Curious Explorer: A Provable Exploration Strategy in Policy Learning

Marco Miani, Maurizio Parton, and Marco Romito

Abstract—A coverage assumption is critical with policy gradient methods, because while the objective function is insensitive to updates in unlikely states, the agent may need improvements in those states to reach a nearly optimal payoff. However, this assumption can be unfeasible in certain environments, for instance in online learning, or when restarts are possible only from a fixed initial state. In these cases, classical policy gradient algorithms like REINFORCE can have poor convergence properties and sample efficiency. Curious Explorer is an iterative state space pure exploration strategy improving coverage of any restart distribution \( \rho \). Using \( \rho \) and intrinsic rewards, Curious Explorer produces a sequence of policies, each one more exploratory than the previous one, and outputs a restart distribution with coverage based on the state visitation distribution of the exploratory policies. This paper main results are a theoretical upper bound on how often an optimal policy visits poorly visited states, and a bound on the error of the return obtained by REINFORCE without any coverage assumption. Finally, we conduct ablation studies with REINFORCE and TRPO in two hard-exploration tasks, to support the claim that Curious Explorer can improve the performance of very different policy gradient algorithms.

Index Terms—Exploration, PAC algorithms, reinforcement learning.

I. INTRODUCTION

The term Policy Gradient (PG) includes a family of RL methods that parameterize the policy with a \( C^\infty \) parameterization and use an estimate of the gradient to maximize the expected long-term reward of the MDP. PG methods belong to a larger family of policy optimization methods which learn directly the optimal policy, while the so-called value-based methods learn a value function first.

There are several reasons to prefer PG methods to value-based methods [1, Section 13.1], and this has led to a huge success of PG methods in deep RL. Such success has been largely driven by experiments, without theoretical guarantees. The main issue is that the objective function is non-convex, and so the only guarantees of convergence are to local optima. Moreover, these guarantees come from stochastic approximation theory, and require the strong assumption of an oracle providing the exact gradient, or at least the possibility of obtaining unbiased estimates of the gradient [2].

This is now rapidly changing. Indeed, the last few years have seen a proliferation of theoretical results answering fundamental questions for several important PG algorithms, using one of the several theoretical performance measures [3] for RL. [4] describe MDP structural properties for convergence to the global optimum. This is true in particular for finite MDP with a complete parameterization. [5] prove that a variant of PPO [6] equipped with over parameterized neural networks converges to the global optimum at a rate sub linear in the number of iterations, and [7] prove a similar result for actor-critic schemes [8] based on natural PG [9]. Provable characterizations of computational, approximation, and sample size issues are given in a comprehensive paper by [10], [11]. Provable regret bounds are found in [12], [13] and [14].

All the papers listed above either assume coverage or ergodicity. These two hypotheses are as strong as widely accepted in the literature. Coverage assumes that trajectories can be restarted from states sampled by a distribution everywhere positive, also called a soft reset model. Ergodicity assumes that for any stationary policy the induced Markov chain is irreducible and aperiodic. For a more thorough background of coverage and ergodicity, see [15] and [16, Section 2]. These assumptions completely eliminate the classic RL exploration-exploitation balancing problem. For instance, in [17] it is shown that in the task known as tabletop organization starting from the ideal stationary distribution of the optimal policy improves over the standard initial state distribution the learning speed, and also the performance by nearly 18%. Yet, those hypothesis are rarely satisfied in real problems. The primary contribution of this paper is Curious Explorer, a “helper” algorithm that provides an approximation of coverage for any (tabular) environment.
II. ORIGINAL CONTRIBUTION

From a broad perspective, the novel contributions of this paper are an original pure exploration strategy that we call Curious Explorer, a corresponding theoretical analysis framework that does not assume coverage or ergodicity, and a clear separation between exploration and optimization.

The exploration strategy is outlined in Sections III and VI. It is a theoretical framework that does not require coverage or ergodicity. Curious Explorer (hereafter, CE) is Algorithm 2. CE is an iterative procedure constructing at each step a pure-exploration policy. Starting from the uniform policy, CE optimizes at each step an intrinsic reward for reaching poorly visited states (defined by (5)), using any optimizer opt. The state visitation distribution of this optimized policy is then used as a reset model at the next step, so that the new cycle of optimization by opt produces a new optimized policy, and thus a new state visitation distribution with improved exploration. After a certain number of exploration steps, where the original extrinsic reward is never used, CE returns those optimized policies. Theorem 1 is the main theoretical contribution of this paper: if opt is PAC, then the visitation probability of poorly visited states under any policy goes to 0 for a large enough number of exploration steps in the CE procedure.

Our framework provides a conceptual and practical separation between exploration and optimization. Conceptual, because any optimizer opt can be used for the exploration steps in Algorithm 2. Practical, because the implementation is modular; as long as opt takes an OpenAI Gym environment and a number of episodes as input, and returns a policy, then opt can be used without modifications in CE. See Sections “Solver REINFORCE” and “Solver TRPO” in the source code for details. Such a clear separation between exploration and optimization is not common, but is something that, in our opinion, should be pursued when researching a novel exploration framework.

III. METHODS AND CLAIMS

CE is an iterative procedure that samples from a (finite, tabular) MDP with any restart distribution ρ (not necessarily soft), takes any optimization algorithm opt as input, and outputs a list of policies whose state visitation distributions can then be used as a soft reset model µ. At every step of the iteration, µ improves exploration on previously poorly visited states (defined by (5)).

The idea behind the theoretical analysis in Section VI is simple. The state space S is split into poorly visited states K and its complement S − K. By simulating coverage with good bounds on S − K, we can use any provable opt to obtain theoretical guarantees on S − K. Where we do not have good bounds, that is, on poorly visited states K, we can guarantee that those states cannot be visited more than a certain frequency by any policy, and therefore not even by the optimal policy. See Theorem 1, main theoretical result of this paper.

The most important claim in this paper is that our exploration framework is useful for theoretical research. In fact, while CE does not require coverage of the restart distribution ρ or ergodicity, it can be used with an optimizer opt that satisfies theoretical properties under a coverage assumption on ρ. Since CE simulates coverage, one can use opt with the reset model µ given by the last policy returned by CE. This combination of CE and opt, denoted by opt+CE, is a new instance of opt that maintains some of the original theoretical properties of opt (because of the theoretical bounds given by Theorem 1) without a coverage assumption on ρ (because µ is soft). In a broad sense, CE can be used to “remove the coverage hypothesis” from existing algorithms. In support of this claim, we take the recent work [18], where REINFORCE with coverage is shown to be PAC. We remove the coverage assumption, and use Theorem 1 to obtain an estimate of the state visitation distribution on the set of poorly visited states (Corollary 5) and an estimate of the return obtained by CE+REINFORCE (Corollary 6). Note that a good bound on the mismatch coefficient in Corollary 6 would still allow to obtain a PAC instance of REINFORCE without any coverage assumption. This is the main future development of this work, see Section VIII.

Next claim is that CE can improve exploration on hard-exploration tasks, independent of the theoretical properties of the optimizer opt. This is supported by ablation studies in Section VII and in Supplementary Material.

For the ablation studies, we selected REINFORCE and TRPO as optimizers opt. This choice is due to the theoretical nature of our paper, because REINFORCE and TRPO are both strongly theoretically supported. Moreover, in the family of methods not focusing on exploration, they are at opposite ends of the spectrum: REINFORCE enjoys theoretical guarantees but is not sample efficient, while TRPO is one of the best empirical optimizers. As a side note, the fact that CE works without modifications with two very different opt supports the claim that the pursued separation between exploration and optimization is useful.

Since the theoretical guarantees of CE depend on a tabular description of the MDP, we did not use classical hard-exploration MDPs like Atari games. We instead selected two commonly used hard-exploration tabular MDPs: the Consecutive Crossroad Traps [10] and the Diabolical Combination Lock [19], with a depth of 5 (easy), 10 (intermediate), 20 (challenging). At depth 20 on the Combination Lock, the addition of CE exploration to TRPO shows a major improvement on the learning performance, see Fig. 2. Please note that when opt is not PAC, as for TRPO, theoretical guarantees of Theorem 1 do not hold anymore, but CE still improves exploration.

A final note. We do not make the claim that CE is a better than state-of-the-art exploration strategies like for instance Go-Explore in [20] or HOMER in [21]. This would not make sense, because CE exploration efficiency depends on the choice of the optimizer opt used to build the exploration policies. In particular, REINFORCE and TRPO should not be mistaken for “exploration baselines” in the experimental evaluation. Clearly, the supported claim that CE improves the exploration of different optimizers suggests that opt+CE could explore better than opt also when opt is a state-of-the-art exploration strategy. However, this claim would require a completely different experimental evaluation. This is not done here and is left as a future development, see Section VIII.
IV. RELATED WORK

Since CE enjoys theoretical guarantees, and we do not claim that it outperforms other exploration strategies, we compare CE only with theoretical papers. Thus, we are intentionally leaving out recent state-of-the-art empirical exploration strategies like in [20], [22], [23], [24], to name a few. A very nice, comprehensive and unified empirical comparison of different exploration methods in RL can be found at [25].

The idea of promoting exploration by rewarding poorly visited states is not new. In the literature these methods appear with various umbrella terms, including intrinsic reward, curiosity-driven exploration, optimism in the face of uncertainty. See [26], [27], [28] for an interesting approach to this concept, and the very recent [29]. While the extrinsic reward is returned by the environment, an intrinsic reward is artificially assigned to poorly visited states. In CE the intrinsic reward is the main driver of exploration.

Model-Based Interval Estimation (MBIE) with Exploratory Bonus (MBIE-EB) [30] uses an intrinsic reward based on the number of visits: $1/n(s,a)$. While MBIE-EB is model based, and in this differs from CE, its count-based intrinsic reward is related to our notion of poorly visited states as states with small state visitation probability, see Definition (5).

Bayesian Exploration Bonus (BEB) [31] uses a Bayesian approach, with count-based intrinsic reward $1/(1+n(s,a))$. Interestingly, in this paper it is shown that if the intrinsic reward decays faster than $1/n(s,a)$, then the method is not PAC. So in particular they show that neither BEB nor true Bayesian methods can be PAC. CE is not affected by this result, because its intrinsic reward is constant.

More recently, [32] and [33] extended count-based methods to function approximation. Extending CE exploration strategy to the function approximation setting is one of the possible future developments, see Section VIII.

Explicit Explore or Exploit (E3) [16] is the first provably near-optimal polynomial-time algorithm for sample complexity that does not assume coverage. It is also the first to introduce the distinction between known and unknown states, which allows using distinct periods of pure exploration and pure exploitation and to quantify at each step the relationship between exploration and exploitation. In CE, the notion of poorly visited states (5) can be seen as a derivation of this idea, as well as the clear separation between exploration and optimization.

R-MAX [34] is very close to CE. It is a generalization and simplification of E3, with several very important achievements. For instance, in R-MAX the bias introduced by optimism under uncertainty is theoretically justified for the first time. In short, R-MAX changes the reward of poorly visited states to the best possible reward (hence the name), preserving the reward for the other states. Similarly, CE assigns the best possible reward to poorly visited states, but changes to 0 every other reward, making the exploration more effective. Another difference is that the threshold for poorly visited states is constant for R-MAX, while for CE follows a specific schedule, that can be used to tune the depth of the exploration. Moreover, R-MAX has a polynomial bound in states, actions, and mixing time $T$ (for every starting state, after $T = T(\epsilon)$ steps the policy is $\epsilon$-close to the optimal one), while CE has a bound that depends only on states, actions and $\gamma$. R-MAX is more general than CE, because it covers zero-sum stochastic games, not only MDPs. Finally, as with E3, R-MAX does not require coverage.

E3 and its evolution R-MAX provide a theoretical framework that does not require coverage, similarly to CE. This seems to undermine our originality claim in Section II. However, E3 and R-MAX replace coverage with the very strong assumption of ergodicity. Ergodicity implies that there is an $\epsilon$-mixing time $T$, thus the starting state is irrelevant because after $T$ steps the policy will become nearly-optimal. This mixing time is an unknown and potentially huge parameter, so any theoretical bound including $T$ is weaker than it seems. R-MAX contains $T$ in the final bound. CE does not require ergodicity.

More recent theoretical exploration strategies similar to CE are DMQ in [35] and HOMER in [21]. While the theoretical guarantees of DMQ need strong structure hypotheses that are not satisfied, for instance, by the Diabolical Combination Lock, HOMER is instead similar from several aspects to CE.

HOMER and CE share the idea of dividing the algorithm into steps, each producing a wider coverage of states exploiting the coverage of the previous steps. The main difference is that, at each of these steps, HOMER finds a cover policy for each state (for each kinematic equivalence class, but $N_{KD} = |S|$ when the MDP is in canonical form) optimizing with intrinsic reward of 1 on that sole state, while CE finds a single cover policy optimizing with intrinsic reward of 1 on the whole set of poorly visited states. Another key difference is that HOMER needs exactly $H$ (the MDP time horizon, potentially huge) steps for its theoretical bound, while CE bounds are valid after any number of steps. HOMER and CE can be compared with respect to several other aspects, and this is done in Table 1 in Supplementary Material.

V. NOTATION

We work with a finite and tabular Markov Decision Process (MDP) $M := (S, \rho, \pi, \mathcal{A}, r, \mathcal{P})$, where $S$ is the state space, $\rho$ is the restart distribution, $\mathcal{A}$ is the action space, $r : S \times \mathcal{A} \to [0, 1]$ is the reward function and $\mathcal{P} : S \times \mathcal{A} \to \Delta(S)$ is the transition model. The set of stationary policies $\pi$ is denoted by $\Pi$. The setting is discounted episodic, and $\gamma \in [0, 1]$ denotes the discount factor. A $\mu$-reset model, for a distribution $\mu$ on $S$, allows episodes to begin with states sampled from $\mu$, rather than $\rho$. Although $\mu$ and $\rho$ both determine initial states, they have different roles in this paper, because the goal of CE is to build an empirical distribution with full coverage starting from $\rho$. In fact, later we will assume $\rho$ to be the worst possible distribution in terms of coverage, that is, concentrated on a single starting state $s_0$.

The value of the MDP $V_{\pi, r, \mathcal{P}, \gamma}(\mu)$ is a normalized average over trajectories starting from $\mu$ and following $\pi$, and by the state visitation distribution $d_{\mu, P, \gamma}(s)$ we mean the normalized, discounted, future-state visitation distribution obtained by starting from $\mu$ and following $\pi$, as defined in [36, Formula 2.1]:

$$
d_{\mu, P, \gamma}(s) := (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \Pr(S_t = s | \pi, \mu)
$$
Algorithm 1: visit(π, μ). Given π, a MDP (S, s0, A, r, P), and a μ-Reset Model, Returns a State $s \sim d^π_{\mu}$.d^\pi_{\mu}(\cdot)$.

1: Empirically sample $s$ from $μ$.
2: while true do
3: With probability $1 - γ$:
4: break
5: Choice action $a$ according to $π(\cdot|s)$.
6: Perform $a$ in $M$ and go to state $s'$ according to the unknown $P(\cdot|s, a)$.
7: Set $s = s'$.
8: end while
9: return $s$

Note that the normalization factor $1 - γ$ is needed because otherwise $d^π_{\mu}$ would not be a probability distribution. See [36] for more details.

For every $μ$, a distributional characterization of the value of the MDP holds:

$$V_{π, r, P, γ}(μ) = \sum_{s \in S, a \in A} d^π_{μ}(s)π(a|s)r(s, a)$$  \hspace{1cm} (1)

In particular, when $μ$ is the true restart distribution $ρ$, (1) gives the true (normalized) value of the MDP. We denote by $π^*$ any optimal policy, that is, any policy maximizing $V_{π, r, P, γ}(ρ)$. Note that in the tabular setting there always exists at least one $π^*$ that is optimal irrespective of the initial distribution. Hereafter, we assume without loss of generality that $μ$ is concentrated on a fixed initial state $s_0$. Note that this is the worst possible assumption in terms of coverage.

The sample complexity $N(ε, δ)$ of a RL algorithm $A$ is the minimum number of samples $n \in \mathbb{N}$ such that

$$P(\text{error}(A \text{ with } n \text{ samples}) ≥ ε) < δ$$  \hspace{1cm} (2)

where error is measured as the expectation of some appropriate loss function, and the samples can be either the number of episodes or the number of time steps. An optimization algorithm is Probably Approximately Correct or PAC if for any $ε > 0$ and $δ ∈ (0, 1)$ its sample complexity $N(ε, δ)$ is finite. Given that exponential bounds are trivial to achieve, significant theoretical results are those that guarantee polynomial bounds. See for instance [37].

VI. THEORETICAL ANALYSIS

Full proofs of results in this section can be found in the Appendix in Supplementary Material.

A. Simulating the Visitation Distribution

Aim of Algorithm 1 is to empirically estimate the visitation distribution $d^π_{μ}(\cdot)$. This is done by taking as input a policy $π$, a MDP, and a $μ$-reset model, and then using $π$ until a break occurs. Since a break occurs at every step with probability $1 - γ$, at the end of Algorithm 1 we are in a state empirically sampled from $d^π_{μ}(\cdot)$. We denote this empirical distribution with visit($π, μ$), to distinguish it from the theoretical distribution $d^π_{μ}(\cdot)$. Our exploration strategy leverages Algorithm 1 as a spread function over $μ$, in a convolutional smoothing fashion. In fact, visit($π, μ$) returns a state empirically sampled from $\tilde{μ}$, where $\tilde{μ}$ is given by:

$$\tilde{μ}(s') = μ \ast d^π_{μ}(s') = \sum_{s \in S} μ(s)d^π_{μ}(s) = d^π_{μ}(\cdot)$$  \hspace{1cm} (3)

That is, moving from $μ$ to $\tilde{μ}$, the reset probability of each state $s$ is “spread” across all other states according to the state visitation distribution $d^π_{μ}(\cdot)$. Iterating this argument, concentrated distributions are diffused to softer models.

Intuition is helped through an analogy with image blurring. Represent a reset model with an image where each pixel is a state and the color depends on the reset probability of that state, ranging continuously from black if 0 to white if 1. Thus, a $s_0$-reset model corresponds to an image that is all black but for one white pixel. The visit algorithm spreads the $s_0$ pixel light across other pixels according to the “point spread function” $d^π_{μ}(\cdot)$. If more than one initial pixel is not black, the corresponding spread lights will merge linearly, each weighted by the respective light intensity, in the same way as visitation distributions in (3). As every analogy, similarities are limited. In image blurring the spread function is the same for each pixel, thus blurring a one-white-pixel image towards a uniform gray image. In our case the dynamical reset blurs the model according to the state visitation distribution.

Note that a theoretical analysis on how the initial state distribution affects the performance can be found in [36]. The main idea is that learning an optimal policy often requires policy improvement at states that are unlikely to be visited. Spreading the starting state distribution, as CE does, can encourage policy improvement at those unlikely states, see [36, Corollary 4.5].

B. The Exploration Phase

We now describe Algorithm 2, that is, CE. At each step $n = 0, 1, \ldots$ the policy $π_{n+1}$ is the result of a curiosity-driven optimization strategy:

$$μ_{-1} = s_0 \xrightarrow{π_0} μ_{-1} \ast d^{π_{-1}; P, γ} =: μ_{0} \xrightarrow{π_1} \ldots \xrightarrow{π_{n-1}} μ_{n-2} \ast d^{π_{n-1}; P, γ} =: μ_{n-1} \xrightarrow{π_n} \ldots$$  \hspace{1cm} (4)

The above strategy (4) is initialized with a uniform policy $π_0$. Then, a subprocess is started to compute an “optimally exploring” policy $π_{n+1}$. This subprocess is indeed a standard value optimization run on a virtual MDP $M_n$ and can thus be made through any RL optimization method $\text{opt}$. During the subprocess, each trajectory restarts according to a mixture of $s_0$ (or $ρ$) and the reset model $μ_n$ simulated by Algorithm 1. We explicitly state the dependence of the optimization method from the reset model, namely, $π_{n+1} = \text{opt}(M_n, μ_n)$, where $M_n = (S, s_0, A_n, r_n, P)$ is a MDP which has the same states, starting distribution, actions and transitions as the original MDP $M$, but different reward function. The reward function $r_n$ is
Curious Explorer. Given a MDP $(S, s_0, A, r, P)$, a RL Optimization Algorithm $\text{opt}(\cdot, \cdot)$, and $N \in \mathbb{N}$, Returns $N$ Exploratory Policies.

1: Schedule the visit thresholds $\{\beta_n\}_{n \in \mathbb{N}}$.
2: If $\text{opt}$ is PAC, schedule its error bounds $\{\epsilon_n\}_{n \in \mathbb{N}}$, and its confidence levels $\{\delta_n\}_{n \in \mathbb{N}}$.
3: Set $\mu_{-1} := \mathbb{I}_{\{s = s_0\}}$.
4: Set $\pi_0 := \pi_U$, the uniform policy.
5: for $n = 0, 1, 2, \ldots, N$ do
6: Empirically estimate $D_n = d_{\mu_{n-1}}^s \mathbb{P}, \gamma$.
7: Compute the set $\mathcal{K}_n$ of poorly visited states:
   \[ \mathcal{K}_n = \{ s \in S | \sum_{i=0}^n D_i(s) \leq \beta_n \} \]
8: Change the reward of the MDP to an intrinsic reward on poorly visited states: $r_n(s, a) = 1_{\{s \in \mathcal{K}_n\}}$.
9: Set $\mu_n(s) = \frac{1}{2} D_n(s) + \frac{1}{2} \mathbb{I}_{\{s = s_0\}}$.
10: Use $\text{opt}$ on $\mathcal{M}_n := (S, s_0, A, r_n, P)$ to approximate the optimal policy $\pi_{n+1} = (\epsilon_n, \delta_n) \circ \text{opt}(\mathcal{M}_n, \mu_n)$.
11: end for
12: Return $\{\pi_n\}_{n=1}^{N+1}$.

an intrinsic reward motivating the optimizer in exploring less explored states. To measure how many states are visited, we recursively define the set of poorly visited states at step $n$ as
\[ \mathcal{K}_n = \left\{ s \in S \left| \sum_{i=0}^n d_{\mu_{i-1}}^s \mathbb{P}, \gamma (s) \leq \beta_n \right\} \right. \]

where a sequence of $\beta_n \geq 0$ is suitably chosen as a hyperparameter of CE. The intrinsic reward is 1 on poorly visited states, 0 otherwise: $r_n(s, a) := 1_{\{s \in \mathcal{K}_n\}}$, where $1_{\{\cdot\}}$ denotes the indicator function on sets. In the extreme case $\beta_n = 0$, states will be never poorly visited, because any (reachable) state has a non-zero probability of being visited under the uniform policy. At the other end there is $\beta_n$ going to infinite faster than $n$, and in this case every state will eventually be poorly visited forever.

Note that the original reward is completely forgotten, in contrast with R-MAX. In other words, the process on each trajectory is “lured” to visit those states whose overall visitation distribution in previous steps falls below a threshold. Each reward function $r_n$ is action-independent, therefore for every policy $\pi$ we get
\[ V_{\pi, r_n, \mathbb{P}, \gamma}(\mu_{n-1}) = \sum_{s \in S, a \in A} d_{\mu_{n-1}}^s \mathbb{P}, \gamma (s) \pi(a | s) r_n(s, a) \]
\[ = \sum_{s \in S} d_{\mu_{n-1}}^s \mathbb{P}, \gamma (s) \mathbb{I}_{\{s \in \mathcal{K}_n\}} = \sum_{s \in \mathcal{K}_n} d_{\mu_{n-1}}^s \mathbb{P}, \gamma (s) \]

where we have used the distributional characterization (1) of the value function. Thus, maximizing the value function in $\mathcal{M}_n = (S, s_0, A, r_n, P)$ with respect to the $\mu_{n-1}$-reset model corresponds to maximizing the sum of visitation distributions on poorly visited states.

A few remarks are in order. The first is that at each step the new reset model is defined as a balance between the visitation distribution collected with the current policy and the original starting distribution $\rho(= s_0)$. This has a twofold purpose. On the one hand the true restart distribution $\rho$ is meaningful for the problem. On the other hand it allows for a uniform control of the number of poorly visited states, as well as of the visitation distribution on $\mathcal{K}_n$. The second remark is over the choice of the sequence of thresholds $\beta_n$ that identify poorly visited states. The choice of a constant sequence $\beta_n$ makes poorly visited states in CE the same as unknown states in R-MAX. For the linear choice $\beta_n = \beta \cdot (n + 1)$, for some number $\beta > 0$, the sum in (5) can be rewritten as an average of the visitation distributions. Thus, each non-poorly visited state on average has accumulated a proportion of visits which is larger than $\beta$. In particular, on $S - \mathcal{K}_n$ the CE output $\mu_n$ of the exploration phase is bounded from below, and this allows using theoretical results requiring a coverage hypothesis. In particular, if the subprocess $\text{opt}$ is PAC under a coverage hypothesis, so that theoretical upper bounds $\epsilon_n$ on the error of the returned policy hold, we can use them to provide theoretical guarantees on the exploration.

This discussion leads us to the main theorem of this paper. Assume to follow Algorithm 2 for $N$ steps and let $Q_N := \bigcap_{n=N/2}^N \mathcal{K}_n$ be the states “more poorly visited than others”. Let $\bar{n}(s) := \max\{n \leq N | s \in \mathcal{K}_n\}$ be the last step when $s$ is poorly visited, and recall that $\Pi$ is the set of all stationary policies.

Theorem 1: Let an MDP $\mathcal{M} = (S, s_0, A, r, P)$, an $s_0$-reset model, a discount factor $\gamma$ and a PAC $\text{opt}$ be given. Assume that at step $n$ the policy returned by $\text{opt}$ is at least $\epsilon_n$, optimal with probability at least $1 - \delta_n$. Follow Algorithm 2 with any non-decreasing sequence of thresholds $\{\beta_n\}_{n \leq N}$. Then for every $N$ with probability at least $1 - \sum_{n=0}^N \delta_n$ we have
\[ \mu_N(s) \geq \frac{1}{2(N+1)} \beta_{N/2} \forall s \notin Q_N \]
and
\[ \sum_{s \in Q_N} d_{s_0}^s \mathbb{P}, \gamma (s) \leq \max_{\pi \in \Pi} \sum_{s \in Q_N} d_{s_0}^s \mathbb{P}, \gamma (s) \]
\[ \leq \frac{4}{N} \left( \sum_{n=0}^N \epsilon_n + \sum_{s \in S} \beta_{n}(s) \right) \max_{\pi \in \Pi} d_{s_0}^s \mathbb{P}, \gamma (s) \]

Remark 2: Theorem 1 should not lead one to think that CE can only be used when $\text{opt}$ is PAC. In fact, from a very general perspective, the main claim of our paper is the following: whenever CE is used with an optimizer $\text{opt}$ that satisfies certain theoretical properties under a coverage assumption, then the combination $\text{opt} + CE$ of CE and $\text{opt}$ is a new optimizer that maintains some of the original theoretical properties of $\text{opt}$, without requiring coverage. This happens because CE provides $\text{opt}$ with a “simulated coverage”. This is a very general statement that needs to be supported with specific theoretical properties. We support this general statement by choosing PAC as an example of theoretical property in Theorem 1, and subsequent corollaries. Yet, this is just an example to support the statement, and the PAC hypothesis is not required elsewhere in the paper. To support the claim that CE is useful also when $\text{opt}$ is not PAC, we provide in Section VII-B examples of hard exploration tasks where TRPO+CE performs much better than TRPO alone.

Remark 3: The bound in Theorem 1 is a general result on the estimate of the visitation distribution provided by $\text{opt} + CE,$
when $\text{opt}$ is PAC. This estimate is described in terms of the number $N + 1$ of CE iterations, the PAC error bounds $\epsilon_n$, $n = 0, \ldots, N$, chosen for $\text{opt}$, the CE thresholds $\beta_n(s)$ at the last step when $s$ was still poorly visited, and the exploitative factor $\sum_{s \in S} \max_{\pi \in \Pi} d^*_{\pi_0, \gamma}(s) = \epsilon$. This last term is the more interesting: it is always $\leq |S|$, is fixed, and is indeed a characteristic parameter for every MDP. The quantity $\epsilon$ measures, in terms of number of samples, the overall cost of gathering information about the structure of the MDP. Thus $\epsilon$ depends on the complexity of the graph geometry of the MDP induced by the transition model and by all possible policies. For instance, in a MDP with a few well-separated paths leading to final states (a tree structure, for example), $\epsilon$ is of the order of the number of branches. While it can be large, $\epsilon$ is constant with respect to $N$, and thus we “get rid of it” in the bound with large $N$. The intuition is the number of different paths: if one has a single chain path, where one can only move forward, that term is 1; if one has multiple separated single chains, that term is the number of paths. In a task with $p$ different independent paths, the exploration has a factor $p$ in the complexity. The fact that the CE bound contains a term conveying all this geometric information is, in our opinion, an additional bonus.

Remark 4: In conclusion, Theorem 1 tells us that on average, the maximum possible states visitation in $Q_N$ is bounded from above by the average of the $\epsilon_n$ and a constant divided by $N$, and thus is arbitrarily small for a larger and larger number of steps. This, combined with a simulated reset model with guaranteed coverage outside $Q_N$ given by (6), can be used to prove theoretical convergence bounds of $\text{opt}$ without the coverage assumption needed by $\text{opt}$ to be PAC. We give an example of this procedure in the next subsection.

C. The Improvement Phase With REINFORCE

In Sections VI-A and VI-B, we utilized $\text{opt}$ as optimizer in the CE exploration strategy. Now we take the output of this exploration strategy, that is, the exploration policies, and use them together with Algorithm 1 to estimate an empirical distribution $\mu_N$. We then use REINFORCE with the reset model $\mu_N$ obtained by CE. The algorithm obtained is denoted by REINFORCE+CE.

With a careful choice of parameters, and a restart distribution $\mu$ with coverage, REINFORCE can be proved to be PAC. Indeed, after $i$ episodes, with probability at least $1 - \delta$, Theorem 6 of [18] shows that REINFORCE returns a policy $\pi$ such that

$$V_{\pi^*, r, \gamma}(s) - V_{\pi, r, \gamma}(s) \leq C \frac{|S|^2 |A|^2}{(1 - \gamma)^2} \log(i/\delta)^{5/2} \frac{1}{i^{1/6}} \left\| d_{\mu_0, \gamma}^{\pi^*, \gamma} \right\|_\infty$$

for a universal constant $C$, independent of the specific MDP. The last term on the right-hand side, known as the mismatch coefficient (between the state visitation distribution $d_{\mu_0, \gamma}^{\pi^*, \gamma}$ and the $\mu$-reset model), clearly shows, at this level, the necessity of the coverage assumption for $\mu$.

If we now use Theorem 1 with the bound (8) on the error provided by REINFORCE+CE, we are able to measure the overall visitation of the set of poorly visited states in this context. Please note that the coverage assumption of [18] is no more necessary, because coverage in REINFORCE+CE is provided by CE.

Corollary 5: Let an MDP $\mathcal{M} = (S, s_0, A, r, \mathcal{P})$, a $s_0$-reset model and a discount factor $\gamma$ be given. Set $\text{opt} =$REINFORCE, tuned according to [18, Theorem 6], and perform it for $i(n)$ episodes at each step $n$, with $\beta_n := \delta/N$.

Then for every step $N$, with probability at least $1 - \delta$ we have

$$\max_{\pi \in \Pi} \sum_{s \in Q_n} d_{\pi_0, \gamma}^{\pi}(s) \leq \frac{4}{N} \left( C \frac{|S|^2 |A|^2}{(1 - \gamma)^2} \log N \sum_{n=0}^{N} \frac{\log(i(n)/\delta)^{5/2}}{i(n)^{1/6}} \max_{\pi \in \Pi} \frac{d_{\pi_0, \gamma}^{\pi}}{\mu_n} \right)^2 + \sum_{s \in S} \beta_n(s) + \sum_{s \in S} \max_{\pi \in \Pi} d_{\pi_0, \gamma}^{\pi}(s)$$

Finally, we obtain an estimate of the reward obtained by REINFORCE+CE.

Corollary 6: Let an MDP $\mathcal{M} = (S, s_0, A, r, \mathcal{P})$, a $s_0$-reset model and a discount factor $\gamma$ be given. Perform CE to obtain a simulated $\mu_N$-reset model. Perform REINFORCE on $\mathcal{M} = (\mathcal{S}, \mu_N, A, r, \mathcal{P})$ and call $\pi$ the policy returned after $i$ episodes. Then with probability at least $1 - \delta$ it holds

$$V_{\pi^*, r, \gamma}(s_0) - V_{\pi, r, \gamma}(s_0) \leq C \frac{|S|^2 |A|^2}{(1 - \gamma)^2} \frac{\log(i/\delta)^{5/2}}{i^{1/6}} \left\| d_{\mu_0, \gamma}^{\pi^*, \gamma} \right\|_\infty$$

A good bound on the above mismatch coefficient would show that REINFORCE+CE is PAC without any coverage assumption, see Section VIII.

Remark 7: Since the exploration policies returned by Algorithm 2 have coverage on every reachable state by construction, the denominator in (10) is never zero. Thus, the coverage assumption of [18] is removed, and Corollary 6 supports the claim made in Section III that in a broad sense, $CE$ can be used to “remove the coverage hypothesis” from existing algorithms.

Remark 8: The quantities used for the bound in Theorem 1 are very specific to CE. Other algorithms would use quantities quite different in nature, so it seems difficult to compare them in general. Sometimes this is possible: for instance, HOMER, the algorithm introduced in [21], uses the MDP time horizon $H$, potentially huge, for its theoretical bound. In fact, its bound needs exactly $H$ steps, while CE bounds are valid after any number of steps $N$. Another example is $R$-MAX, that has a polynomial bound in states, actions, and mixing time $T$: for every starting state, after $T = T(e)$ steps, the policy is $\epsilon$-close to the optimal one. CE bound in Corollary 6, instead, depends only on states, actions and $\gamma$. However, notice that $R$-MAX is more general than CE, because it covers zero-sum stochastic games, not only MDPs.
A. Exploration Efficiency of CE With the Optimizer TRPO

Exploration efficiency of CE can be understood by looking at the evolution of the simulated \( \mu_n \)-reset models in Fig. 1. The improved exploration on the non-locked paths appears already at the first steps. A video in Supplementary Material shows the exploration evolution for all steps.

In the appendix we provide an additional representation of how CE improves exploration, by scatterplots showing how much states approach the uniform distribution as the iterations of CE increase.

B. Improved Learning Performance of TRPO

TRPO can be used on the lock with the simulated reset model given by CE. We denote this instance of TRPO by TRPO+CE. Here we compare the performance of TRPO and TRPO+CE by looking at the return averaged over episodes. We use undiscounted return, because episodes have constant length. Results are shown in Fig. 2. The sparsity of the reward makes learning very difficult for TRPO alone (red), and its average return stalls at around 0.2, while TRPO+CE (green) collects an average return of almost 0.5.

For TRPO alone, a policy value improvement never happens on the first 1000 episodes. Moreover, the average is outside the quartile, which means that the improvement is due to rare “lucky breaks”, when TRPO learns everything right away. For TRPO+CE, after the initial exploration phase (flat green line), the first policy value improvement happens after less than 500 episodes (see additional Figures in Supplementary Material) and later the average stays inside the quartile. As a comparison, a uniform random policy would find a positive reward with probability \( 1/2^{21} \).

C. Comparison of CE Against MBIE-EB and R-MAX

Since MBIE-EB and R-MAX perform a curiosity-driven exploration with a strategy similar to CE (see Related Work), we performed a comparison between them.

For MBIE-MB, due to the unavailability of source code, we made a comparison using the original tasks presented in the MBIE-EB paper, namely RiverSwim and SixArms [30, Fig. 1, page 1327]. In this way, we can use the performances described in the bar plots [30, Figs. 2 and 3, page 1328]. Moreover, since in those bar plots MBIE-MB is compared with MBIE, \( E^3 \), and R-MAX, we get indirectly a comparison of CE against those other algorithms.

The cumulative reward obtained by TRPO+CE on RiverSwim after 5000 steps, averaged on 20 runs, is similar to that of the other four algorithms, see Fig. 3, and is actually the cumulative reward of the optimal policy. This likely means that RiverSwim at depth 6 is an easy task for algorithms dedicated to efficient exploration.
exploration. On SixArms, MBIE-EB and MBIE perform much better than R-MAX and E³, and slightly better than TRPO+CE. However, TRPO+CE performs significantly better than R-MAX and E³, see Fig. 4. This is probably due to the high efficiency of the TRPO optimizer used with CE.

We also measured the performance of R-MAX in the tasks used in this paper (Consecutive Crossroads Trap and Diabolical Combination Lock at depths 5, 10, and 20), to have a more direct comparison between the exploration strategy of R-MAX and CE. For this, we adapted the R-MAX source code available at https://github.com/ambarishgurjar/R-Max-Reinforcement-Learning-Algorithm-for-Exploration. On CCT5 and DCL5 both R-MAX and TRPO+CE perform almost optimally. As for RiverSwim at depth 6, this likely means that CCT5 and DCL5 are not a challenge for algorithms dedicated to efficient exploration. On CCT10, TRPO+CE still finds an almost-optimal policy, while R-MAX performance is slightly below optimal, see Fig. 5. DCL10 appears more challenging for R-MAX, where it collects an average reward of around 0.8, while the policy found by TRPO+CE is still almost-optimal, see Fig. 6. Finally, we tested TRPO+CE and R-MAX on DCL20, a very hard exploration task. Both algorithms perform suboptimally, with TRPO+CE learning faster than R-MAX. However, notice that while TRPO+CE has converged after around 80000 episodes, R-MAX has still room for improvement, because it has not yet converged after 100000 episodes.

Summing up, TRPO+CE appears slightly better than R-MAX, at least on the tasks used in this paper. A more thorough comparison, with several tasks of different nature, different optimizers opt, and with a separate hyperparameters optimization for TRPO+CE and R-MAX, is subject of future work.
name a few) as $\text{opt}$, and compare $\text{opt}$ and $\text{opt} + \text{CE}$ on several tasks. In order to measure the improvement given by CE, the tasks should be so hard that $\text{opt}$ does not completely explore it.

Theorem 1 assumes exactly values of visitation distributions, which is unfeasible. In fact, we estimate $d_{\pi_{n-1}}$, by running visit for $i(n)$ samples and counting occurrences of each state $(i(n)$ can be the same used for $\text{opt}$ without influencing asymptotic convergence rate). Counting visits directly during the learning phase of $\text{opt}$ would be more sample efficient, but less precise, since the policy changes in the process. If we set a tolerance margin around $\beta$ and use Azuma-Hoeffding to bound from above the probability of being out of that interval, this bound is exponentially decreasing in the number of samples, because samples are independent. Thus, Theorem 1 still holds with the estimates in place of exact values. However, an explicit computation of the bound as a function of $i(n)$ would be interesting.

The solving procedure maintains a strong separation between exploration and improvement. This is one of the original contributions, see Section II, and is therefore strongly pursued in this paper. However, it is not sample efficient. To improve sample efficiency, one could use off-policy learning with the true rewards during the exploration phase. Moreover, the two phases can be performed at once considering the maximum between intrinsic and extrinsic reward, in the spirit of $\text{R-MAX}$.

CE works only in the tabular setting. To overcome this limitation, one could use pseudo-counts to define poorly visited states, and use the theoretical analysis in [32].

Finally, to the best of our knowledge, this is the first paper trying to explicitly avoid the widespread theoretical assumptions of coverage and ergodicity. We believe that research in this direction will foster innovative theoretical results in RL.

**ACKNOWLEDGMENT**

Computational resources provided by CLAI - Computational Logic and Artificial Intelligence laboratory, University of Chieti-Pescara, and by the Maths Department of the University of Pisa.

A heartfelt thank to Leonardo Robol for his help and assistance.

**REFERENCES**

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.

[2] H. Robbins and S. Monro, “A Stochastic Approximation Method,” Ann. Math. Statist., vol. 22, no. 3, pp. 400–407, 1951.

[3] C. Dann, T. Lattimore, and E. Brunskill, “Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5713–5723.

[4] J. Bhandari and D. Russo, “Global optimality guarantees for policy gradient methods,” 2019, arXiv: 1906.01786.

[5] B. Liu, Q. Cai, Z. Yang, and Z. Wang, “Neural trust region/proximal policy optimization attains globally optimal policy,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 1–12.

[6] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” 2017, arXiv: 1707.06347.

[7] L. Wang, Q. Cai, Z. Yang, and Z. Wang, “Neural policy gradient methods: Global optimality and rates of convergence,” in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–46.

[8] V. Konda and J. Tsitsiklis, “Actor-critic algorithms,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 1008–1014.

---

**VIII. LIMITATIONS AND FUTURE WORK**

PAC guarantees in [18] are built over a result of [10], stating:

$$V_{\pi, \gamma}(s_0) - V_{\pi, \gamma}(s_0) \leq \frac{1}{1-\gamma} \left\| \nabla \pi V_{\pi} \right\|_{\infty}$$

If the true gradient is small, so is the error. It is indeed in this key step that the distribution mismatch appear, and with it the need of coverage. The logic of theoretical bounds in CE can be used to avoid this distribution mismatch dependence. In fact, CE provides two “orthogonal” guarantees on the visitation distribution: a pointwise lower bound outside $Q_n$, given by (6), and an average upper bound inside $Q_n$, given by (7). More explicitly, Theorem 1 gives

$$V_{\pi, \gamma}(s_0) - V_{\pi, \gamma}(s_0) \leq \frac{1}{1-\gamma} \sum_{s \notin Q_n} d_{\pi_{n-1}}(s) + \frac{1}{1-\gamma} \left\| \nabla \pi V_{\pi} \right\|_{\infty} \sum_{s \notin Q_n} d_{\pi_{n-1}}(s)$$

In perspective, the next step in the analysis of CE will be to derive an upper bound on the distribution mismatch coefficient in $Q_n$ using the theoretical bounds of CE itself. Indeed, so far we only know that $\mu_N$ is soft and the mismatch is finite, with explicit bounds outside $Q_n$, but not in $Q_n$. This is by far the main limitation of this paper. Experiments show although that the mismatch is usually not too big.

Another important limitation was due to a lack of computational resources. Since HOMER in [21] is similar from several points of view to CE, an important future development is an experimental evaluation of CE on the very same tasks as in HOMER, that is, Combination Locks with depth up to 100, ten actions and anti-shaped reward.

In this paper, we do not try to support the claim that CE improves the exploration of $\text{opt}$, otherwise we would have chosen $\text{opt}$ different from REINFORCE and TRPO. However, it is not implausible that CE could be used to improve exploration strategies. To support this claim one should select several state-of-the-art exploration algorithms (Go-Explore and HOMER to
[9] S. M. Kakade, “A natural policy gradient,” in Proc. Adv. Neural Inf. Process. Syst., 2001, pp. 1531–1538.
[10] A. Agarwal, S. M. Kakade, J. D. Lee, and G. Mahajan, “On the theory of policy gradient methods: Optimality, approximation, and distribution shift,” 2019, arXiv:1908.00261.
[11] A. Agarwal, S. M. Kakade, J. D. Lee, and G. Mahajan, “Optimality and approximation with policy gradient methods in Markov decision processes,” in Proc. 33rd Conf. Learn. Theory, 2020, pp. 64–66.
[12] Y. Abbasi-Yadkori, P. Bartlett, K. Bhatia, N. Lazic, C. Szepesvari, and G. Weisz, “POLITEX: Regret bounds for policy iteration using expert prediction,” in Proc. 35th Int. Conf. Mach. Learn., 09-15 Jun. 2019, pp. 3692–3702.
[13] C. Wei, M. Jafaria-Jahromi, H. Luo, H. Sharma, and R. Jain, “Model-free reinforcement learning in infinite-horizon average-reward Markov decision processes,” in Proc. 37th Int. Conf. Mach. Learn., 2020, pp. 10170–10180.
[14] R. Ormer, “Regret bounds for reinforcement learning via Markov chain concentration,” J. Artif. Intell. Res., vol. 67, pp. 115–128, 2020.
[15] R. L. Tweedie, “Criteria for ergodicity, exponential ergodicity and strong ergodicity of Markov processes,” J. Appl. Probability, vol. 18, pp. 122–130, 1981.
[16] M. J. Kearns and S. P. Singh, “Near-optimal reinforcement learning in polynomial time,” Mach. Learn., vol. 49, no. 2/3, pp. 209–232, 2002.
[17] A. Sharma, R. Ahmad, and C. Finn, “A state-distribution matching approach to non-episodic reinforcement learning,” in Proc. Int. Conf. Mach. Learn., Baltimore, Maryland, USA, 2022, pp. 1964–19657.
[18] J. Zhang, J. Kim, B. O’Donoghue, and S. P. Boyd, “Sample efficient reinforcement learning with REINFORCE,” 2020, arXiv:2010.11364.
[19] A. Agarwal, M. Henaff, S. M. Kakade, and W. Sun, “PC-PC: Policy cover directed exploration for provable policy gradient learning,” in Proc. Adv. Neural Inf. Process. Syst. 33: Ann. Conf. Neural Inf. Process. Syst., 2020, pp. 13399–13411.
[20] A. Ecoffet, J. Huizinga, J. Lehman, K. O. Stanley, and J. Clune, “First return, then explore,” Nature, vol. 590, pp. 580–586, Feb. 2021.
[21] D. Misra, M. Henaff, A. Krishnamurthy, and J. Langford, “Kinematic state abstraction and provably efficient rich-observation reinforcement learning,” in Proc. 37th Int. Conf. Mach. Learn., 2020, pp. 6961–6971.
[22] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, “Curiosity-driven exploration by self-supervised prediction,” in Proc. 34th Int. Conf. Mach. Learn., 2017, pp. 2778–2787.
[23] A. P. Badia et al., “Never give up: Learning directed exploration strategies,” in Proc. 8th Int. Conf. Learn. Representations, Addis Ababa, Ethiopia, Apr. 26-30, 2020, pp. 1–28.
[24] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. D. Turck, and P. Abbeel, “VIME: Variational information maximizing exploration,” in Proc. Adv. Neural Inf. Process. Syst. 29: Ann. Conf. Neural Inf. Process. Syst. Process. Syst., 2016, pp. 1109–1117.
[25] T. Yang et al., “Exploration in deep reinforcement learning: From single-agent to multiagent domain,” IEEE Trans. Neural Netw. Learn. Syst., vol. 35, no. 7, pp. 8762–8782, 2020.
[26] P.-Y. Oudeyer and F. Kaplan, “What is intrinsic motivation? A typology of computational approaches,” Front. Neurorobot., vol. 1, 2007, Art. no. 108.
[27] C. Bai et al., “Principled exploration via optimistic bootstrapping and backward induction,” in Proc. 38th Int. Conf. Mach. Learn., 2021, pp. 577–587.
[28] M. Gaon and R. I. Brafman, “Reinforcement learning with non-Markovian rewards,” in Proc. 34th AAAI Conf. Artif. Intell., AAAI 2020, 32nd Innov. Appl. Artif. Intell. Conf., 10th AAAI Symp. Educ. Adv. Artif. Intell., New York, NY, USA, Feb. 07-12, 2020, pp. 3980–3987.
[29] J. Liu et al., “Ovd-explorer: Optimism should not be the sole pursuit of exploration in noisy environments,” in Proc. 38th AAAI Conf. Artif. Intell., 36th Conf. Innov. Appl. Artif. Intell., 14th Symp. Educ. Adv. Artif. Intell., 2024, pp. 13954–13962.
[30] A. L. Strehl and M. L. Littman, “An analysis of model-based interval estimation for Markov decision processes,” J. Comput. Syst. Sci., vol. 74, no. 8, pp. 1309–1311, 2008.
[31] J. Z. Kolter and A. Y. Ng, “Near-Bayesian exploration in polynomial time,” in Proc. 25th Ann. Int. Conf. Mach. Learn., 2009, pp. 513–520.
[32] M. G. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, “Unifying count-based exploration and intrinsic motivation,” in Proc. Adv. Neural Inf. Process. Syst. 29: Ann. Conf. Neural Inf. Process. Syst. 2016, pp. 1471–1479.
[33] H. Tang et al., “#Exploration: A study of count-based exploration for deep reinforcement learning,” in Proc. Adv. Neural Inf. Process. Syst. 30: Ann. Conf. Neural Inf. Process. Syst., 2017, pp. 2753–2762.
[34] R. I. Brafman and M. Tennenholtz, “R-MAX - A general polynomial time algorithm for near-optimal reinforcement learning,” J. Mach. Learn. Res., vol. 3, pp. 213–231, 2002.
[35] S. S. Du, Y. Luo, R. Wang, and H. Zhang, *Provably Efficient Q-Learning with Function Approximation via Distribution Shift Error Checking Ora- cle*, Red Hook, NY, USA: Curran Associates Inc., 2019, pp. 8060–8070.
[36] S. M. Kakade and J. Langford, “Approximately optimal approximate reinforcement learning,” in Proc. 19th Int. Conf. Mach. Learn., Morgan Kaufmann, 2002, pp. 267–274.
[37] A. L. Strehl, L. Li, and M. L. Littman, “Reinforcement learning in finite MDPs: PAC analysis,” J. Mach. Learn. Res., vol. 10, pp. 2413–2444, 2009.
[38] I. Kostrikov, “PyTorch implementation of TRPO.” 2018. [Online]. Available: https://github.com/ikostrikov/pytorch-trpo#pytorch-implementation-of-trpo

Marco Miani received the undergraduate degree in mathematics from the University of Pisa and subsequently earned a second-level master’s degree from the Scuola Normale Superiore, achieving full marks in all degrees. He is currently working toward the PhD degree in the cognitive systems section with the Technical University of Denmark (DTU). Currently, his research is centered on the application of differential geometry to uncertainty quantification in Bayesian neural networks.

Maurizio Parton received the PhD degree in mathematics from the University of Pisa, in 2001. He then joined the University of Chieti-Pescara, where he is currently full professor in geometry. In 2016, impressed by the achievements of AlphaGo, he began exploring reinforcement learning, and then deep learning. Today, he is dedicated to the mathematical foundations underlying reinforcement learning, deep learning, and geometric deep learning. Believing that interdisciplinary approaches are crucial for tackling the multifaceted challenges of our complex world, he has engaged in several interdisciplinary research projects that integrate AI with game theory, computer science, graph theory, biology, health, and computational law. He is deeply committed to disseminating the fundamental principles and techniques of AI.

Marco Romito received the PhD degree in mathematics from the University of Pisa, in 2001. He then joined the Department of Mathematics of the University of Florence as Lecturer in mathematical analysis. He is currently full professor in probability and mathematical statistics with the University of Pisa. He has been visiting research member at MSR Berkeley, HIM Bonn, Newton Institute Cambridge, Bernoulli centre Lausanne, and invited professor with ENS Bretagne and Center for Mathematical Sciences, Wuhan. His research interests lie in stochastic analysis, stochastic PDEs and the mathematical theory of turbulence. Currently his research interests include also the mathematics of machine learning and of artificial intelligence.