the stochastic setting. (e)-(f) Non-stationarity is introduced in episode 5,000. A b addition is performed after every 2,000 and 4,000 episodes of learning in the deterministic (a), (c), (e) and stochastic (b), (d), (f) cases respectively. In all the learning curves, the large dips in *COIN* occur when a b addition is performed.

in Section 3.1.3 whereas *COIN* does not since it visits (s, a) pairs infinitely often via continual b additions. The results in Figures 2c and 2d, where H=31, offer similar insights but in addition, highlight the advantages of optimistic initialization (Figure 2c) and *COIN*. We speculate that the advantage stems from their curiosity property.

Note that we use a count-based intrinsic reward proposed in Bellemare et al. [12] and combine it with ϵ-greedy exploration (*Curious ϵ-greedy*). This approach leads to improved learning efficiency over ϵ-greedy demonstrating that curiosity is helpful in this domain.

4.3.2 Results for non-stationary bridge. Although optimistic initialization with Q-learning can converge to an optimal policy, it is known to potentially reach a local optimum when the MDP is non-stationary. Extending the bridge problem, let a new optimal goal emerge at a new location. At the same time, a new path is added such that it leads to the new optimal solution from the start state. An illustration of the new problem is shown in Figure 1b. We allow Q-learning to first converge on the old setting and require it to adjust to the new setting. The results in Figures 2e and 2f on the *non-stationary bridge (H=7)* tasks show that all methods except *COIN* fail to reach the optimal solution. Since optimistic initialization does not visit state-action pairs infinitely often, it fails to find the new optimal solution whereas *COIN* succeeds in doing so.

These results positively answer empirical questions on the capability of *COIN* in solving a simple instance of a non-stationary MDP by updating b continually, demonstrating its ability to successfully address a major drawback of "vanilla" optimistic initialization.

5 DUAL-*COIN*

We present dual-*COIN*, an alternative view of *COIN*, where, instead of adding b to the Q-function, we modify the reward function to induce a similar effect. Consider the case where for $b > 0$ the reward function is updated such that $R_{-b} := R - b$. That is we subtract a positive constant from the reward function. It can be shown that the Q-function learned using Q-learning converges to $Q_{-b} := Q - \frac{b}{1-\gamma}$, where Q is the Q-function learned on R (see Lemma 2 for proof). A similar observation was also presented in prior work [42, 58] for the optimistic initialization case.

Note that, when the MDP has a large or continuous state space, a function approximator is commonly utilized to represent the Q-function. Updating the Q-function to $Q(s, a) := Q(s, a) + b$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$ might become intractable in this case. Dual-*COIN* makes *COIN* practical in such settings.

5.1 Theoretical properties of dual-*COIN*

As stated in section 4.1, we remind the reader that the following theoretical properties and analysis apply to tabular Q-learning.

Lemma 2 (Additive property of Q-values). *If a constant, b, is uniformly added to R, Q-values updated using Q-learning converge to $Q + \frac{b}{1-\gamma}$ for an infinite-horizon MDP.*

Proof.

$$Q(s_t, a_t) = \mathbb{E}\left[r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1})\right]$$

Let $R(s,a) := R(s,a) + b, \forall (s,a) \in \mathcal{S} \times a \in \mathcal{A}$ and Q_{+b} be the new Q-function.

$$\begin{aligned}
Q_{+b}(s_t, a_t) &= \mathbb{E}\left[r_t + b + \gamma \max_{a_{t+1}} Q_{+b}(s_{t+1}, a_{t+1})\right] \\
&= \mathbb{E}\left[r_t + b + \gamma\left(r_{t+1} + b + \max_{a_{t+2}} Q_{+b}(s_{t+2}, a_{t+2})\right)\right] \\
&= b\sum_{t=1}^{T}\gamma^t + \mathbb{E}\left[r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1})\right] \\
&= b\frac{1-\gamma^T}{1-\gamma} + Q(s_t, a_t)
\end{aligned}$$

For an infinite horizon MDP, i.e., $T \to \infty$,

$$Q_{+b}(s_t, a_t) = Q(s_t, a_t) + \frac{b}{1-\gamma}$$

□

5.1.1 *b setting in dual-COIN:.*

$$b_{dual} = \left(\frac{1-\gamma}{1-\gamma^T}\right)\max_{s \in \mathcal{S}^\eta}\left\{\max_{(s,a\in\mathcal{A})} Q(s,a) - \min_{(s,a\in\mathcal{A})} Q(s,a)\right\} + \epsilon_b,$$

where $\mathcal{S}^\eta$ is set of the states visited under η so far and ϵ_b is a small positive value.

Lemma 3 (Dual-*COIN* ensures infinite often visitation (P2)). *Assuming that R is updated at each COIN iteration, any $(s,a) \in \mathcal{S}\times\mathcal{A}$ must be visited infinitely often under dual-COIN, i.e., $\sum_{i=0}^{\infty} Pr_i(s,a) = \infty$.*

Proof. We prove via induction that any (s,a) pair must be visited within a bounded number of *COIN* iterations. [4] As a result, an infinite number of iterations would result in infinite-often visitation.

Base case: *(every action, $a \in \mathcal{A}$, will be visited at s after a bounded number of visitations to s, and specifically for $s = s_0$)*:

Notation:

- $Q^{(m)}$: Q-function after m *COIN* iterations since a' was last visited or never visited.
- a^g: $\arg\max_a Q^{(m)}(s,a)$.

Let a' be unvisited by the greedy policy in m *COIN* iterations,

$$Q^{(m)}(s_0, a') < \max_a Q^{(m)}(s_0, a),\ \forall m \tag{6}$$

[4] In dual-*COIN*, a *COIN* iteration corresponds to subtracting b_{dual} from R.

As $Q^{(m)}(s_0, a^g)$ is updated following Equation 1, from Lemma 2, for a low enough α,

$$\max_a Q^{(m)}(s_0, a) < Q^*\left(\arg\max_a Q^{(m)}(s_0, a)\right) - m\epsilon_b\left(\frac{1-\gamma^T}{1-\gamma}\right) \tag{7}$$

Since, $\mathcal{R}$ is bounded, Q^* is bounded. From Equations (6) and (7) it must be that $\exists\ m$ such that,

$$Q^{(m)}(s_0, a') = \max_a Q^{(m)}(s_0, a) \tag{8}$$

Hence, a' must be visited within a finite number of dual-*COIN* iterations. Further, since $b_{dual} > 0$, an infinite number of *COIN* iterations will occur resulting in a' being visited infinitely often.

Induction assumption: *(every action, $a \in \mathcal{A}$, at s_{n-1} will be visited after a bounded number of visitations to s_{n-1}, where s_{n-1} is any state reachable from s_0 in n steps.)*

Induction step: Following the (general claim) base case, every action at s_{n-1} will be visited after a bounded number of visitations to s_{n-1}. Hence, for any state, s_n, if $\mathcal{P}(s_n|s_{n-1}, a) > 0$, it is reachable in $(n+1)$ steps from s_0 with a non-zero probability. As such, the same argument as the one presented in the base case can be applied. Thus, a' will be visited at s_n an infinite number of times. □

Corollary 2 (Dual-*COIN* satisfies coverage at limit (P1)). *Assuming that R is updated at each COIN iteration, any $(s,a) \in \mathcal{S}\times\mathcal{A}$ must be selected under dual-COIN, i.e., $\prod_{i=0}^{\infty}(1 - Pr_i(s,a)) = 0$.*

Corollary 2 follows from Lemma 3 since infinite often visitation implies coverage at the limit.

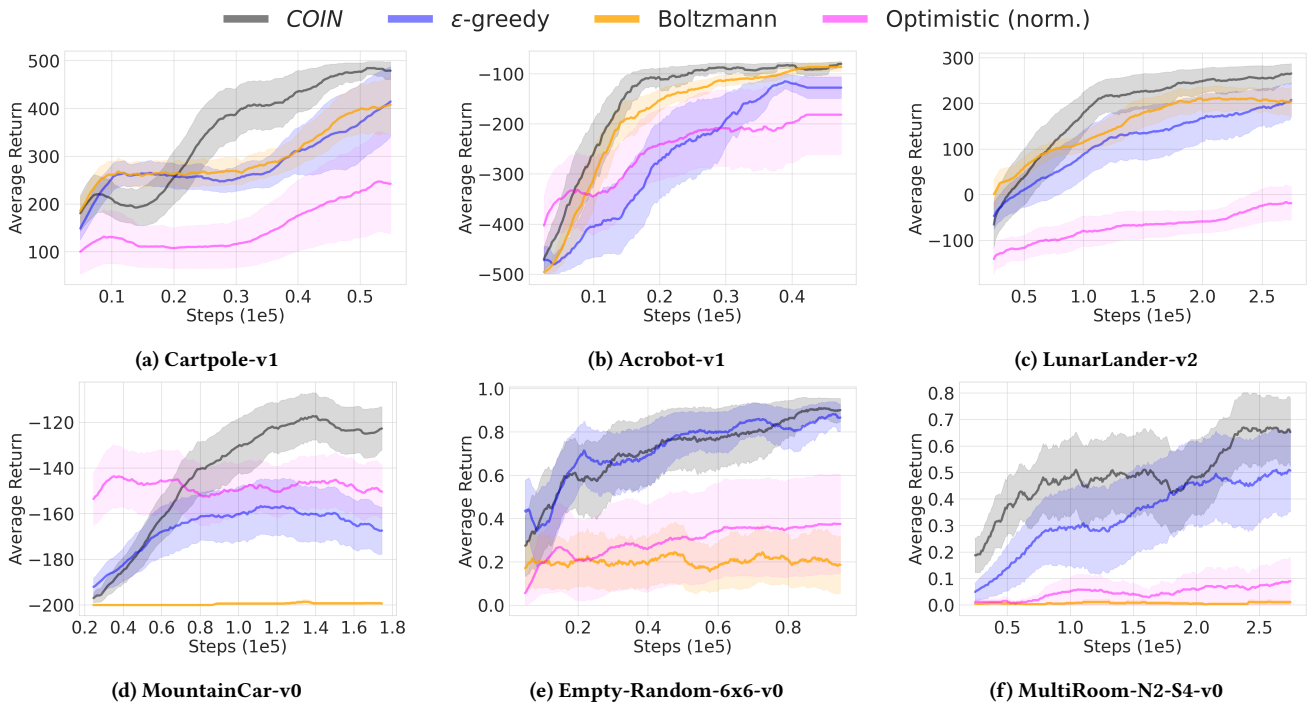
Remark 2. Dual-*COIN* has the property of curiosity.

After a finite number of *COIN* iterations, subtracting b from R continually leads to underestimated Q-values along the greedy trajectories. Specifically, as Q-values of visited (s,a) pairs decrease, the expected TD-error is negative. Since action visitation in *COIN* relies only on a hardmax over the Q-values, similar to optimistic initialization, $Q(s,a)$ for an action a visited at s with a small α, in expectation, decreases monotonically.

5.2 Empirical study

5.2.1 Domains: We evaluate dual-*COIN* on 6 benchmark domains with continuous state spaces, from the OpenAI gym [16] and MiniGrid [21], covering both dense and sparse reward functions. The domains with sparse reward functions were also used in Sun et al. [58] to demonstrate the effectiveness of their proposed optimistic initialization-based approach on tasks with sparse rewards. In 3 of the 4 OpenAI gym domains, i.e., 'Cartpole-v1', 'Acrobot-v1', and 'LunarLander-v2' have dense reward functions. The 4th domain, 'MountainCar-v0', has a sparse reward function. A positive reward is received by the agent only upon reaching the goal state, and a negative reward at all other states. In the MiniGrid domains, i.e., 'Empty-Random-6x6-v0' and 'MultiRoom-N2-S4-v0' the state is only partially observable (agent-local view of the grid only). MiniGrid domains also have a sparse reward function where a positive reward is given only on reaching the goal state and 0 otherwise.

5.2.2 Baselines. We use exploration strategies belonging to categories C1 (sampling-based), i.e., ϵ-greedy, and Boltzmann, and C3



(a) Cartpole-v1 (b) Acrobot-v1 (c) LunarLander-v2

(d) MountainCar-v0 (e) Empty-Random-6x6-v0 (f) MultiRoom-N2-S4-v0

Figure 3: Learning curves of exploration strategies with vanilla DQN on benchmark domains. The shaded region represents 2 standard deviations of the return over 5 trials. The curves have been smoothed for visual clarity and hence dual-*COIN* dips may not be prominent. Dual-*COIN* consistently performs on par or better compared to the baseline exploration strategies.

(optimism-based), i.e., Optimistic (norm.) [42], as baselines for comparison since, similar to *COIN*, they do not require learning any additional estimator. We employ vanilla deep Q-network learning (DQN) [44] adapted from Achiam [1] as the underlying learning algorithm for all of the baselines. These experiments aim to demonstrate that *COIN* is a general exploration strategy and can perform competitively with respect to popular exploration strategies without additional assumptions.

5.2.3 Results. The graphs presented in Figure 3 provide a positive answer to the generalizability and competitiveness of dual-*COIN*. Dual-*COIN* outperforms the baselines in terms of sample efficiency on 4 out of 6 domains, namely, 'CartPole-v1' (Figure 3a), 'LunarLander-v2' (Figure 3c), 'MountainCar-v0' (Figure 3d), and 'MultiRoom-N2-S4-v0' (Figure 3e) while performing on par in the remaining domains. In particular, we notice that in the sparse reward tasks, dual-*COIN* consistently has better sample efficiency. We believe that dual-*COIN*'s property to induce curiosity plays a major role in this. Complete details of the domains, hyperparameter settings of dual-*COIN* and the baselines, and the codebase for these experiments are available at https://github.com/Pi-Star-Lab/coin. [5]

[5]In practice, we perform a *COIN* iteration when the average episodic returns are fairly stable. That is the dispersion index of the returns is less than a threshold.

6 SUMMARY

We present a novel optimistic initialization-based approach, *COIN*, possessing a unique set of properties associated with effective exploration strategies. It performs continual optimistic initialization of Q-values to overcome the limitations of optimistic initialization in stochastic and non-stationary environments. We provide theoretical evidence of *COIN* possessing infinite-often visitation property which helps it overcome these limitations. We validate our claims on the bridge crossing problem. Compared to common existing exploration strategies, we demonstrate the superiority of *COIN* in solving long-horizon stochastic and non-stationary bridge problems. Extending *COIN* to continuous state spaces, we then present dual-*COIN*. Empirical results on 6 benchmark domains support our claim that *COIN* is a general exploration strategy by outperforming 3 common existing exploration strategies on 4 out of 6 domains. We observe that *COIN* is more effective than these strategies in sparse reward benchmark domains which we speculate is a result of its curiosity-driven behavior.

ACKNOWLEDGMENTS

The reported work has taken place in the PiStar AI and Optimization Lab at Texas A&M University. PiStar is supported in part by NSF (IIS-2238979).

REFERENCES

[1] Joshua Achiam. 2018. Spinning Up in Deep Reinforcement Learning. (2018).

[2] Neziha Akalin and Amy Loutfi. 2021. Reinforcement learning approaches in social robotics. *Sensors* 21, 4 (2021), 1292.

[3] Robin Allesiardo, Raphaël Féraud, and Odalric-Ambrym Maillard. 2017. The non-stationary stochastic multi-armed bandit problem. *International Journal of Data Science and Analytics* 3 (2017), 267–283.

[4] Susan Amin, Maziar Gomrokchi, Harsh Satija, Herke van Hoof, and Doina Precup. 2021. A survey of exploration methods in reinforcement learning. *arXiv preprint arXiv:2109.00157* (2021).

[5] Mikkel ArnØ, John-Morten Godhavn, and Ole Morten Aamo. 2020. Deep reinforcement learning applied to managed pressure drilling. In *SPE Norway Subsurface Conference*. OnePetro.

[6] Peter Auer. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3, Nov (2002), 397–422.

[7] Peter Auer and Ronald Ortner. 2006. Logarithmic online regret bounds for undiscounted reinforcement learning. *Advances in neural information processing systems* 19 (2006).

[8] Peter Auer and Ronald Ortner. 2010. UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica* 61, 1-2 (2010), 55–65.

[9] James Ault, Josiah P. Hanna, and Guni Sharon. 2020. Learning an Interpretable Traffic Signal Control Policy. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS '20)*. 88–96.

[10] James Ault and Guni Sharon. 2021. Reinforcement Learning Benchmarks for Traffic Signal Control. In *Proceedings of the Thirty-fifth Conference on Neural Information Processing Systems (NeurIPS 2021) Datasets and Benchmarks Track*.

[11] Andrew Gehret Barto, Steven J Bradtke, and Satinder P Singh. 1991. *Real-time learning and control using asynchronous dynamic programming*. University of Massachusetts at Amherst, Department of Computer and Information Science.

[12] Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. 2016. Unifying count-based exploration and intrinsic motivation. *Advances in neural information processing systems* 29 (2016).

[13] Dimitri Bertsekas and John N Tsitsiklis. 1996. *Neuro-dynamic programming*. Athena Scientific.

[14] Omar Besbes, Yonatan Gur, and Assaf Zeevi. 2014. Stochastic multi-armed-bandit problem with non-stationary rewards. *Advances in neural information processing systems* 27 (2014).

[15] Ronen I Brafman and Moshe Tennenholtz. 2002. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research* 3, Oct (2002), 213–231.

[16] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. OpenAI Gym. *arXiv preprint arXiv:1606.01540* (2016).

[17] Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. 2018. Large-Scale Study of Curiosity-Driven Learning. In *International Conference on Learning Representations*.

[18] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. 2018. Exploration by random network distillation. In *International Conference on Learning Representations*.

[19] Tapas Kumar Chandra. 2012. *The Borel-Cantelli Lemma*. Springer Science & Business Media.

[20] Nuttapong Chentanez, Andrew Barto, and Satinder Singh. 2004. Intrinsically motivated reinforcement learning. *Advances in neural information processing systems* 17 (2004).

[21] Maxime Chevalier-Boisvert, Lucas Willems, and Suman Pal. 2018. *Minimalistic Gridworld Environment for Gymnasium*. https://github.com/Farama-Foundation/Minigrid

[22] Kai Lai Chung and Paul Erdos. 1952. On the application of the Borel-Cantelli lemma. *Trans. Amer. Math. Soc.* 72, 1 (1952), 179–186.

[23] Balázs Csanád Csáji and László Monostori. 2008. Value function based reinforcement learning in changing Markovian environments. *Journal of Machine Learning Research* 9, 8 (2008).

[24] Chris Dann, Yishay Mansour, Mehryar Mohri, Ayush Sekhari, and Karthik Sridharan. 2022. Guarantees for epsilon-greedy reinforcement learning with function approximation. In *International conference on machine learning*. PMLR, 4666–4689.

[25] Sheelabhadra Dey, Sumedh Pendurkar, Guni Sharon, and Josiah P Hanna. 2021. A Joint Imitation-Reinforcement Learning Framework for Reduced Baseline Regret. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 3485–3491.

[26] Eyal Even-Dar and Yishay Mansour. 2001. Convergence of optimistic and incremental Q-learning. *Advances in neural information processing systems* 14 (2001).

[27] Mehdi Fatemi, Shikhar Sharma, Harm Van Seijen, and Samira Ebrahimi Kahou. 2019. Dead-ends and secure exploration in reinforcement learning. In *International Conference on Machine Learning*. PMLR, 1873–1881.

[28] Justin Fu, John Co-Reyes, and Sergey Levine. 2017. Ex2: Exploration with exemplar models for deep reinforcement learning. *Advances in neural information processing systems* 30 (2017).

[29] Florian Fuchs, Yunlong Song, Elia Kaufmann, Davide Scaramuzza, and Peter Dürr. 2021. Super-human performance in gran turismo sport using deep reinforcement learning. *IEEE Robotics and Automation Letters* 6, 3 (2021), 4257–4264.

[30] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. 2018. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905* (2018).

[31] Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. 2016. Vime: Variational information maximizing exploration. *Advances in neural information processing systems* 29 (2016).

[32] Tommi Jaakkola, Michael Jordan, and Satinder Singh. 1993. Convergence of stochastic iterative dynamic programming algorithms. *Advances in neural information processing systems* 6 (1993).

[33] Leslie Pack Kaelbling. 1993. *Learning in embedded systems*. MIT press.

[34] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. 1996. Reinforcement learning: A survey. *Journal of artificial intelligence research* 4 (1996), 237–285.

[35] Vivek Kesireddy, Georgy Kompantsev, Sheelabhadra Dey, Eduardo Gildin, Enrique Z. Losoya, and Narendra Vishnumolakala. 2023. Maximizing Efficiency of Deep-Reinforcement Learning Agents in Autonomous Directional Drilling with Hyperparameter Optimization *(SPE/AAPG/SEG Unconventional Resources Technology Conference, Vol. Day 3 Thu, June 15, 2023)*. D031S063R004.

[36] B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A Al Sallab, Senthil Yogamani, and Patrick Pérez. 2021. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems* 23, 6 (2021), 4909–4926.

[37] Jens Kober, J Andrew Bagnell, and Jan Peters. 2013. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research* 32, 11 (2013), 1238–1274.

[38] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971* (2015).

[39] Long-Ji Lin. 1992. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine learning* 8, 3 (1992), 293–321.

[40] Sam Lobel, Omer Gottesman, Cameron Allen, Akhil Bagaria, and George Konidaris. 2022. Optimistic Initialization for Exploration in Continuous Control. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 7612–7619.

[41] Manuel Lopes, Tobias Lang, Marc Toussaint, and Pierre-Yves Oudeyer. 2012. Exploration in model-based reinforcement learning by empirically estimating learning progress. *Advances in neural information processing systems* 25 (2012).

[42] Marlos C Machado, Sriram Srinivasan, and Michael Bowling. 2015. Domain-independent optimistic initialization for reinforcement learning. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[43] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*. PMLR, 1928–1937.

[44] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *nature* 518, 7540 (2015), 529–533.

[45] Shakir Mohamed and Danilo Jimenez Rezende. 2015. Variational information maximisation for intrinsically motivated reinforcement learning. *Advances in neural information processing systems* 28 (2015).

[46] Sindhu Padakandla. 2021. A survey of reinforcement learning algorithms for dynamically varying environments. *ACM Computing Surveys (CSUR)* 54, 6 (2021), 1–25.

[47] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. 2017. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*. PMLR, 2778–2787.

[48] Pascal Poupart, Nikos Vlassis, Jesse Hoey, and Kevin Regan. 2006. An analytic solution to discrete Bayesian reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*. 697–704.

[49] Martin L Puterman. 1990. Markov decision processes. *Handbooks in operations research and management science* 2 (1990), 331–434.

[50] Tabish Rashid, Bei Peng, Wendelin Boehmer, and Shimon Whiteson. 2020. Optimistic exploration even with a pessimistic initialisation. *arXiv preprint arXiv:2002.12174* (2020).

[51] Dmitry B Rokhlin. 2019. Robbins–Monro conditions for persistent exploration learning strategies. In *Modern Methods in Operator Theory and Harmonic Analysis: OTHA 2018, Rostov-on-Don, Russia, April 22-27, Selected, Revised and Extended Contributions 8*. Springer, 237–247.

[52] Jürgen Schmidhuber. 1991. A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*. 222–227.

[53] Jürgen Schmidhuber. 2010. Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *IEEE transactions on autonomous mental development* 2, 3 (2010), 230–247.

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

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

[56] Satinder Singh, Tommi Jaakkola, Michael L Littman, and Csaba Szepesvári. 2000. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine learning* 38 (2000), 287–308.

[57] Bradly C Stadie, Sergey Levine, and Pieter Abbeel. 2015. Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814* (2015).

[58] Hao Sun, Lei Han, Rui Yang, Xiaoteng Ma, Jian Guo, and Bolei Zhou. 2022. Exploit Reward Shifting in Value-Based Deep-RL: Optimistic Curiosity-Based Exploration and Conservative Exploitation via Linear Reward Shaping. In *Advances in Neural Information Processing Systems*, Vol. 35. 37719–37734.

[59] Richard S Sutton. 1990. Integrated modeling and control based on reinforcement learning and dynamic programming. *Advances in neural information processing systems* 3 (1990).

[60] Richard S Sutton. 1995. Generalization in reinforcement learning: Successful examples using sparse coarse coding. *Advances in neural information processing systems* 8 (1995).

[61] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[62] István Szita and András Lőrincz. 2008. The many faces of optimism: a unifying approach. In *Proceedings of the 25th international conference on Machine learning*. 1048–1055.

[63] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. 2017. # exploration: A study of count-based exploration for deep reinforcement learning. *Advances in neural information processing systems* 30 (2017).

[64] Sebastian B Thrun and Knut Möller. 1991. Active exploration in dynamic environments. *Advances in neural information processing systems* 4 (1991).

[65] Michel Tokic. 2010. Adaptive ε-greedy exploration in reinforcement learning based on value differences. In *Annual Conference on Artificial Intelligence*. Springer, 203–210.

[66] Michel Tokic and Günther Palm. 2011. Value-difference based exploration: adaptive control between epsilon-greedy and softmax. In *Annual conference on artificial intelligence*. Springer, 335–346.

[67] John N Tsitsiklis. 1994. Asynchronous stochastic approximation and Q-learning. *Machine learning* 16, 3 (1994), 185–202.

[68] Narendra Vishnumolakala, Vivek Kesireddy, Sheelabhadra Dey, Eduardo Gildin, and Enrique Z. Losoya. 2023. Optimizing Well Trajectory Navigation and Advanced Geo-Steering Using Deep-Reinforcement Learning *(SPE Annual Technical Conference and Exhibition, Vol. Day 2 Tue, October 17, 2023)*. D021S012R003.

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

[70] Christopher John Cornish Hellaby Watkins. 1989. Learning from delayed rewards. (1989).

[71] Marco A Wiering. 1999. *Explorations in efficient reinforcement learning*. Ph.D. Dissertation. University of Amsterdam.

[72] Kok-Lim Alvin Yau, Junaid Qadir, Hooi Ling Khoo, Mee Hong Ling, and Peter Komisarczuk. 2017. A survey on reinforcement learning models and algorithms for traffic signal control. *ACM Computing Surveys (CSUR)* 50, 3 (2017), 1–38.