CEM: Constrained Entropy Maximization for Task-Agnostic Safe Exploration

Yang, Q.; Spaan, M.T.J.

Publication date
2023

Document Version
Final published version

Published in
The Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)

Citation (APA)
Yang, Q., & Spaan, M. T. J. (2023). CEM: Constrained Entropy Maximization for Task-Agnostic Safe Exploration. In *The Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)*

CEM: Constrained Entropy Maximization for Task-Agnostic Safe Exploration

Qisong Yang, Matthijs T. J. Spaan

Delft University of Technology, The Netherlands
{q.yang, m.t.j.spaan}@tudelft.nl

Abstract

In the absence of assigned tasks, a learning agent typically seeks to explore its environment efficiently. However, the pursuit of exploration will bring more safety risks. An under-explored aspect of reinforcement learning is how to achieve safe efficient exploration when the task is unknown. In this paper, we propose a practical Constrained Entropy Maximization (CEM) algorithm to solve task-agnostic safe exploration problems, which naturally require a finite horizon and undiscounted constraints on safety costs. The CEM algorithm aims to learn a policy that maximizes state entropy under the premise of safety. To avoid approximating the state density in complex domains, CEM leverages a k-nearest neighbor entropy estimator to evaluate the efficiency of exploration. In terms of safety, CEM minimizes the safety costs, and adaptively trades off safety and exploration based on the current constraint satisfaction. The empirical analysis shows that CEM enables the acquisition of a safe exploration policy in complex environments, resulting in improved performance in both safety and sample efficiency for target tasks.

Introduction

Despite the remarkable achievements in many fields, safety and exploration are still the main challenges faced by reinforcement learning (RL; Sutton and Barto 2018; Mnih et al. 2015). Exploration is critical to avoid the learning agent finally converging into a suboptimal policy. However, in safety-critical domains, unlimited exploration is unacceptable (Dulac-Arnold et al. 2021; García and Fernández 2015). For instance, while running a power network, an agent trying unlimited exploration could cause a blackout (Marot et al. 2020; Subramanian et al. 2021). Hence, encouraging exploration is bound to increase safety risks.

Many learning problems may start from an unsupervised setting. The knowledge gained can make an agent easier to achieve a variety of tasks later. When employing a safe exploration policy as a safe guide (SaGui; Yang et al. 2022a), an agent can adapt safely and quickly to a revealed task, especially when the task's reward signal is sparse. In this paper, we focus on learning such a task-agnostic safe exploration policy. While task-agnostic exploration has been given attention (Lee et al. 2019; Hazan et al. 2019; Tao, François-Lavet, and Pineau 2020; Badia et al. 2019; Mutti, Pratissoli, and Restelli 2021; Seo et al. 2021; Liu and Abbeel 2021b), its safety aspects are still under-explored.

Prior approaches to boosting exploration usually shape the reward signal using an exploration bonus (Stadie, Levine, and Abbeel 2015; Bellemare et al. 2016; Ostrovski et al. 2017; Tang et al. 2017; Pathak et al. 2017; Haarnoja et al. 2018a,b; Fox, Choshen, and Loewenstein 2018; Sun et al. 2019; Pathak, Gandhi, and Gupta 2019; Burda et al. 2019a,b; Seo et al. 2021). Most of them are based on a measure of state novelty to lead the agent to new unseen states. However, these typically heuristic measures are not part of the optimization objectives. They are designed to only transiently affect the process of learning, but not the final result. In contrast, to quantify exploration in a more principled way, Lee et al. (2019); Hazan et al. (2019); Tao, François-Lavet, and Pineau (2020); Badia et al. (2019); Mutti, Pratissoli, and Restelli (2021); Seo et al. (2021); Liu and Abbeel (2021b) propose to encourage uniform coverage of the state space. With an explicit target to maximize the entropy of the state density, the interpretability of the learned exploration policy is improved significantly (Seo et al. 2021).

In safe RL, it is natural to formulate safety concerns by constraints (Achiam et al. 2017; Qin, Chen, and Fan 2021; Yang et al. 2021, 2022b). In this case, safety can be decoupled from reward to mitigate the issue of constructing a single reward signal that must carefully trade off task performance and safety. When our focus is solely on efficient exploration, however, it is not clear how we can design a traditional reward signal to maximize the state entropy. With additional safety concerns, it is even more challenging to construct a single reward signal that is sensible for both safety and exploration. Therefore, in task-agnostic safe exploration, the need to treat safety as a constraint is exacerbated.

In safety-constrained RL problems, the discounted long-term costs are usually constrained within a pre-defined cost limit (Achiam et al. 2017; Liu, Ding, and Liu 2020; Yang et al. 2020; Kamran et al. 2022). However, for industrial and robotic settings (Jardine, Lin, and Banjevic 2006; Boutilier and Lu 2016; De Nijs, Spaan, and de Weerdt 2015), the safety constraints are always built on the real costs within a finite horizon instead of the discounted cost-return. For instance, a safety constraint for an electric vehicle is based on its real battery capacity, so the battery consumption cannot be discounted.

In this paper, we aim to achieve safe and efficient exploration when the so-called target task is unknown. We propose to formulate the problem by maximizing the entropy of the state density under safety constraints. Likewise, we designed the Constrained Entropy Maximization (CEM) algorithm, which leverages the k-nearest neighbor state entropy estimator to avoid approximating the full state density, which hardly scales to complex domains (Hazan et al. 2019; Lee et al. 2019). Based on the real costs, CEM leverages an adaptive safety weight (Lagrangian multiplier) to automatically trade off exploration and safety during policy updates. We improve the safety of the policy by calculating the gradient on the discounted cost-return but updating the safety-weight following the undiscounted real costs.

Summarizing, our main contributions are as follows: *i*) we propose a practical and approximately convergent CEM algorithm for task-agnostic safe exploration problems, *ii*) and we empirically show that CEM enables the acquisition of a safe exploration policy in the complex domain, and that the policy benefits the target tasks.

Preliminaries

In this section, we present the background and notation.

Constrained Markov Decision Processes

A constrained Markov decision process (CMDP; Altman 1999; Borkar 2005) is defined as a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, c, d, T, \iota \rangle$: a state space $\mathcal{S}$, an action space $\mathcal{A}$, a probabilistic transition function $\mathcal{P}: \mathcal{S} \times \mathcal{A} \to Dist(\mathcal{S})$, a reward function $r: \mathcal{S} \times \mathcal{A} \to [r_{min}, r_{max}]$, a cost function $c: \mathcal{S} \times \mathcal{A} \to [c_{min}, c_{max}]$, a given safety threshold d, a time horizon T, and an initial state distribution $\iota \in Dist(\mathcal{S})$. An MDP (Puterman 2014) can be seen as an unbounded CMDP with $d = \infty$. A stationary policy $\pi : \mathcal{S} \to Dist(\mathcal{A})$ is a map from states to probability distributions over actions, with $\pi(a|s)$ denoting the probability of selecting action a in state s. We denote the set of all stationary policies by Π.

In this work, an agent interacts with a CMDP, without knowledge about the transition, reward, and cost functions, generating a trajectory $\tau \in \mathcal{T}$, which is a sequence of transitions $\langle (s_0, a_0, r_0, c_0, s'_0), (s_1, a_1, r_1, c_1, s'_1), \cdots \rangle$. A trajectory starts from $s_0 \sim \iota(\cdot)$, then, at each timestep t the agent is in a state $s_t \in \mathcal{S}$, and takes an action $a_t \in \mathcal{A}$. Then, it gets a reward $r_t = r(s_t, a_t)$, a cost $c_t = c(s_t, a_t)$, and steps into a successor state $s'_t \sim \mathcal{P}(\cdot \mid s_t, a_t)$. This process repeats starting from $s_{t+1} = s'_t$, until some terminal condition is met, such as reaching the time horizon T. Then, a new trajectory starts.

For a complex and long-horizon problem, it is common to introduce a discount factor γ to make the problem tractable. In this paper, we assume that we have no access to the reward signal. For safety-costs, we express the value function as $V^c_\pi(s) = \mathbb{E}_{(s_t,a_t)\in\mathcal{T}_\pi}[\sum_{t=0}^{\infty} \gamma^t c_t | s_0 = s]$ and action-value function as $Q^c_\pi(s,a) = \mathbb{E}_{(s_t,a_t)\in\mathcal{T}_\pi}[\sum_{t=0}^{\infty} \gamma^t c_t | s_0 = s, a_0 = a]$. The advantage function for costs is $A^c_\pi(s,a) = Q^c_\pi(s,a) - V^c_\pi(s)$. In a traditional CMDP, the goal of the agent is to learn a policy that maximizes the expected return for each episode such that the generated costs remain below the given threshold d. In this paper, we have a similar formulation for the constraint, but a completely different optimization objective related to the state entropy.

Induced State Density

We use a state density function $\rho : \mathcal{S} \mapsto \mathbb{R}_{\geq 0}$ that quantifies the distribution of states within the state space $\mathcal{S}$. When a policy π is applied to a CMDP, it influences the state distribution over time. For each time step t, the state density function $\rho^\pi_t(s)$ calculates the concentration of states at that moment. The initial state distribution ι serves as the starting point, and the policy interaction with the CMDP produces the state density $\rho^\pi_t(s) = \rho(s_t = s|\pi)$ for each subsequent time step $t > 0$. For CMDPs with finite horizon T, the stationary density of state s can be expressed as:

$$\rho^\pi_T(s) = \frac{1}{T}\sum_{t=1}^{T} \rho^\pi_t(s), \qquad (1)$$

which is the average state density, and $\int_{\mathcal{S}} \rho^\pi_T(s)\mathrm{d}s = 1$.

However, a full model of state density estimation for $\rho^\pi_T(s)$ does not scale easily to complex domains (Hazan et al. 2019; Lee et al. 2019), where we need to avoid modeling the state density directly. We choose to use the k-nearest neighbors (k-NN) entropy estimator $\hat{\mathcal{H}}^k_N(\rho)$ by a group of particles $\{s_i\}_{i=1}^N$ to avoid estimating the state density directly (Singh et al. 2003). In the RL process, we may need to use the samples from the current policy to estimate the state entropy of the target policy, for which we can employ an Importance-Weighted (IW) k-NN estimator $\hat{\mathcal{H}}^k_N(\rho|\rho')$ (Ajgl and Šimandl 2011). We refer the reader to Mutti, Pratissoli, and Restelli (2021) for the detailed expression of $\hat{\mathcal{H}}^k_N(\rho|\rho')$.

Task-Agnostic Safe Exploration

In this section, we define the learning objective and safety for task-agnostic safe exploration (TASE), where only safety signals are provided. Without the reward signal, the agent aims to explore the world safely and efficiently. The obtained policy with safe exploration capabilities may provide useful prior knowledge required to enhance the safety in potential target tasks.

In this paper, we focus on a finite-horizon setting like the work by Lee et al. (2019); Mutti, Pratissoli, and Restelli (2021). Most real-world constrained RL problems naturally require a finite horizon and typically constraints on safety costs do not include discounts, which also mitigates the problem of designing a safety threshold based on the discounted cost-return (Walraven and Spaan 2018). For instance, an electric vehicle can take its battery capacity as the cost limit d. Naturally, we can select the horizon T in alignment with the horizon of the target task that the policy is expected to confront. When the target task is not clear, we can tune T to balance the exploration efficiency and quality (Mutti, Pratissoli, and Restelli 2021).

Definition 1 (Safety within a finite horizon)**.** *A policy π is safe if its expected accumulated costs $\mathbb{E}_{(s_t,a_t)\sim\mathcal{T}_\pi}\left[\sum_{t=1}^{T} c_t\right]$ over finite-horizon T remains below a safety threshold d.*

Then, we formulate the TASE problem as maximizing the entropy of the *average state density* under the premise of safety:

$$\max_{\pi\in\Pi} \mathcal{H}(\rho_T^\pi) \text{ s.t.} \mathop{\mathbb{E}}_{(s_t,a_t)\sim\mathcal{T}_\pi}\left[\sum_{t=1}^{T} c_t\right] \leq d. \tag{2}$$

We are particularly interested in problems where the set of initial states is small, since they are more challenging for task-agnostic exploration. If the trajectory can start at any state, it will be meaningless to maximize the state entropy.

Safety-Constrained Entropy Maximization

In this section, we first clarify that traditional value function based methods are not suitable for maximizing the state entropy when the original environment reward does not exist. To achieve task-agnostic safe exploration (TASE), we will establish the duality of the original problem, then propose a practical algorithm called Constrained Entropy Maximization (CEM) for TASE with convergence guarantees.

Vulnerable Reliance on Return

Traditional RL agents learn from the reward signal when interacting with the environment. We call this original environment signal as *extrinsic reward*, and the signal designed for encouraging exploration as *intrinsic reward*. When we have no access to the extrinsic reward, it is important to ask whether we can design an intrinsic reward, such that we can solve the TASE problems by traditional RL methods.

When learning is only for exploration without extrinsic rewards, we need to design an intrinsic reward that is stationary and implies efficient exploration of the environment in the standard RL framework. Many different intrinsic rewards are designed in previous works, e.g., count-based exploration (Bellemare et al. 2016; Ostrovski et al. 2017), prediction-based exploration (Stadie, Levine, and Abbeel 2015; Pathak et al. 2017), and auxiliary task (Fox, Choshen, and Loewenstein 2018; Burda et al. 2019a). However, they are not easy to be generalized to explicitly maximize the state entropy in the task-agnostic setting.

Duality of Constrained Entropy Maximization

The standard RL algorithms that optimize long-term rewards cannot solve the TASE problem (2) directly. Even for constrained RL algorithms, traditional RL rewards are also necessary. Without a reward signal, the TASE problem (2) is dual to a problem that is solvable in a Lagrangian way. We denote the Lagrangian multiplier for $\mathbb{E}_{(s_t,a_t)\sim\mathcal{T}_\pi}[\sum_{t=1}^{T} c_t] \leq d$ as $\omega : \Pi \to \mathbb{R}_{\geq 0}$. Note that ω is an overall safety evaluation of the current policy and does not depend on the state. Then we consider the following optimization problem:

$$\min_{\omega\geq 0} \max_{\pi} \mathcal{G}(\pi, \omega) \doteq f(\pi) - \omega g(\pi), \tag{3}$$

where $f(\pi) = \mathcal{H}(\rho_T^\pi)$, and $g(\pi) = \mathbb{E}_{(s_t,a_t)\sim\mathcal{T}_\pi}[\sum_{t=0}^{T} c_t] - d$. Alternating between optimizing π and ω can gradually adjust the Lagrange multiplier until the Karush-Kuhn-Tucker (KKT; Gordon and Tibshirani 2012) condition $\omega g(\pi) = 0$ is satisfied.

We search for a policy within a parametric space of stochastic differentiable policies $\Pi_\Theta = \{\pi_\theta : \theta \in \Theta\}$. Ideally, we have two loss functions for the constrained optimization problem (2), i.e.,

$$\begin{aligned} J_\pi(\theta) &= \omega g(\theta) - f(\theta), \\ J_s(\omega) &= -\omega g(\theta). \end{aligned} \tag{4}$$

In practice, if we calculate the policy gradient based on $\mathbb{E}_{(s_t,a_t)\sim\mathcal{T}_{\pi_\theta}}\left[\sum_{t=1}^{T} c_t\right]$, the training is likely to be unstable because of the high variance in policy evaluation, especially for complex and long-horizon problems (Kakade 2001; Peters and Bagnell 2010). Instead, we will optimize the policy π by using the gradient of its induced long-term discounted costs, i.e.,

$$\overline{g}(\theta) = \mathbb{E}_{(s_t,a_t)\sim\mathcal{T}_{\pi_\theta}}\left[\sum_{t=0}^{\infty} \gamma^t c_t\right] - \overline{d},$$

where $\overline{d} = \frac{d}{T(1-\gamma)}$ is the discounted approximation of d.

We propose to replace $g(\theta)$ in $J_\pi(\theta)$ by $\overline{g}(\theta)$, but $J_s(\omega)$ remains as in Eq. 4, because the constraint satisfaction of a policy can be easily estimated by the real costs of the sampled trajectories. In the following, we argue that it is valid to optimize the policy π by minimizing the discounted cumulative costs until the original undiscounted cost constraint is satisfied.

Theorem 1. *Let the constrained optimization in* (2) *be feasible with a solution $\mathscr{S}^* = (\theta^*, \omega^*)$ that satisfies the KKT conditions, which is found by minimizing the loss functions* (4)*. Then, $\overline{\mathscr{S}}^* = (\theta^*, \overline{\omega}^*)$ with $\overline{\omega}^* = \frac{\omega^*}{h'(0)}$ is a solution to the problem obtained by replacing $g(\theta)$ in $J_\pi(\theta)$ with $h(g(\theta))$, where $h : \mathbb{R} \to \mathbb{R}$ is a strictly monotone increasing function. The reverse also holds.*

If we have a long episode length $T \gg 1/(1-\gamma)$ and on-policy sampling at each gradient step, $\overline{g}(\theta)$ is approximately an affine function of $g(\theta)$, i.e.,

$$\overline{g}(\theta) \doteq \frac{g(\theta)}{T(1-\gamma)}, \tag{5}$$

Therefore, invoking Theorem 1, we can optimize the policy π for (2) by calculating the gradient on the discounted cost-return $\overline{g}(\theta)$, but updating the safety weight ω based on the undiscounted real costs. We refer the reader to Appendix A for the proof of Theorem 1 and derivation of Eq. 5.

The CEM Algorithm

To solve the safety-constrained entropy maximization problem (2) in complex domains, we propose the CEM method (Algorithm 1) to optimize the policy within Π_Θ. At each gradient step, CEM will perform a series of fine-tuned optimizations centered around the current policy (Schulman et al. 2015). We take the trust region as a constraint to ensure that the optimizations are conducted within a reliable and stable neighborhood of the current policy θ'. Considering the trust-region threshold δ, we are presented with a constrained optimization problem as follows:

$$\max_{\theta \in \Theta} \hat{\mathcal{H}}_k(\rho_T(\theta)) \quad \text{s.t.} \begin{cases} D_{KL}(\rho_T(\theta)||\rho_T(\theta')) \leq \delta \\ \underset{(s_t,a_t)\sim\mathcal{T}_\theta}{\mathbb{E}} \left[\sum_{t=1}^{T} c_t\right] \leq d. \end{cases} \tag{6}$$

Before updating the policy, we can determine the safety weight ω by evaluating the current safety performance. With the current policy parameters θ', we can sample a batch of trajectories of length T (Algorithm 1, lines 3-8). We use λ_π and λ_ω to represent the learning rate for the policy π and safety weight ω respectively. Then, we can update the safety weight (Algorithm 1, line 9) by

$$\omega \leftarrow \max(0, \omega + \lambda_\omega \hat{g}(\theta')), \tag{7}$$

where

$$\hat{g}(\theta') = \frac{1}{N_T} \sum_{n=1}^{N_T} \left[\sum_{t=1}^{T} c_t | (s_t, a_t) \sim \mathcal{T}_{\pi_{\theta'}} \right] - d, \tag{8}$$

where N_T is the number of trajectories. Then, we construct the loss function for the policy

$$J_\pi(\theta) = J_{\mathcal{H}}(\theta) + \omega J_g(\theta), \tag{9}$$

$$\begin{aligned} \text{where} \quad & J_{\mathcal{H}}(\theta) = -\hat{\mathcal{H}}_k(\rho_T(\theta)|\rho_T(\theta')) \\ \text{and} \quad & J_g(\theta) = \overline{g}(\theta) - \overline{g}(\theta'). \end{aligned}$$

The loss function for the state entropy $J_{\mathcal{H}}(\theta)$ is based on the IW k-NN estimator $\hat{\mathcal{H}}_N^k(\rho|\rho')$. Note that the entropy cannot be calculated directly, but needs to be estimated based on the current policy, the target policy, and the sampled particles from the current policy. We can first compute the normalized importance weight for each sample, then approximate the state density $\hat{\mathcal{H}}_N^k(\rho|\rho')$ (Mutti, Pratissoli, and Restelli 2021). Although these samples are not all independent (trajectories are sampled independently, but states within a trajectory are correlated), we observe satisfactory behavior when k and the number of trajectories are sufficiently large.

Notice that we use the surrogate advantage $\overline{g}(\theta) - \overline{g}(\theta')$ to approximate our objective in minimizing the discounted safety costs, and build our loss function J_g for safety. The surrogate advantage is a measure of how the target policy π_θ performs in safety relative to the current policy $\pi_{\theta'}$ using data from $\pi_{\theta'}$ (Schulman et al. 2015), i.e.,

$$J_g(\theta) \doteq \underset{(s_t,a_t)\sim\mathcal{T}_{\pi_{\theta'}}}{\mathbb{E}} \left[\frac{\pi_\theta\,(a|s)}{\pi_{\theta'}\,(a|s)} A^c_{\pi_{\theta'}}\,(s,a) \right], \tag{10}$$

where $A^c_\pi(s,a) = Q^c_\pi(s,a) - V^c_\pi(s)$ is the advantage function for costs. The surrogate advantage is designed for maximizing the long-term return, but it can be easily adapted to minimize the discounted safety costs $\overline{g}(\pi)$ in our setting. We refer the reader to (Schulman et al. 2015) for the proof of Eq. 10.

Algorithm 1: Constrained Entropy Maximization

```
Require: Initial parameters $T$, $N$, $\delta$, $\lambda$, $k$, and $d$
 1: initialize $\theta$, $\omega$, $\mathcal{D} \leftarrow \emptyset$, $\theta' \leftarrow \theta$
 2: for each epoch do
 3:     for each environment step do
 4:         $a_t \sim \pi_{\theta'}(a_t|s_t)$
 5:         $c_t \sim c(a_t|s_t)$
 6:         $s_{t+1} \sim \mathcal{P}(s_{t+1}|s_t, a_t)$
 7:         $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, c_t, s_{t+1})\}$
 8:     end for
 9:     $\omega \leftarrow \max(0, \omega + \lambda_\omega \hat{g}(\theta'))$
10:     while $D_{KL}(\rho_T(\theta)||\rho_T(\theta')) \leq \delta$ do
11:         $\theta \leftarrow \theta + \lambda_\pi \nabla_\theta J_\pi(\theta)$
12:     end while
13:     $\theta' \leftarrow \theta$
14:     $\mathcal{D} \leftarrow \emptyset$
15: end for
Output: Safe exploration policy $\pi_\theta$
```

At each gradient step, we exploit a KL estimator $\hat{D}_{KL}(\rho\|\rho')$ to compute the trust-region constraint. We refer the reader to (Ajgl and Šimandl 2011; Mutti, Pratissoli, and Restelli 2021) for the detailed derivation and expression of $\hat{D}_{KL}(\rho\|\rho')$. While the updated policy satisfies $\hat{D}_{KL}(\rho_T(\theta)||\rho_T(\theta')) \leq \delta$, we can optimize the policy several times (Algorithm 1, lines 10-12) by

$$\begin{aligned} \theta &\leftarrow \theta + \lambda_\pi \nabla_\theta J_\pi(\theta) \\ &= \theta + \lambda_\pi \nabla_\theta J_{\mathcal{H}}(\theta) + \lambda_\pi \omega \nabla_\theta J_g(\theta), \end{aligned} \tag{11}$$

where

$$\nabla_\theta J_g(\theta) = \frac{1}{N} \sum_{n=1}^{N} \nabla_\theta \left[\frac{\pi_\theta\,(a_n|s_n)}{\pi_{\theta'}\,(a_n|s_n)} A^c_{\pi_{\theta'}}\,(s_n, a_n) \right].$$

We employ Theorem 5.1 by Mutti, Pratissoli, and Restelli (2021) to compute the gradient of the IW entropy estimator in $\nabla_\theta J_{\mathcal{H}}(\theta)$, where θ is updated without constraints. Using the Lagrangian cost constraint, we leverage ω to balance safety during policy updates instead of as a reward-shaping factor.

Bounds on Approximate Convergence

In this section, we demonstrate the approximate convergence of CEM to the optimal solution by transforming the problem (3) into the framework by Qin, Chen, and Fan (2021). At each gradient step, CEM updates the policy based on the gradient descent algorithm with the modifications shown by Theorem 1, which finds the direction of the maximum increase in the entropy of the average state density but only considers the immediate surroundings of the current policy. Thus, the policy ascent is noisy due to limited samples and constrained due to the trust-region constraint. Then, the question is whether the gradient descent of the weight ω is sufficiently perturbed to no longer find a solution. Theorem 2 by Qin, Chen, and Fan (2021) has shown that a density-constrained RL algorithm can eventually converge around the optimal policy even under suboptimal policy updates at each gradient step.

The amended Lagrangian optimization from Theorem 1 can be written as:

$$\begin{aligned} &\max_{\omega\geq 0}\mathcal{F}(\omega), \text{where}\\ &\mathcal{F}(\omega)=\omega g(\rho^*(\omega)) - \mathcal{H}(\rho^*(\omega)), \text{ and}\\ &\rho^*(\omega)=\arg\min_{\rho}\omega\overline{g}(\rho) - \mathcal{H}(\rho). \end{aligned} \tag{12}$$

In this representation, the state density $\rho = \rho_T^\pi$ is implicitly generated by the policy π. The associated discounted safety costs can be expressed as

$$\begin{aligned} \overline{g}(\rho) &= \int_S \rho_T^\pi(s) V_\pi^c(s)\mathrm{d}s - \overline{d}\\ &\approx \frac{1}{1-\gamma}\left[\int_S \rho_T^\pi(s)c(s)\mathrm{d}s - \frac{d}{T}\right], \end{aligned}$$

where $\overline{g}(\rho)$ is (approximately) affine in ρ, and we assumed that costs $c(s)$ are incurred by the presence in states, not by actions.

Theorem 2 by Qin, Chen, and Fan (2021) is built on the assumption that the optimization function is strongly convex. Because $-\mathcal{H}(\rho)$ is convex and $\overline{g}(\rho)$ is (approximately) affine in ρ, our optimization function $\omega\overline{g}(\rho) - \mathcal{H}(\rho)$ in Eq. 12 is convex in ρ but not necessarily strongly convex, which depends on the underlying distribution and the specific form of the entropy measure. Even when $\omega\overline{g}(\rho) - \mathcal{H}(\rho)$ is not strongly convex, we can add a regularization term to enforce strong convexity, ensuring that it has a unique minimum and steepness that increases when moving away from the minimum. When $\omega\overline{g}(\rho) - \mathcal{H}(\rho)$ is strongly convex, let its modulus be μ, which measures the degree of convexity.

We optimize ω to achieve $\max_{\omega\geq 0}\mathcal{F}(\omega)$. The set of its optimal solutions is denoted as $\Omega^* = \{\omega | \mathcal{F}(\omega) = \max_{\omega\geq 0}\mathcal{F}(\omega)\}$, with $\overline{\omega}^* \in \Omega^*$ in line with Theorem 1. For a given ω, we use the TRPO method (with discounted safety costs) to solve the state density optimization problem. For a suboptimal update in policy, we assume the imperfect solution $\hat{\rho}$ satisfies

$$\omega g(\hat{\rho}) - \mathcal{H}(\hat{\rho}) - \mathcal{F}(\omega) \leq \epsilon.$$

The corresponding update in the safety weight is $\omega \leftarrow \max(0, \omega + \lambda_\omega \nabla\hat{\mathcal{F}}(\omega))$, where $\nabla\hat{\mathcal{F}}(\omega) = g(\hat{\rho})$. Then, we can invoke Lemma 2 and Theorem 2 by Qin, Chen, and Fan (2021) to get the following convergence result.

Convergence result For a step size $\lambda_\omega \leq \mu$, CEM with suboptimal policy updates will converge to a $\hat{\omega}$ that satisfies

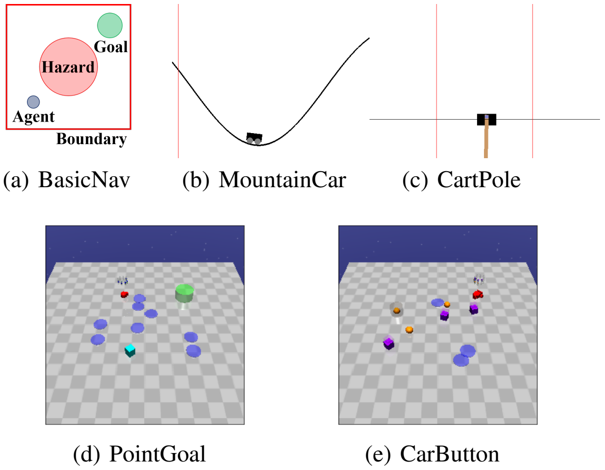
$$\min_{\omega'\in\Omega^*}\|\hat{\omega} - \omega'\| \leq \psi\sqrt{\epsilon/\mu}$$

with constant $\psi > 0$. With another constant $\xi > 0$, $\mathcal{F}(\hat{\omega})$ also converge to a bounded neighborhood of its optimal value:

$$\min_{\omega'\in\Omega^*}\|\mathcal{F}(\hat{\omega}) - \mathcal{F}(\omega')\| \leq \xi\epsilon/\mu^2.$$

Empirical Analysis

We evaluate our method based on a wide variety of TASE benchmarks. We organize our empirical analysis as follows: 1) We demonstrate that CEM can facilitate learning a safe exploration policy in various complex environments; 2) We reveal that the safe exploration policy can benefit the target tasks in safety and sample efficiency.



(a) BasicNav (b) MountainCar (c) CartPole

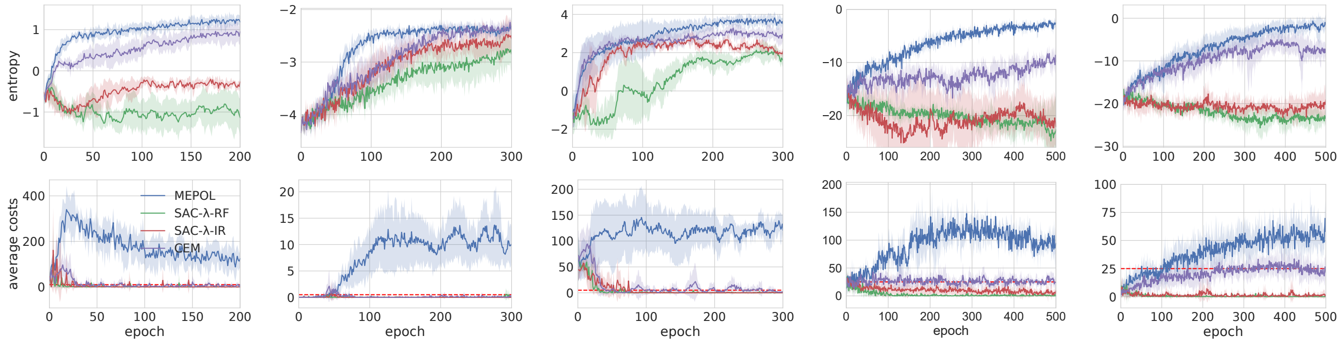
(d) PointGoal (e) CarButton

Figure 1: Safety-constrained exploration tasks with different complexity levels, i.e., the state spaces, the type of the obstacles, and the potential target tasks.

Benchmarks We first evaluate our safe unsupervised exploration in a 2D navigation domain BasicNav (2D states, Figure 1(a)), where a hazard in the center should be avoided. Then, we consider two continuous illustrative domains: MountainCar (2D, Figure 1(b)) and CartPole (4D, Figure 1(c)). Note that they are different from the original versions in OpenAI Gym (Brockman et al. 2016) because of the additional constraints. In MountainCar, the constraint is to not go too far to the left (indicated by the red line in Figure 1(b)), every step the cart is too far to the left a cost of 1 is incurred. In CartPole, the constraint is to keep the cart in a certain region. Finally, we test our method in a set of continuous control, high-dimensional environments from the Safety Gym suite (Todorov, Erez, and Tassa 2012; Ray, Achiam, and Amodei 2019): PointGoal (36D, Figure 1(d)), CarButton (56D, Figure 1(e)). In PointGoal, we control the point robot to navigate in the 2D map to reach a goal while trying to avoid a vase and several hazards. In CarButton, we control a more complex car robot to push the right button while trying to avoid the wrong button, several moving gremlins, and several fixed hazards. In all environments, $c = 1$ if an unsafe interaction happens, and $c = 0$ otherwise. All experiments are performed over 10 runs with different random seeds and the plots show the mean and standard deviation of all runs. More details about the environments and experiments are provided in Appendix B.

Evaluation of Safe Exploration

During training, the agent is not aware of the extrinsic environment reward, i.e., $r(s, a) = 0, \forall s \in \mathcal{S}, a \in \mathcal{A}$. The policy is evaluated in terms of its entropy value $\hat{\mathcal{H}}_k(\rho_T(\theta))$ and the average episodic costs over each epoch. We hand-



(a) BasicNav (b) MountainCar (c) CartPole (d) PointGoal (e) CarButton

Figure 2: Comparison of MEPOL, SAC-λ-RF, SAC-λ-IR, and CEM during training in exploration (top row) and safety (bottom row). The solid lines are the average of all runs, and the shaded area is the standard deviation. The red dashed lines indicate the safety thresholds.

tune hyperparameter k to attain reasonable performance of the entropy estimator. We choose the horizon T according to the potential task for the agent in each specific environment. We analyze the sensitivity of the parameters in Appendix C. To evaluate how our method performs in pure safe exploration tasks, we compare CEM with three baselines:

MEPOL To show how well the agent can explore the world without taking into account any safety concerns, we also take MEPOL as a baseline, which is a state-of-the-art algorithm in maximizing the state entropy (Mutti, Pratissoli, and Restelli 2021).

SAC-λ-RF To efficiently explore the world, we first consider SAC-λ (Ha et al. 2020) to maximize the policy entropy under the safety constraints with $r(s, a) = 0 : \forall s \in \mathcal{S}, a \in \mathcal{A}$, rather than optimize the state entropy directly.

SAC-λ-IR Inspired by the off-policy version for efficient exploration in the work by Seo et al. (2021), we introduce an auxiliary reward $r(s) := \log(\|s - s^{k\text{-NN}}\|_2 + 1)$ to further enhance the exploration under the framework of SAC-λ (Ha et al. 2020).

As we show in Figure 2, compared to the safe methods (SAC-λ-RF, SAC-λ-IR, and CEM), MEPOL shows the ability to acquire policies with remarkably strong exploration across all domains, but it does not satisfy the safety constraint. Both SAC-λ-RF and SAC-λ-IR can converge to safe policies even in more complex environments. In the classic control environments MountainCar and CartPole, the two SAC-λ methods can also make prominent improvements in state entropy, but failed in BasicNav (Figure 2(a)), PointGoal (Figure 2(d)), and CarButton (Figure 2(e)). In general, with the benefits from the intrinsic reward, SAC-λ-IR attained higher state entropy than SAC-λ-RF. Compared to all the baselines, only CEM managed to learn a policy that finally gets remarkable results in exploration and satisfies the safety constraint.

After training, we leverage the heat maps in Figure 3 to show the exploration of the final policies in the illustrative environments BasicNav, MountainCar, and CartPole. Note that the states in CartPole are 4D, but we just focus on the cart position and pole angle. We can see that MEPOL can always achieve efficient exploration in all environments. However, the unsafe areas are also covered by the learned agents. The exploration heat maps also show that the two SAC-λ methods are too conservative in safety. Even though SAC-λ-IR is generally better than SAC-λ-RF, the learned agent cannot cover the safe areas well, especially in BasicNav and CartPole. Only CEM can efficiently explore the safe areas in all the illustrative environments.

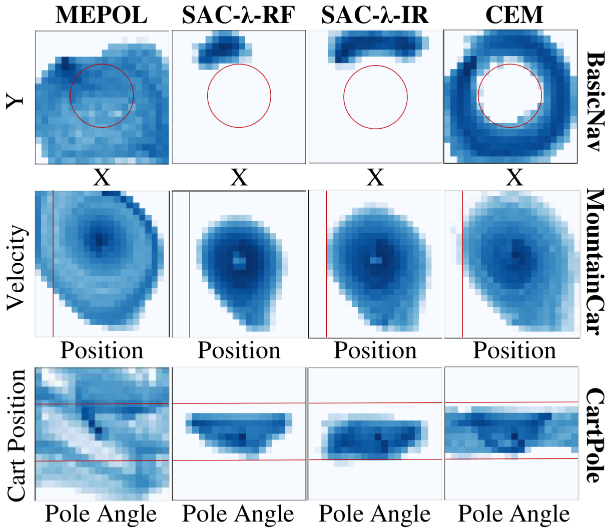
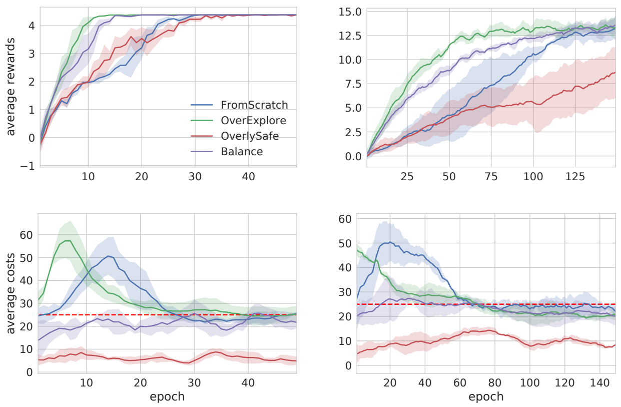


Figure 3: Exploration analysis after training. The heat maps show the final state density of the learned policies. The red line indicates the dangerous area.

Evaluation of Safe Transfer Learning

In this section, we evaluate how a safe exploration policy learned by CEM can benefit the target tasks in safety and sample efficiency. To evaluate the policy in safety, we use the safety costs generated during the interaction with the



(a) PointGoal (b) CarButton

Figure 4: Effects of the quality of the safe exploration policies on the target tasks. The solid lines are the average of all runs, and the shaded area is the standard deviation. The red dashed lines indicate the safety thresholds.

environment. In terms of performance, we use the average episodic rewards over 100 episodes in an extra test process after each epoch. In the target tasks, the extrinsic environment reward is revealed to the agent. We leverage the safe exploration policy to guide learning in the off-policy safe guide (SaGui; Yang et al. 2022a) framework, which achieves safe transfer learning by two mechanisms: *i*) Adaptively regularize the student policy to the guide policy based on the student's safety; *ii*) Use the safe exploration policy as a recovery policy when the student starts to take unsafe actions.

To show how the quality of the safe exploration policies plays a role in learning, we use the policy learned by CEM to represent a teacher with a good balance between safety and exploration (Balance). For comparison, we use the policy learned by MEPOL to represent an unsafe teacher but with efficient exploration over the whole state space (OverExplore), so we deactivate the recovery mechanism with this unsafe policy in SaGui for a fair comparison. On the other hand, we also use the policy learned SAC-λ-RF, which is safe but very conservative in exploration (OverlySafe). We also take the agent that starts learning from scratch (FromScratch) as a baseline.

In Figure 4, we show how the quality of the safe exploration policies influences the learning in the target tasks, where the agent needs to reach the goal in PointGoal, and push the right button in CarButton. In general, we can observe that the different safe exploration policies benefit the target tasks in different ways. The agent guided by the OverExplore policy can learn to get high rewards quickly, but cannot get obvious improvement in safety compared to learning from scratch. The OverlySafe policy can stay the agent to be absolutely safe when interacting with the environment. However, the resulting performance is even worse than learning from scratch. The policy learned by CEM (Balance) can guide the agent to obtain high rewards quickly under the condition of ensuring the safety of training.

Related Work

Task-agnostic exploration has been studied in three different directions, estimating the environment dynamics (Jin et al. 2020; Tarbouriech et al. 2020), learning a transferable meta-reward function (Bechtle et al. 2021; Zheng et al. 2020), and learning an efficient exploration policy (Hazan et al. 2019; Lee et al. 2019; Tarbouriech and Lazaric 2019; Mutti and Restelli 2020; Mutti, Pratissoli, and Restelli 2021; Guo et al. 2021; Nedergaard and Cook 2022). These works made impressive progress in exploring the environment efficiently without a reward signal (Laskin et al. 2021b). Nevertheless, task-agnostic exploration with safety concerns is still underexplored. Compared to our method, their learned policies cannot explore safely, which is important when we need to explore the real world and the target tasks are safety critical.

The constrained cross-entropy method proposed by Wen and Topcu (2018) could be extended to the TASE problem, but its efficiency under the state-entropy maximization objective has not yet been tested. To some extent, SAC-λ can also be used to solve our problem. By maximizing the policy entropy, the agent trained by SAC-λ tends to have diverse behaviors, but it does not imply efficient exploration of the environment. With an additional intrinsic reward, the exploration of SAC-λ can be enhanced (Yang et al. 2022a), but the interpretability of the learned policy in exploration is not clear. Achiam et al. (2017); Liu, Ding, and Liu (2020); Yang et al. (2020) propose a series of constrained policy optimization methods, where the constraints are built on long-term costs instead of real costs within a finite horizon. To apply their methods in our domain, more work is needed to process the different optimization objective and constraint.

Conclusions and Future Work

In this paper, we propose the CEM algorithm to solve safety-constrained entropy maximization problems in a completely reward-free manner. We argue that it is more practical to formulate the problem to be finite-horizon without discounting, which mitigates the problem of designing a safety threshold based on the discounted cost-return. To trade off exploration with safety, we adaptively change the safety weights based on the undiscounted real costs. Accordingly, we can update the policy under the adjusted balance between safety and exploration. The learned policy can maximize exploration under the premise of safety even in complex continuous-control domains, and benefit the potential target tasks in sample efficiency and safety. To address more complex problems in the future, it is critical to abstract the state space to make the state entropy estimation easier and more effective (Tao, François-Lavet, and Pineau 2020; Liu and Abbeel 2021b; Seo et al. 2021; Yarats et al. 2021). Also, it is promising to consider different pretraining settings, e.g., maximum entropy over the state-action pairs (Zhang, Cai, and Li 2021), the maximum mutual information between tasks and policy-induced states (Liu and Abbeel 2021a) or between state transitions and latent skill vectors (Laskin et al. 2021a), pretraining in a class of multiple environments (Mutti, Mancassola, and Restelli 2022), and pretraining for history-based policies (Mutti, De Santi, and Restelli 2021).

Acknowledgments

We would like to thank Dr. Simon H. Tindemans for discussing the presented idea, and Dr. Thiago Dias Simão for his valuable feedback. Qisong Yang is supported by Xidian University.

References

Achiam, J.; Held, D.; Tamar, A.; and Abbeel, P. 2017. Constrained Policy Optimization. In *Proceedings of the 34th International Conference on Machine Learning*, 22–31. PMLR.

Ajgl, J.; and Šimandl, M. 2011. Particle based probability density fusion with differential Shannon entropy criterion. In *14th International Conference on Information Fusion*, 1–8. IEEE.

Altman, E. 1999. *Constrained Markov decision processes*, volume 7. CRC Press.

Badia, A. P.; Sprechmann, P.; Vitvitskyi, A.; Guo, D.; Piot, B.; Kapturowski, S.; Tieleman, O.; Arjovsky, M.; Pritzel, A.; Bolt, A.; et al. 2019. Never Give Up: Learning Directed Exploration Strategies. In *International Conference on Learning Representations*.

Bechtle, S.; Molchanov, A.; Chebotar, Y.; Grefenstette, E.; Righetti, L.; Sukhatme, G.; and Meier, F. 2021. Meta learning via learned loss. In *2020 25th International Conference on Pattern Recognition (ICPR)*, 4161–4168. IEEE.

Bellemare, M.; Srinivasan, S.; Ostrovski, G.; Schaul, T.; Saxton, D.; and Munos, R. 2016. Unifying count-based exploration and intrinsic motivation. *Advances in Neural Information Processing Systems*, 29.

Borkar, V. S. 2005. An actor-critic algorithm for constrained Markov decision processes. *Systems & control letters*, 54(3): 207–213.

Boutilier, C.; and Lu, T. 2016. Budget allocation using weakly coupled, constrained Markov decision processes. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, 52–61.

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Openai gym. *arXiv preprint arXiv:1606.01540*.

Burda, Y.; Edwards, H.; Pathak, D.; Storkey, A.; Darrell, T.; and Efros, A. A. 2019a. Large-Scale Study of Curiosity-Driven Learning. In *International Conference on Learning Representations*.

Burda, Y.; Edwards, H.; Storkey, A.; and Klimov, O. 2019b. Exploration by random network distillation. In *International Conference on Learning Representations*.

De Nijs, F.; Spaan, M. T. J.; and de Weerdt, M. 2015. Best-response planning of thermostatically controlled loads under power constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Dulac-Arnold, G.; Levine, N.; Mankowitz, D. J.; Li, J.; Paduraru, C.; Gowal, S.; and Hester, T. 2021. Challenges of real-world reinforcement learning: definitions, benchmarks and analysis. *Machine Learning*, 110(9): 2419–2468.

Fox, L.; Choshen, L.; and Loewenstein, Y. 2018. DORA The Explorer: Directed Outreaching Reinforcement Action-Selection. In *International Conference on Learning Representations*.

García, J.; and Fernández, F. 2015. A Comprehensive Survey on Safe Reinforcement Learning. *The Journal of Machine Learning Research*, 16(1): 1437–1480.

Gordon, G.; and Tibshirani, R. 2012. Karush-Kuhn-Tucker conditions. *Optimization*, 10(725/36): 725.

Guo, Z. D.; Azar, M. G.; Saade, A.; Thakoor, S.; Piot, B.; Pires, B. A.; Valko, M.; Mesnard, T.; Lattimore, T.; and Munos, R. 2021. Geometric entropic exploration. *arXiv preprint arXiv:2101.02055*.

Ha, S.; Xu, P.; Tan, Z.; Levine, S.; and Tan, J. 2020. Learning to Walk in the Real World with Minimal Human Effort. In *Proceedings of the 2020 Conference on Robot Learning*, 1110–1120. PMLR.

Haarnoja, T.; Zhou, A.; Abbeel, P.; and Levine, S. 2018a. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In *Proceedings of the 35th International Conference on Machine Learning*, 1861–1870. PMLR.

Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; and Levine, S. 2018b. Soft Actor-Critic Algorithms and Applications. *arXiv preprint arXiv:1812.05905*.

Hazan, E.; Kakade, S.; Singh, K.; and Van Soest, A. 2019. Provably Efficient Maximum Entropy Exploration. In *Proceedings of the 36th International Conference on Machine Learning*, 2681–2691. PMLR.

Jardine, A. K.; Lin, D.; and Banjevic, D. 2006. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, 20(7): 1483–1510.

Jin, C.; Krishnamurthy, A.; Simchowitz, M.; and Yu, T. 2020. Reward-free exploration for reinforcement learning. In *International Conference on Machine Learning*, 4870–4879. PMLR.

Kakade, S. M. 2001. A natural policy gradient. *Advances in Neural Information Processing Systems*, 14.

Kamran, D.; Simão, T. D.; Yang, Q.; Ponnambalam, C. T.; Fischer, J.; Spaan, M. T. J.; and Lauer, M. 2022. A Modern Perspective on Safe Automated Driving for Different Traffic Dynamics using Constrained Reinforcement Learning. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, 4017–4023. IEEE.

Laskin, M.; Liu, H.; Peng, X. B.; Yarats, D.; Rajeswaran, A.; and Abbeel, P. 2021a. CIC: Contrastive Intrinsic Control for Unsupervised Skill Discovery. In *Deep RL Workshop NeurIPS 2021*.

Laskin, M.; Yarats, D.; Liu, H.; Lee, K.; Zhan, A.; Lu, K.; Cang, C.; Pinto, L.; and Abbeel, P. 2021b. URLB: Unsupervised Reinforcement Learning Benchmark. In *Deep RL Workshop NeurIPS 2021*.

Lee, L.; Eysenbach, B.; Parisotto, E.; Xing, E.; Levine, S.; and Salakhutdinov, R. 2019. Efficient Exploration via State Marginal Matching. *arXiv preprint arXiv:1906.05274*.

Liu, H.; and Abbeel, P. 2021a. Aps: Active pretraining with successor features. In *International Conference on Machine Learning*, 6736–6747. PMLR.

Liu, H.; and Abbeel, P. 2021b. Behavior from the void: Unsupervised active pre-training. *Advances in Neural Information Processing Systems*, 34: 18459–18473.

Liu, Y.; Ding, J.; and Liu, X. 2020. IPO: Interior-point policy optimization under constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 4940–4947.

Marot, A.; Donnot, B.; Romero, C.; Donon, B.; Lerousseau, M.; Veyrin-Forrer, L.; and Guyon, I. 2020. Learning to run a power network challenge for training topology controllers. *Electric Power Systems Research*, 189: 106635.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M. A.; Fidjeland, A.; Ostrovski, G.; Petersen, S.; Beattie, C.; Sadik, A.; Antonoglou, I.; King, H.; Kumaran, D.; Wierstra, D.; Legg, S.; and Hassabis, D.

2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533.

Mutti, M.; De Santi, R.; and Restelli, M. 2021. The Importance of Non-Markovianity in Maximum State Entropy Exploration. In *ICML 2021 Workshop on Unsupervised Reinforcement Learning*.

Mutti, M.; Mancassola, M.; and Restelli, M. 2022. Unsupervised reinforcement learning in multiple environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 7850–7858.

Mutti, M.; Pratissoli, L.; and Restelli, M. 2021. Task-agnostic exploration via policy gradient of a non-parametric state entropy estimate. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 9028–9036.

Mutti, M.; and Restelli, M. 2020. An intrinsically-motivated approach for learning highly exploring and fast mixing policies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 5232–5239.

Nedergaard, A.; and Cook, M. 2022. k-Means Maximum Entropy Exploration. *arXiv preprint arXiv:2205.15623*.

Ostrovski, G.; Bellemare, M. G.; Oord, A.; and Munos, R. 2017. Count-based exploration with neural density models. In *International conference on machine learning*, 2721–2730. PMLR.

Pathak, D.; Agrawal, P.; Efros, A. A.; and Darrell, T. 2017. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, 2778–2787. PMLR.

Pathak, D.; Gandhi, D.; and Gupta, A. 2019. Self-supervised exploration via disagreement. In *International conference on machine learning*, 5062–5071. PMLR.

Peters, J.; and Bagnell, J. A. 2010. Policy Gradient Methods. *Scholarpedia*, 5(11): 3698.

Puterman, M. L. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Qin, Z.; Chen, Y.; and Fan, C. 2021. Density Constrained Reinforcement Learning. In *Proceedings of the 38th International Conference on Machine Learning*, 8682–8692. PMLR.

Ray, A.; Achiam, J.; and Amodei, D. 2019. Benchmarking Safe Exploration in Deep Reinforcement Learning. https://cdn.openai.com/safexp-short.pdf. Accessed: 2019-11-21.

Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M.; and Moritz, P. 2015. Trust Region Policy Optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, 1889–1897. JMLR.org.

Seo, Y.; Chen, L.; Shin, J.; Lee, H.; Abbeel, P.; and Lee, K. 2021. State entropy maximization with random encoders for efficient exploration. In *International Conference on Machine Learning*, 9443–9454. PMLR.

Singh, H.; Misra, N.; Hnizdo, V.; Fedorowicz, A.; and Demchuk, E. 2003. Nearest neighbor estimates of entropy. *American journal of mathematical and management sciences*, 23(3-4): 301–321.

Stadie, B. C.; Levine, S.; and Abbeel, P. 2015. Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814*.

Subramanian, M.; Viebahn, J.; Tindemans, S. H.; Donnot, B.; and Marot, A. 2021. Exploring grid topology reconfiguration using a simple deep reinforcement learning approach. In *2021 IEEE Madrid PowerTech*, 1–6. IEEE.

Sun, Y.; Duan, Y.; Gong, H.; and Wang, M. 2019. Learning low-dimensional state embeddings and metastable clusters from time series data. *Advances in Neural Information Processing Systems*, 32.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement Learning: An Introduction*, volume 2. MIT press.

Tang, H.; Houthooft, R.; Foote, D.; Stooke, A.; Xi Chen, O.; Duan, Y.; Schulman, J.; DeTurck, F.; and Abbeel, P. 2017. # exploration: A study of count-based exploration for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 30.

Tao, R. Y.; François-Lavet, V.; and Pineau, J. 2020. Novelty search in representational space for sample efficient exploration. *Advances in Neural Information Processing Systems*, 33.

Tarbouriech, J.; and Lazaric, A. 2019. Active exploration in markov decision processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 974–982. PMLR.

Tarbouriech, J.; Shekhar, S.; Pirotta, M.; Ghavamzadeh, M.; and Lazaric, A. 2020. Active model estimation in markov decision processes. In *Conference on Uncertainty in Artificial Intelligence*, 1019–1028. PMLR.

Todorov, E.; Erez, T.; and Tassa, Y. 2012. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5026–5033.

Walraven, E.; and Spaan, M. T. J. 2018. Column generation algorithms for constrained POMDPs. *Journal of Artificial Intelligence Research*, 62: 489–533.

Wen, M.; and Topcu, U. 2018. Constrained cross-entropy method for safe reinforcement learning. *Advances in Neural Information Processing Systems*, 31.

Yang, Q.; Simão, T. D.; Jansen, N.; Tindemans, S. H.; and Spaan, M. T. J. 2022a. Training and transferring safe policies in reinforcement learning. In *AAMAS 2022 Workshop on Adaptive Learning Agents*.

Yang, Q.; Simão, T. D.; Tindemans, S. H.; and Spaan, M. T. J. 2021. WCSAC: Worst-Case Soft Actor Critic for Safety-Constrained Reinforcement Learning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*, 10639–10646. AAAI Press.

Yang, Q.; Simão, T. D.; Tindemans, S. H.; and Spaan, M. T. J. 2022b. Safety-constrained reinforcement learning with a distributional safety critic. *Machine Learning*, 1–29.

Yang, T.; Rosca, J.; Narasimhan, K.; and Ramadge, P. J. 2020. Projection-Based Constrained Policy Optimization. In *8th International Conference on Learning Representations*, 1–10. OpenReview.net.

Yarats, D.; Fergus, R.; Lazaric, A.; and Pinto, L. 2021. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, 11920–11931. PMLR.

Zhang, C.; Cai, Y.; and Li, L. H. J. 2021. Exploration by Maximizing Rényi Entropy for Reward-Free RL Framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 10859–10867.

Zheng, Z.; Oh, J.; Hessel, M.; Xu, Z.; Kroiss, M.; Van Hasselt, H.; Silver, D.; and Singh, S. 2020. What can learned intrinsic rewards capture? In *International Conference on Machine Learning*, 11436–11446. PMLR.