Safe Reinforcement Learning via Confidence-Based Filters

Sebastian Curi[1*], Armin Lederer[2*], Sandra Hirche[2], Andreas Krause[1]

Abstract— **Ensuring safety is a crucial challenge when deploying reinforcement learning (RL) to real-world systems. We develop *confidence-based safety filters*, a control-theoretic approach for certifying state safety constraints for nominal policies learnt via standard RL techniques, based on probabilistic dynamics models. Our approach is based on a reformulation of state constraints in terms of cost functions, reducing safety verification to a standard RL task. By exploiting the concept of *hallucinating inputs*, we extend this formulation to determine a "backup" policy which is safe for the unknown system with high probability. The nominal policy is minimally adjusted at every time step during a roll-out towards the backup policy, such that safe recovery can be guaranteed afterwards. We provide formal safety guarantees, and empirically demonstrate the effectiveness of our approach.**

I. INTRODUCTION

When agents operate autonomously in unknown environments, they need the ability to adapt to new situations. This adaptiveness can be achieved using reinforcement learning [1], which allows autonomous agents to modify their behavior according to observations of the environment. Such reinforcement learning approaches have been demonstrated to achieve state-of-the-art performance on various problems where high-fidelity simulations are available [2], but they cannot be directly applied to real-world autonomous systems because their safe operation must always be guaranteed. This has lead to safety being a major hurdle for the application of reinforcement learning in real-world applications [3].

Related work: Due to this high relevance of safety in reinforcement learning, it has been the focus of a variety of recent approaches (see [4], [5] for surveys). A common framework for safe reinforcement learning are constrained Markov decision processes (CMDP) [6]. In the CMDP setting, constraints are posed on the expected cumulative cost along roll-outs of a policy. This allows to treat the cumulative cost analogously to rewards, such that methods such as trust region policy optimization can be adapted to maintain constraint satisfaction when initialized with a safe policy [7]. When no initially safe policy is known, a Lagrangian relaxation can be used to asymptotically find safe policies [8]. This dual representation of the constrained optimization problem can also be combined with techniques such as upper confidence reinforcement learning to guarantee learning rates, e.g., for linear CMDPs [9]. However, constraint violation during training cannot be excluded in general, which often prevents the usage in real-world applications.

Control theoretic methods consider safety through constraints on the system states, but often severely restrict the allowed policy and system classes. For example, linear quadratic regulators can be learned efficiently under polytopic constraints on the system states as shown by [10]. The limitation to linear dynamics can be relaxed in the case of deterministic systems by employing model predictive control (MPC) techniques, such that the performance can be iteratively improved [11]. To allow active exploration, MPC can be combined with reinforcement learning ideas by ensuring safety using a suitable MPC parameterization [12]. Moreover, Monte-Carlo approximations can be used to deal with the stochastic dynamics commonly found in reinforcement learning problems [13], although this comes at the price of a prohibitive computational complexity.

To achieve the beneficial properties of both control theoretical and reinforcement learning based approaches, it has recently been proposed to employ reinforcement learning for finding the optimal policy, while in a secondary step a control method is used to certify safety and, if necessary, adapt the applied action [14]. This approach can be realized using control barrier functions [15], or a "backup" policy, which is locally safe in some region of the state space [16]. While designing control barrier functions is challenging in general, determining locally safe policies often requires solving computationally expensive optimization problems on-line [16] or can only be applied to linearized systems [17]. Therefore, the practical applicability of such safety filters in combination with highly flexible RL techniques is currently limited.

Our contributions: We mitigate these weaknesses by proposing *hallucinating upper confidence safety filters* for ensuring safety of arbitrary policies applied to stochastic, nonlinear systems for which merely a model with high probability error bounds is known. To this end, we first establish a *relationship between state constraints and level sets of value functions*. Using the concept of hallucinating inputs [18], we show that these value functions can be efficiently estimated with standard reinforcement learning methods. Our approach can be naturally extended to *finding safe policies*, by formulating it as a robust reinforcement learning problem. These safe policies can then be used for *computationally efficient*

This work was supported by the European Research Council (ERC) Consolidator Grant "Safe data-driven control for human-centric systems (CO-MAN)" under grant agreement No 864686 and under the European Unions Horizon 2020 research and Innovation Program Grant "Reliable Data-Driven Decision Making in Cyber-Physical Systems (RADDICS)" agreement No 815943. It also received support from the Swiss National Science Foundation under NCCR Automation, grant agreement 51NF40 180545. A. L. gratefully acknowledges financial support from the German Academic Scholarship Foundation. Published under © IEEE 2022.

[1] LAS Group, Department of Computer Science, ETH Zurich, Switzerland. Email: [scuri,krausea]@inf.ethz.ch

[2] Chair of Information-oriented Control, Department of Electrical and Computer Engineering, Technical University of Munich, Germany. Email: [armin.lederer, hirche]@tum.de

*These authors contributed equally.

on-line safety adaptation of arbitrary reinforcement learning policies. We demonstrate the effectiveness of the proposed method on deep RL benchmark tasks.

The remainder of this paper is structured as follows. In Section II, we formalize the problem setting. The hallucinating upper confidence filter is explained and shown to yield safe policies in Section IV. Finally, the performance of the safety filter is demonstrated in simulations in Section V, before the paper is concluded in Section VI.

II. PROBLEM STATEMENT AND BACKGROUND

We consider a discrete-time dynamical system

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}^{\star}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{\omega}_k, \tag{1}$$

where $\boldsymbol{x}_k \in \mathcal{X} \subset \mathbb{R}^{d_x}$ are states, $\boldsymbol{u}_k \in \mathcal{U} \subset \mathbb{R}^{d_u}$ control actions, $\boldsymbol{\omega}_k \sim \rho$ is process noise sampled from a zero-mean probability distribution ρ[1], and $\boldsymbol{f}^{\star} : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ denotes the unknown deterministic transition function. The control actions $\boldsymbol{u}_k$ are determined using a policy $\boldsymbol{\pi} \colon \mathcal{X} \to \mathcal{U}$, with the goal to maximize an expected cumulative return

$$\begin{aligned} R(\boldsymbol{f}^{\star}, \boldsymbol{\pi}; \boldsymbol{x}) = \mathbf{E}_{\boldsymbol{\omega}}\left[\sum_{k=0}^{\infty} \gamma^k r(\boldsymbol{x}_k, \boldsymbol{\pi}(\boldsymbol{x}_k))\right], && \text{(2a)} \\ \text{s.t. } \boldsymbol{x}_{k+1} = \boldsymbol{f}^{\star}(\boldsymbol{x}_k, \boldsymbol{\pi}(\boldsymbol{x}_k)) + \boldsymbol{\omega}_k,\ \boldsymbol{\omega}_k \sim \rho, && \text{(2b)} \\ \boldsymbol{x}_0 = \boldsymbol{x}, && \text{(2c)} \end{aligned}$$

where $r : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ is a known immediate reward function and $\gamma \in (0, 1)$ is a discount factor.

In practice, the policy $\boldsymbol{\pi}$ must additionally ensure safety of the closed-loop dynamical system, e.g., because damage to the system described by $\boldsymbol{f}^{\star}$ must be avoided. In the RL literature, this is typically addressed through constrained Markov decision processes, which additionally consider a constraint on a cumulative cost function

$$C(\boldsymbol{f}^{\star}, \boldsymbol{\pi}; \boldsymbol{x}) = \mathbf{E}_{\boldsymbol{\omega}}\left[\sum_{k=0}^{\infty} \gamma^k c(\boldsymbol{x}_k)\right] < \xi, \tag{3}$$

where $c : \mathcal{X} \to \mathbb{R}$ is an immediate cost, $\xi \in \mathbb{R}$ is a constant specifying the constraint, and $\boldsymbol{x}_k$ is defined iteratively through (1) with actions $\boldsymbol{u}_k = \boldsymbol{\pi}(\boldsymbol{x}_k)$ and initial state $\boldsymbol{x}_0 = \boldsymbol{x}$. Therefore, an optimization problem of the form

$$\begin{aligned} \boldsymbol{\pi}^{*} = \arg\max_{\boldsymbol{\pi}} R(\boldsymbol{f}^{\star}, \boldsymbol{\pi}; \boldsymbol{x}) && \text{(4a)} \\ \text{s.t. } C(\boldsymbol{f}^{\star}, \boldsymbol{\pi}; \boldsymbol{x}) < \xi && \text{(4b)} \end{aligned}$$

is usually solved to determine safe policies.

While this problem can be directly solved by adapting standard RL algorithms with techniques akin to Lagrangian relaxation [8], this approach generally cannot ensure safety *during* training. Moreover, it does not reflect the fact that the safety of many systems is defined in terms of a set of safe states $\mathcal{X}_{\text{safe}} \subset \mathcal{X}$ and its complement $\mathcal{X}_{\text{unsafe}} = \mathcal{X} \setminus \mathcal{X}_{\text{safe}}$. For example, an autonomously driving car should not leave the road, which directly defines the road as $\mathcal{X}_{\text{safe}}$. When using the natural indicator $\mathbf{1}_{\boldsymbol{x} \in \mathcal{X}_{\text{unsafe}}}$ as cost function, satisfying Equation (4b) bounds the discounted probability of violating the constraints by ξ. Nonetheless, this does not guarantee that constraints will not be violated when deploying $\boldsymbol{\pi}^{*}$.

Therefore, we consider safety in terms of state constraints $\boldsymbol{x}_k \in \mathcal{X}_{\text{safe}}$, which we require to hold with high probability, since the process noise $\boldsymbol{\omega}$ generally prevents deterministic guarantees. This leads to the following definition of safety.

Definition 2.1: A policy $\boldsymbol{\pi}$ is K-step δ-safe for a state $\boldsymbol{x} \in \mathcal{X}$ if it holds that $\mathbb{P}\left(\boldsymbol{x}_k \in \mathcal{X}_{\text{safe}}\ \forall k = 0, \ldots, K | \boldsymbol{x}_0 = \boldsymbol{x}\right) \geq 1 - \delta$, where states $\boldsymbol{x}_k$ are defined in (1).

The concept of K-step δ-safety is commonly found in stochastic model predictive control, where it is typically referred to as joint chance constraint [19].

Remark 2.1: We consider finite values of K because ensuring δ-safety over an infinite horizon, i.e., $K = \infty$, is not possible for unbounded process noise $\boldsymbol{\omega}$ in general. This can be easily seen for a system with $\boldsymbol{f}^{\star} = \mathbf{0}$ and i.i.d. zero mean Gaussian noise $\boldsymbol{\omega}$, which almost surely leaves any compact safe set $\mathcal{X}_{\text{safe}}$ eventually.

In order to obtain the optimal policy $\boldsymbol{\pi}^{*}_{\text{safe}}$ ensuring δ-safety, we generally need to consider the optimization problem

$$\begin{aligned} \boldsymbol{\pi}^{*}_{\text{safe}} = \arg\max_{\boldsymbol{\pi}}\ R(\boldsymbol{f}^{\star}, \boldsymbol{\pi}; \boldsymbol{x}) && \text{(5a)} \\ \text{s.t. } \mathbb{P}\left(\boldsymbol{x}_k \in \mathcal{X}_{\text{safe}}\ \forall k = 0, \ldots, K | \boldsymbol{x}_0 = \boldsymbol{x}\right) \geq 1 - \delta. && \text{(5b)} \end{aligned}$$

Solving this optimization problem is challenging since there usually exists no closed-form expression for the probability (5b), such that computationally expensive uncertainty propagation methods have to be employed, e.g., generalized polynomial chaos expansions [20].

In order to efficiently determine approximate solutions for (5), we follow the idea of [17] and separate it into two phases: an initial phase for determining a nominal policy $\boldsymbol{\pi}^{*}$ using an arbitrary method, followed by an on-line phase in which a safety filter is employed to adapt the policy $\boldsymbol{\pi}^{*}$ to ensure K-step δ-safety. Since we cannot ensure safety without any knowledge about $\boldsymbol{f}^{\star}$, we assume to have access to a set of *plausible* models $\mathcal{M} = \{\boldsymbol{f} \mid |\boldsymbol{f} - \boldsymbol{\mu}| \leq \beta \boldsymbol{\sigma}\}$ described by a nominal model $\boldsymbol{\mu} \colon \mathcal{X} \times \mathcal{U} \to \mathbb{R}^{d_x}$, the state-action dependent uncertainty about the model $\boldsymbol{\sigma} : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^{d_x}$, and a constant scaling factor $\beta \in \mathbb{R}_{+}$. We assume that this set of models is *well-calibrated*, i.e., $\boldsymbol{f}^{\star} \in \mathcal{M}$ with high probability, as formalized in the following.

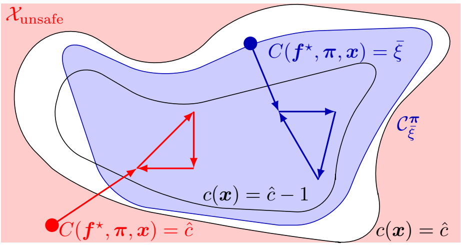
Assumption 2.1 ([18]): The statistical model is calibrated with respect to $\boldsymbol{f}^{\star}$, i.e., there exists a $\beta \in \mathbb{R}_{+}$ such that, with probability at least $1 - \delta_f$, it holds jointly for all $\boldsymbol{x}, \boldsymbol{u} \in \mathcal{X} \times \mathcal{U}$ that $|\boldsymbol{f}^{\star}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u})| \leq \beta \boldsymbol{\sigma}(\boldsymbol{x}, \boldsymbol{u})$, element-wise.

Since the statistical model is often obtained by applying supervised machine learning to data obtained from policy roll-outs [18], the uncertainty usually decreases with the number of roll-outs. Thereby, this assumption typically enables less conservative and higher performant policies over time.

Using Assumption 2.1, we investigate the following two sub-problems for the derivation of the safety filter.

a) State Constraints as Cumulative Cost: In order to enable the application of reinforcement learning methods, we consider the problem of converting the K-step δ-safety constraint (5b) for known dynamics $\boldsymbol{f}^{\star}$ into a constraint on

[1] We consider constant noise distributions ρ for notational simplicity, but our approach directly extends to state and action dependent distributions.



$\mathcal{X}_{\text{unsafe}}$
$C(\boldsymbol{f}^\star,\boldsymbol{\pi},\boldsymbol{x})=\bar{\xi}$
$\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$
$c(\boldsymbol{x})=\hat{c}-1$
$C(\boldsymbol{f}^\star,\boldsymbol{\pi},\boldsymbol{x})=\hat{c}$
$c(\boldsymbol{x})=\hat{c}$

Fig. 1. The expected cumulative cost can be 0 even if the immediate cost c at the first state is greater than 0, as this positive cost can be compensated by negative costs afterwards (red trajectory). Therefore, $\mathcal{C}^{\boldsymbol{\pi}}_{\hat{c}} \not\subset \mathcal{X}_{\text{safe}}$, such that we have to consider the tightened threshold $\bar{\xi}$, which ensures that states $\boldsymbol{x} \in \mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$ start with immediate cost $c(\boldsymbol{x}) \leq \hat{c}$ (blue trajectory).

an expected cumulative cost function. We show that this can be achieved by deriving a condition of the form

$$\mathbf{E}_{\boldsymbol{\omega}}[C(\boldsymbol{f}^\star,\boldsymbol{\pi};\boldsymbol{f}^\star(\boldsymbol{x},\boldsymbol{\pi})+\boldsymbol{\omega})]<\xi. \tag{6}$$

for suitably chosen immediate costs c, cf. Section III.

b) Safety Filter: Using this condition, we derive a novel approach for computing safe policies $\boldsymbol{\pi}_{\text{safe}}$ for systems with unknown dynamics $\boldsymbol{f}^\star$. This allows us to address the problem of ensuring the safety of a possibly unsafe nominal policy $\boldsymbol{\pi}^*$ on-line using a confidence-based filter

$$\hat{\boldsymbol{\pi}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{U}} \|\boldsymbol{\pi}^*(\boldsymbol{x})-\boldsymbol{u}\|, \tag{7a}$$

$$\text{s.t. } \max_{\boldsymbol{f}\in\mathcal{M}} \mathbf{E}_{\boldsymbol{\omega}}[C(\boldsymbol{f},\boldsymbol{\pi}_{\text{safe}};\boldsymbol{x}')] < \xi, \tag{7b}$$

$$\boldsymbol{x}' = \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})+\boldsymbol{\omega},\ \boldsymbol{\omega}\sim\rho, \tag{7c}$$

which outputs the closest safe action to $\boldsymbol{\pi}^*$. We derive tractable formulations for this filter in Section IV.

III. EXPRESSING STATE CONSTRAINTS THROUGH COST FUNCTIONS

To reformulate the δ-safety constraint into a constraint on cumulative costs, we first show in Section III-A that sub-level sets of C contained in $\mathcal{X}_{\text{safe}}$ can be easily defined. Based on this result, we derive sufficient conditions on the cost function, which allow to conclude safety from cumulative cost constraints in Section III-B, providing useful design freedom.

A. Safe Sub-Level Sets of the Cumulative Cost

For deriving the sub-level set $\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}} = \{\boldsymbol{x}\in\mathcal{X}: C(\boldsymbol{f}^\star,\boldsymbol{\pi};\boldsymbol{x})<\bar{\xi}\}$, $\bar{\xi}\in\mathbb{R}$, which is contained in the set of safe states $\mathcal{X}_{\text{safe}}$, we consider an immediate cost function $c:\mathcal{X}\to\mathbb{R}$ satisfying

$$\underline{c} \leq c(\boldsymbol{x}) \leq \bar{c} \quad \forall \boldsymbol{x}\in\mathcal{X}, \quad c(\boldsymbol{x}) \geq \hat{c} \quad \text{if } \boldsymbol{x}\in\mathcal{X}_{\text{unsafe}} \tag{8}$$

for constants $\underline{c},\bar{c},\hat{c}\in\mathbb{R}$. For example, using the indicator function $\mathbf{1}_{\boldsymbol{x}\in\mathcal{X}_{\text{unsafe}}}$ as cost, which equals 1 for $\boldsymbol{x}\in\mathcal{X}_{\text{unsafe}}$ and 0 otherwise, implies $\underline{c}=0$ and $\bar{c}=\hat{c}=1$. Using this definition, we can define an inner-approximation of the safe set of states $\mathcal{X}_{\text{safe}}$ through the $\hat{c}$ sub-level set of the immediate cost c, which becomes exact if $c(\boldsymbol{x})<\hat{c}$ for all $\boldsymbol{x}\in\mathcal{X}_{\text{safe}}$. Moreover, we can define the expected cumulative cost using (3).

While one might think that the definition of the immediate cost c in (8) ensures that the $\hat{c}$ sub-level set $\mathcal{C}^{\boldsymbol{\pi}}_{\hat{c}}$ of C is also contained in the safe set of states $\mathcal{X}_{\text{safe}}$, this is not true in general. As illustrated by the red trajectory in Fig. 1, the cumulative cost C can equal $\hat{c}$ even if the immediate cost c in the initial state is greater than $\hat{c}$, since negative costs of following states along the trajectory can compensate it. Therefore, the sub-level set $\mathcal{C}^{\boldsymbol{\pi}}_{\hat{c}}$ is generally not completely contained in the set of safe states $\mathcal{X}_{\text{safe}}$, such that we must consider a tightened threshold $\bar{\xi}$. Due to the lower bound $\underline{c}$ of the cost c, this constant $\bar{\xi}$ can be determined using the following lemma.

Lemma 3.1: Consider an immediate cost function $c\colon\mathcal{X}\to\mathbb{R}$ satisfying (8). Then, it holds that $\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}\subset\mathcal{X}_{\text{safe}}$, where

$$\bar{\xi} = \gamma \min_{\boldsymbol{x}\in\mathcal{X}} C(\boldsymbol{f}^\star,\boldsymbol{\pi};\boldsymbol{x}) + \hat{c}. \tag{9}$$

Proof: Due to the lower bound for c, C is lower bounded, such that we obtain $C(\boldsymbol{f}^\star,\boldsymbol{\pi},\boldsymbol{x}) \geq c(\boldsymbol{x}) + \gamma C_{\min}$ from the Bellman equation, where $C_{\min} = \min_{\boldsymbol{x}\in\mathcal{X}} C(\boldsymbol{f}^\star,\boldsymbol{\pi};\boldsymbol{x}) \geq \frac{\underline{c}}{1-\gamma}$. Moreover, due to condition (8) we have $c(\boldsymbol{x})>\hat{c}$ for $\mathcal{X}_{\text{unsafe}}$, which yields $C(\boldsymbol{f}^\star,\boldsymbol{\pi},\boldsymbol{x}) > \hat{c}+\gamma C_{\min}$ for all $\boldsymbol{x}\in\mathcal{X}_{\text{unsafe}}$. Therefore, the level set $\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$ is completely contained in $\mathcal{X}_{\text{safe}}$, i.e., $\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}\subset\mathcal{X}_{\text{safe}}$, which concludes the proof. ■

This lemma relies on the idea that the cumulative cost can be lower bounded by $\min_{\boldsymbol{x}\in\mathcal{X}} C(\boldsymbol{f}^\star,\boldsymbol{\pi};\boldsymbol{x})$, such that any state with immediate cost c greater than $\hat{c}$ also must have an expected cumulative cost greater than $\bar{\xi}$. For the example of the indicator cost, $\bar{\xi}$ can be straightforwardly computed as $\bar{\xi}=1$ since C is trivially lower bounded by 0. It is straightforward to see that this choice of cost function generally allows to accurately approximate $\mathcal{X}_{\text{safe}}$ using $\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$, and indeed $\mathcal{X}_{\text{safe}}=\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$ is possible for deterministic dynamics with $\boldsymbol{\omega}=\mathbf{0}$. However, Lemma 3.1 is not limited to indicator type cost functions, but applies to arbitrary costs c satisfying (8). This is particularly beneficial for computing optimal policies using C, where informative gradients may aid the convergence of common RL techniques. Thus, Lemma 3.1 allows a flexible approximation of the safe set $\mathcal{X}_{\text{safe}}$ suitable for the optimization-based approaches employed in the following sections.

B. Cumulative Cost Safety Conditions

In order to express K-step δ-safety through expected cumulative costs C, it remains to derive conditions which ensure that the system state $\boldsymbol{x}_k$ stays inside the sub-level set $\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$ for all $k=1,\dots,K$ with probability δ. For this purpose, we employ techniques from stochastic stability analysis [21]. In particular, we define the cost-value function of a policy at a given state $\boldsymbol{x}$ as $V_{\boldsymbol{\pi}}(\boldsymbol{x}) \equiv C(\boldsymbol{f}^\star,\boldsymbol{\pi};\boldsymbol{x})$.

Theorem 3.1: Consider an immediate cost function c : $\mathcal{X}\to\mathbb{R}$, which satisfies (8). Assume there exists a class $\mathcal{K}$ function[2] $\alpha:\mathbb{R}\to\mathbb{R}_{0,+}$, such that

$$\mathbf{E}_{\boldsymbol{\omega}}[V_{\boldsymbol{\pi}}(\boldsymbol{x}')] \leq V_{\boldsymbol{\pi}}(\boldsymbol{x}) - \alpha(V_{\boldsymbol{\pi}}(\boldsymbol{x}) - C_{\min}) \tag{10}$$

holds for all $\boldsymbol{x}\in\mathcal{X}_{\text{safe}}$ with $\boldsymbol{x}'=\boldsymbol{f}^\star(\boldsymbol{x},\boldsymbol{\pi}(\boldsymbol{x}))+\boldsymbol{\omega}$. Then,

$$\mathbf{E}_{\boldsymbol{\omega}}[V_{\boldsymbol{\pi}}(\boldsymbol{x}')] \leq \xi < \bar{\xi} \tag{11}$$

guarantees that the policy $\boldsymbol{\pi}$ is K-step $\delta(\xi)$-safe.

Proof: The result directly follows from Lemma 3.1 and Proposition A.1, which ensure δ-safety with $\delta=\delta_{\text{FL}}(\xi)$. ■

Condition (10) effectively resembles a Foster-Lyapunov drift condition, which is satisfied if stochastic stability can be

[2] A function α : $\mathbb{R}_{0,+} \to \mathbb{R}_{0,+}$ is a class $\mathcal{K}$ function, if it is monotonically increasing and $\alpha(0)=0$.

shown with $V_{\boldsymbol{\pi}}$ as a Lyapunov function [22]. Since stability is a well-studied problem, it has been shown that this condition can be satisfied for many dynamics $\boldsymbol{f}^{\star}$, e.g., systems which are asymptotically controllable with respect to the immediate cost c [23]. In contrast to stability theory, Theorem 3.1 does require C to be positive definite or the existence of a class $\mathcal{K}$ function lower bounding C. Therefore, the conditions of Theorem 3.1 are slightly weaker than for stability.

Due to the close relationship to stability, it is straightforward to see that the increase rate of α determines the convergence rate of the system. If α is only slowly growing, a relatively small noise realization can cause an increase in the expected cumulative cost, and thereby, increases the probability δ of leaving the safe set. This can be compensated by choosing a smaller value of ξ, such that there essentially is a larger margin between the safe initial states $\boldsymbol{x}_0$ and the unsafe set $\mathcal{X}_{\mathrm{unsafe}}$. Note that the noise distribution also affects the probability δ through (10), since flat distributions with heavy tails generally cause higher values of $\mathbf{E}_{\boldsymbol{\omega}}[V_{\boldsymbol{\pi}}(\boldsymbol{x})]$ leading to smaller increase rates of α.

IV. HALLUCINATING UPPER CONFIDENCE SAFETY FILTERS

We now derive a tractable safety filter for unknown dynamics, for which merely a set of plausible models is available. To this end, we first show how the safety analysis of Section III can be extended to unknown dynamics by reformulating it as reinforcement learning problem in Section IV-A. Based on this reformulation, we present a natural approach to obtain backup policies for the safety filter by computing a *safe policy* through robust reinforcement learning in Section IV-B. Finally, the pre-computed backup policy is employed in a confidence-based safety filter for ensuring constraint satisfaction in Section IV-C.

A. Safety Certification with Unknown Dynamics

Since we assume only the availability of a set of plausible models $\mathcal{M}$, but not the true dynamics $\boldsymbol{f}^{\star}$, we cannot determine $V_{\boldsymbol{\pi}}$ and consequently cannot directly exploit Theorem 3.1 for determining a safe policy. To overcome this issue, we must be *pessimistic* about the dynamics. In particular, we define the *pessimistic cost-value* as

$$V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) \equiv \max_{\boldsymbol{f}\in\mathcal{M}} C(\boldsymbol{f},\boldsymbol{\pi};\boldsymbol{x}). \tag{12}$$

It is straightforward to see that $V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) \leq \xi \Rightarrow V_{\boldsymbol{\pi}}(\boldsymbol{x}) \leq \xi$ due to Assumption 2.1. Hence, the technique for unknown models reduces to solving (12) and verifying if its value satisfies the conditions of Theorem 3.1. To solve the pessimistic verification problem (12) we reparameterize the functions $\boldsymbol{f} \in \mathcal{M}$ following [18] as

$$\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) = \boldsymbol{\mu}(\boldsymbol{x},\boldsymbol{u}) + \beta\boldsymbol{\Sigma}(\boldsymbol{x},\boldsymbol{u})\boldsymbol{\eta}(\boldsymbol{x},\boldsymbol{u}), \tag{13}$$

where $\boldsymbol{\eta} : \mathcal{X}\times\mathcal{U} \to [-1,1]^{d_x}$, $\boldsymbol{\Sigma}(\boldsymbol{x},\boldsymbol{u}) = \mathrm{diag}(\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{u}))$ and $\boldsymbol{\mu}$ is the model mean function. We refer to $\boldsymbol{\eta}$ as the hallucinating policy in the sequel since it acts on the outputs of the dynamics similarly as a policy acts on its inputs. Moreover, the set of admissible functions $\boldsymbol{\eta}$ is defined via

$$\mathcal{N} = \{\boldsymbol{\eta} : \ -1 \leq \boldsymbol{\eta}(\boldsymbol{x},\boldsymbol{u}) \leq 1 \ \ \forall \boldsymbol{x},\boldsymbol{u} \in \mathcal{X}\times\mathcal{U}\}. \tag{14}$$

To determine that a policy is safe, we must verify that the value of the pessimistic estimate $V_{\boldsymbol{\pi}}^{(p)}$ defined as

$$V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) = \max_{\boldsymbol{\eta}\in\mathcal{N}} \mathbb{E}_{\boldsymbol{\omega}}\left[\sum_{k=0}^{\infty}\gamma^k c(\boldsymbol{x}_k)\right] \tag{15a}$$

$$\text{s.t. } \boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k,\boldsymbol{\pi}(\boldsymbol{x}_k)) + \boldsymbol{\omega}_k, \ \boldsymbol{\omega}_k \sim \rho \tag{15b}$$

$$\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) = \boldsymbol{\mu}(\boldsymbol{x},\boldsymbol{u}) + \beta\boldsymbol{\Sigma}(\boldsymbol{x},\boldsymbol{u})\boldsymbol{\eta}(\boldsymbol{x},\boldsymbol{u}) \tag{15c}$$

$$\boldsymbol{x}_0 = \boldsymbol{x} \tag{15d}$$

satisfies the conditions of Theorem 3.1. Computing $V_{\boldsymbol{\pi}}^{(p)}$ requires to solve an RL problem which can be accomplished using standard RL algorithms, where the policy is $\boldsymbol{\eta}$ and the dynamics is given by (13), i.e., it can be done purely in simulation. Using this formulation as an optimization of the hallucinating policy $\boldsymbol{\eta}$, it is straightforward to extend Theorem 3.1 to unknown dynamics $\boldsymbol{f}^{\star}$ as shown in the following proposition.

Proposition 4.1: Consider a set of plausible models $\mathcal{M}$ satisfying Assumption 2.1 and an immediate cost c, which satisfies (8). If $V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x})$ satisfies

$$\max_{\boldsymbol{\eta}\in\mathcal{N}} \mathbf{E}_{\boldsymbol{\omega}}\Big[V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}')\Big] \leq V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) - \alpha(V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) - C_{\min}) \tag{16}$$

for all $\boldsymbol{x}\in\mathcal{X}_{\mathrm{safe}}$, where $\boldsymbol{x}'$ is the next state defined through the reparameterized dynamics (15b), (15c), for all $\boldsymbol{x}\in\mathcal{X}_{\mathrm{safe}}$, then, $\max_{\boldsymbol{\eta}\in\mathcal{N}}\mathbf{E}_{\boldsymbol{\omega}}[V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}')] \leq \xi < \bar{\xi}$ guarantees K-step δ-safety of $\boldsymbol{\pi}$.

Proof: Due to Assumption 2.1, $V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) \leq \bar{\xi}$ implies $V_{\boldsymbol{\pi}}(\boldsymbol{x}) \leq \bar{\xi}$ with probability at least $1-\delta_f$. Therefore, $\max_{\boldsymbol{\eta}\in\mathcal{N}} \hat{V}(\boldsymbol{x}') \leq \bar{\xi}$ implies that $\boldsymbol{x} \in \mathcal{X}_{\mathrm{safe}}$ due to Lemma 3.1. Moreover, it can be directly shown that

$$\max_{\boldsymbol{\eta}\in\mathcal{N}} \mathbf{E}_{\boldsymbol{\omega}}\Big[V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{f}^{\star}(\boldsymbol{x},\boldsymbol{\pi}(\boldsymbol{x})) + \boldsymbol{\omega})\Big] \leq \max_{\boldsymbol{\eta}\in\mathcal{N}} \mathbf{E}_{\boldsymbol{\omega}}\Big[V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}')\Big]$$

for $\boldsymbol{x}'$ the next state defined through the reparameterized dynamics (15b), (15c) with probability at least $(1-\delta_f)$ due to Assumption 2.1. Due to (16), this yields

$$\max_{\boldsymbol{\eta}\in\mathcal{N}}\mathbf{E}_{\boldsymbol{\omega}}\Big[V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{f}^{\star}(\boldsymbol{x},\boldsymbol{\pi}(\boldsymbol{x})) + \boldsymbol{\omega})\Big] \leq V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) - \alpha(V_{\boldsymbol{\pi}}^{(p)}(\boldsymbol{x}) - C_{\min}),$$

such that we can apply Proposition A.1. Hence safety follows with $\delta = \delta_{\mathrm{FL}}(\xi) + \delta_f - \delta_{\mathrm{FL}}(\xi)\delta_f$. ■

When the model is known accurately, i.e., $\boldsymbol{\sigma}(\boldsymbol{x},\boldsymbol{u}) = \mathbf{0}$ for all $\boldsymbol{x},\boldsymbol{u} \in \mathcal{X}\times\mathcal{U}$, the conditions of Proposition 4.1 intuitively reduce to the conditions of Theorem 3.1.

B. Robust Reinforcement Learning of Safe Policies

Based on the formulation of K-step δ-safety as an optimization problem in Proposition 4.1, it is natural to augment the optimization problem to directly find δ-safe policies by determining the policy that minimizes the pessimistic cost estimate $V_{\boldsymbol{\pi}}^{(p)}$. Namely, we propose to obtain safe policies via

$$\boldsymbol{\pi}_{\mathrm{safe}} := \arg\min_{\boldsymbol{\pi}\in\Pi} V_{\boldsymbol{\pi}}^{(p)} = \arg\min_{\boldsymbol{\pi}\in\Pi}\max_{\boldsymbol{\eta}\in\mathcal{N}} \mathbf{E}_{\boldsymbol{x}}[C(\boldsymbol{f},\boldsymbol{\pi};\boldsymbol{x})] \tag{17}$$

where the dynamics $\boldsymbol{f}$ is the reparameterized dynamics in (13). As (17) can be seen as a robust RL problem, we refer to $\boldsymbol{\pi}_{\text{safe}}$ as the *learned safe policy* in the following. To solve (17) we use standard robust RL techniques such as those in [24], [25], which perform gradient descent for $\boldsymbol{\pi}$ and gradient ascent for $\boldsymbol{\eta}$. Moreover, if the cost c and the discount γ allow to establish the safety of this system for some policy $\boldsymbol{\pi}$, it is straightforward to show that (17) yields a δ-safe policy.

Proposition 4.2: Assume that the learned safe policy $\boldsymbol{\pi}_{\text{safe}}$ is obtained using (17) with sufficiently expressive function classes Π, $\mathcal{N}$, and has a unique solution. If there exists a policy $\boldsymbol{\pi}$ and a class $\mathcal{K}$ function α such that $V^{(p)}_{\boldsymbol{\pi}}$ satisfies (16), $\boldsymbol{\pi}_{\text{safe}}$ is K-step δ-safe for all $\boldsymbol{x}\in\mathcal{X}$ with $V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x})\leq\xi<\bar{\xi}$.

Proof: Given the policy $\boldsymbol{\pi}_{\boldsymbol{x}}$ defined as

$$\boldsymbol{\pi}_{\boldsymbol{x}} = \arg\min_{\boldsymbol{\pi}\in\Pi}\max_{\boldsymbol{\eta}\in\mathcal{N}} C(\boldsymbol{f},\boldsymbol{\pi};\boldsymbol{x}),$$

we can lower bound the expected cost for arbitrary $\boldsymbol{\pi}\in\Pi$ by $\mathbf{E}_{\boldsymbol{x}}[V^{(p)}_{\boldsymbol{\pi}}] \geq \mathbf{E}_{\boldsymbol{x}}[V^{(p)}_{\boldsymbol{\pi}_x}]$. Therefore, for a sufficiently expressive policy class Π, it must follow that $\boldsymbol{\pi}_x = \arg\min_{\boldsymbol{\pi}\in\Pi}\mathbf{E}_{\boldsymbol{x}}[V^{(p)}_{\boldsymbol{\pi}}]$ holds. Moreover, we have $\mathbf{E}_{\boldsymbol{x}}[V^{(p)}_{\boldsymbol{\pi}_x}] \geq \mathbf{E}_{\boldsymbol{x}}[C(\boldsymbol{f},\boldsymbol{\pi}_x;\boldsymbol{x})]$ for arbitrary $\boldsymbol{\eta}\in\mathcal{N}$. Therefore, for a sufficiently expressive function class $\mathcal{N}$, it must hold that $\max_{\boldsymbol{\eta}\in\mathcal{N}}\mathbf{E}_{\boldsymbol{x}}[C(\boldsymbol{f},\boldsymbol{\pi}_x;\boldsymbol{x})] = \mathbf{E}_{\boldsymbol{x}}[V^{(p)}_{\boldsymbol{\pi}_x}]$, such that $\boldsymbol{\pi}_{\text{safe}} = \boldsymbol{\pi}_x$ is ensured due to uniqueness of the solution. Since $V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x}) \leq V^{(p)}_{\boldsymbol{\pi}}(\boldsymbol{x})$ is satisfied due to point-wise optimality of $\boldsymbol{\pi}_x$ and (16) can be straightforwardly reformulated to

$$(1-\gamma)V^{(p)}_{\boldsymbol{\pi}}(\boldsymbol{x}) + \gamma\alpha(V^{(p)}_{\boldsymbol{\pi}}(\boldsymbol{x}) - C_{\min}) \leq c(\boldsymbol{x}),$$

safety of $\boldsymbol{\pi}_{\text{safe}}$ immediately follows from the existence of a δ-safe policy and Proposition 4.1. ■

Since there exist combinations of dynamics $\boldsymbol{f}^{\star}$ and safe sets $\mathcal{X}_{\text{safe}}$ for which safety cannot be ensured, Proposition 4.2 cannot guarantee the $\boldsymbol{\pi}_{\text{safe}}$ to be always K-step δ-safe. However, as discussed in Section III-B, there exist system classes for which (10) can be satisfied. Moreover, the necessary function classes for (17) are well-known for many systems, e.g., it is straightforward to see that continuous systems and costs require piece-wise continuous policies in general.

C. Ensuring Constraint Satisfaction with Safety Filters

While the learned safe policy $\boldsymbol{\pi}_{\text{safe}}$ is safe during a policy roll-out under certain assumptions, it can possibly result in bad performance since it does not consider the reward function r. In contrast, the nominal policy $\boldsymbol{\pi}^{*}$ (4a) results in a high reward of generated trajectories, but can possibly lead to unsafe states. Therefore, we ideally want to maintain the beneficial properties of both policies, while avoiding their shortcomings. The core idea for achieving this relies on a continuous monitoring of every nominal action $\boldsymbol{\pi}^{*}(\boldsymbol{x})$, such that they can be adapted to ensure a safe roll-out of $\boldsymbol{\pi}_{\text{safe}}$ afterwards. Using the reparameterization (13) of the set of plausible models $\mathcal{M}$, this yields our *confidence-based safety filter*

$$\hat{\boldsymbol{\pi}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{U}} \|\boldsymbol{\pi}(\boldsymbol{x}) - \boldsymbol{u}\|, \tag{18a}$$

$$\text{s.t.} \max_{\boldsymbol{n}\in[-1,1]^{d_x}} \mathbf{E}_{\boldsymbol{\omega}}\Big[V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x}')\Big] \leq \xi, \tag{18b}$$

$$\boldsymbol{x}' = \boldsymbol{\mu}(\boldsymbol{x},\boldsymbol{u}) + \beta\boldsymbol{\Sigma}(\boldsymbol{x},\boldsymbol{u})\boldsymbol{n} + \boldsymbol{\omega},\ \ \boldsymbol{\omega}\sim\rho. \tag{18c}$$

Since it cannot be ensured that the state $\boldsymbol{x}_k$ satisfies $V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x}_k) \leq \xi$ for all k, a recovery mechanism steering the system back into this sub-level set is required. This can be straightforwardly achieved using the learned safe policy $\boldsymbol{\pi}_{\text{safe}}$, resulting in the overall roll-out policy

$$\tilde{\boldsymbol{\pi}}(\boldsymbol{x}) = \begin{cases} \hat{\boldsymbol{\pi}}(\boldsymbol{x}) & \text{if } V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x}) \leq \xi \\ \boldsymbol{\pi}_{\text{safe}}(\boldsymbol{x}) & \text{if } V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x}) > \xi \end{cases}. \tag{19}$$

Due to its strong foundation on the learned safe policy $\boldsymbol{\pi}_{\text{safe}}$, the roll-out policy $\tilde{\boldsymbol{\pi}}$ inherits its theoretical safety guarantees as shown in the following theorem.

Theorem 4.1: Consider a set of plausible models $\mathcal{M}$ satisfying Assumption 2.1 and assume that the learned safe policy $\boldsymbol{\pi}_{\text{safe}}$ satisfies the conditions of Proposition 4.1. Then, the confidence-based safety filtered policy (19) is K-step δ-safe for all states $\boldsymbol{x}\in\mathcal{X}$ with $V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x}) \leq \xi < \bar{\xi}$.

Proof: Since $\boldsymbol{\pi}_{\text{safe}}$ satisfies the conditions of Proposition 4.1, the trivial solution $\boldsymbol{u} = \boldsymbol{\pi}_{\text{safe}}(\boldsymbol{x})$ is guaranteed to ensure (18b). Therefore, (18) is feasible for all states $\boldsymbol{x}\in\mathcal{X}$ with $V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x}) \leq \xi$, such that the K-step δ-safety of $\tilde{\boldsymbol{\pi}}$ follows directly from Proposition 4.1. ■

While the safety filter problem (18) is not compatible with standard reinforcement learning methods, it can easily be solved on-line in the fashion of model predictive control. In order to see this, note that the pessimistic cost-value $V^{(p)}_{\boldsymbol{\pi}_{\text{safe}}}(\boldsymbol{x})$ can be efficiently obtained offline using actor-critic methods for reinforcement learning similar to $V^{(p)}_{\boldsymbol{\pi}}$ in Section IV-A. Hence, (18) requires optimization merely for one time step and consequently only for a single actual and hallucinating adversarial action in contrast to similar predictive safety filter approaches [16], [17], which require optimization over a sequence of actions. Therefore, (18) can be solved with comparatively low computational complexity using numerical optimization schemes, which allows a straightforward on-line application as safety filter.

Remark 4.1: For practical implementation of (18), ξ can be considered a tuning parameter. The smaller its value is, the higher the probability of safety is. However, a small ξ will lead to more conservatism of the safety filter, such that it must be carefully chosen to trade-off safety and performance.

V. EXPERIMENTAL RESULTS

In this section, we evaluate the safety filter and compare it with three competing algorithms: the constraint-free model-free algorithm SAC [26], a Lagrangian primal-dual approach with SAC as the base algorithm, which we call CMDP [8], and the model-based alternative Safe-CEM [27]. We consider two widely used environments to test our approach. First, we test it on an airplane pitch control example [28], where the pitch angle θ starts at -0.2 radians and the constraint function is simply $c_t = \theta_t$, such that the angle should never exceed 0. The reward is given by $r_t = -2\theta^2 + 0.02u^2$, where u is the control input. Second, we use the Mujoco Half-Cheetah environment with the default reward function [29]. The constraint is that the forward speed is less than 2. Due to the Cheetah's trot, the penalty is on the average forward speed, calculated as

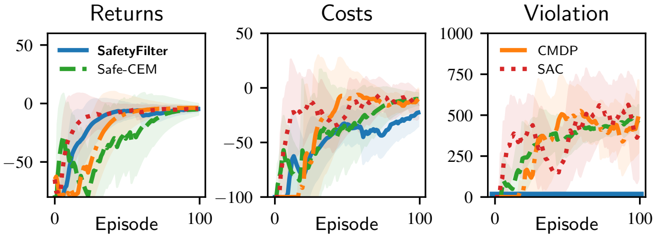


Fig. 2. Total returns, costs, and constraint violation in the Pitch Control environment. Only the safety filter attains **no** constraint violations and achieves comparable performance to the benchmarks.

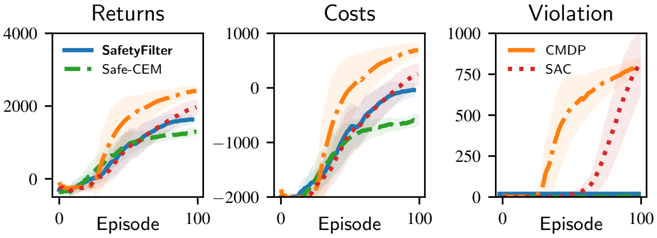


Fig. 3. Total returns, costs, and constraint violation in the Half-Cheetah environment. The safety filter and Safe-CEM achieve **no** constraint violation. These two algorithms perform slightly worse than the benchmarks in terms of returns, but the safety filter performs better than Safe-CEM.

$\bar{v}_t = 0.1 v_t + 0.9\bar{v}_{t-1}$, $\bar{v}_0 = 0$, where v_t is the instantaneous speed and $\bar{v}_t$ is the average speed. Thus we use $c_t = \bar{v}_t - 2$. We run each environment for 100 episodes, each episode for 1000 time steps, using $\gamma = 0.99$ as a discount factor.

To learn the model, we use deterministic ensembles of five members following [18]. Each member is a neural network with 3 fully connected layers of width 200 and Swish nonlinearities. For the first ten episodes, data is collected using a random policy. Such random policy was safe in these environments but only at the given initial conditions, i.e., it is not the learned safe policy used by the safety filter. After the initial exploration phase, the model is pre-trained for 100 iterations using Adam with learning rate 0.0005 and weight decay 0.0001. Then, after each subsequent episode, the model is updated using the additional data collected during the episode. We store the data using an experience replay buffer of at most 100000 transitions. Finally, to solve the safety filter problem (18) we use the cross-entropy method [30] with 1000 particles and 5 iterations per time-step.

In Figure 2, we show the results in the pitch control environment. In this setting, only the Safety Filter algorithm avoids any constraint violation while achieving comparable performance in terms of returns and costs. In Figure 3, we show the results for the Half-Cheetah. Here, both Safe-CEM and the safety filter avoid any constraint violations. However, the safety filter achieves higher returns than Safe-CEM. The main difference between these two environments relies on the backup policy. While in the Cheetah it is enough to *do nothing* in order to stop it, in the Pitch Control environment this is not the case and the *learned* safe backup policy is crucial to ensure safety. Thus, with these two environments we demonstrate the scalability of our method in the Half Cheetah environment as well as the ability to satisfy constraints in the Pitch Control environment.

VI. Conclusion

In this paper, we introduced confidence-based safety filters, a novel approach for ensuring the safety of nominal policies learnt via standard reinforcement learning techniques. The approach relies on a reformulation of state constraints as cumulative costs, such that safety conditions can be expressed as cost constraints. This allows us to obtain safe policies via robust reinforcement learning, which can be used as “backup” policy in a safety filter. We demonstrated the effectiveness and scalability of our approach in simulations.

Appendix

Lemma A.1: If there exists a function $V : \mathcal{X} \to \mathbb{R}$ such that

$$\mathbf{E}_{\omega}[V(\boldsymbol{f}^{\star}(\boldsymbol{x}, \boldsymbol{\pi}(\boldsymbol{x})) + \omega)] \leq \theta_1 \tag{20}$$

for $\theta_1 \in \mathbb{R}$, then, it holds that

$$\frac{\theta_2 - \theta_1}{\theta_2 - \underline{V}} \leq \mathbb{P}\left(\boldsymbol{x}_{k+1} \in \mathcal{V}_{\theta_2} | \boldsymbol{x}_k = \boldsymbol{x}\right) \leq \frac{\theta_2 - \theta_1}{\bar{V} - \theta_2} \tag{21}$$

for every $\underline{V} \leq \theta_1 < \theta_2 < \bar{V}$, where $\underline{V} = \min_{\boldsymbol{x} \in \mathcal{X}} V(\boldsymbol{x})$ and $\bar{V} = \max_{\boldsymbol{x} \in \mathcal{X}} V(\boldsymbol{x})$.

Proof: In order to prove this lemma, we follow the ideas of [21]. It is straightforward to see that

$$(\theta_2 - \underline{V})\mathbb{P}\left(\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2} |\, \boldsymbol{x}_k = \boldsymbol{x}\right) \leq \\ \mathbf{E}_{\boldsymbol{\omega}_k}\left[\mathbb{I}_{\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2}}(V(\boldsymbol{x}_{k+1}) - \underline{V}) |\, \boldsymbol{x}_k = \boldsymbol{x}\right], \tag{22}$$

where $\mathcal{V}_{\theta_2} = \{\boldsymbol{x} \in \mathcal{X} : V(\boldsymbol{x}) < \theta_2\}$, and $\mathbb{I}_{\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2}} = 1$ if $\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2}$ and 0 otherwise, since $\mathbb{I}_{\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2}} V(\boldsymbol{x}_{k+1}) \geq \mathbb{I}_{\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2}} \theta_2$. Moreover, we trivially have

$$\mathbf{E}_{\boldsymbol{\omega}_k}\left[\mathbb{I}_{\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2}}(V(\boldsymbol{x}_{k+1}) - \underline{V}) |\, \boldsymbol{x}_k = \boldsymbol{x}\right] \leq \\ \mathbf{E}_{\boldsymbol{\omega}_k}[(V(\boldsymbol{x}_{k+1}) - \underline{V}) |\, \boldsymbol{x}_k = \boldsymbol{x}] . \tag{23}$$

By combining (20), (22) and (23), we therefore obtain $(\theta_2 - \underline{V})\mathbb{P}\left(\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2} |\, \boldsymbol{x}_k = \boldsymbol{x}\right) \leq -\underline{V} + \theta_1$, which results in

$$\mathbb{P}\left(\boldsymbol{x}_{k+1} \notin \mathcal{V}_{\theta_2} |\, \boldsymbol{x}_k = \boldsymbol{x}\right) \leq \frac{\theta_1 - \underline{V}}{\theta_2 - \underline{V}}. \tag{24}$$

The proof for the upper bound is analogous. ■

Proposition A.1: Assume there exists a function $V : \mathcal{X} \to \mathbb{R}$ and a class $\mathcal{K}$ function $\alpha : \mathbb{R} \to \mathbb{R}_{0,+}$, such that

$$\mathbf{E}_{\omega}[V(\boldsymbol{f}^{\star}(\boldsymbol{x}, \boldsymbol{\pi}(\boldsymbol{x})) + \omega)] - V(\boldsymbol{x}) \leq -\alpha(V(\boldsymbol{x})) \tag{25}$$

holds for all $\boldsymbol{x} \in \mathcal{V}_{\bar{\xi}}$ for $\bar{\xi} \in \mathbb{R}$. Then, $\mathbf{E}_{\omega}[V(\boldsymbol{f}^{\star}(\boldsymbol{x}, \boldsymbol{\pi}(\boldsymbol{x})) + \omega)] \leq \xi$ with $\xi < \bar{\xi}$ ensures

$$\mathbb{P}\left(V(\boldsymbol{x}_k) \leq \bar{\xi}\ \forall k = 1, \ldots, K | \boldsymbol{x}_0 = \boldsymbol{x}\right) \geq 1 - \delta_{\mathrm{FL}}(\xi) \tag{26}$$

with

$$\delta_{\mathrm{FL}}(\xi) = \begin{bmatrix} 1 & \cdots & 0 \end{bmatrix} \begin{bmatrix} 1 & \mathbf{1}^T(\boldsymbol{I} - [\boldsymbol{P}]_+) \\ \mathbf{0} & [\boldsymbol{P}]_+ \end{bmatrix}^K \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix}, \tag{27}$$

where the elements of $\boldsymbol{P}$ are defined as

$$p_{i,j} = \begin{cases} \frac{\theta^i - \theta^j + \alpha(\theta^{j+1} + \underline{V})}{\theta^i - \underline{V}} - \frac{\theta^{i+1} - \theta^j + \alpha(\theta^{j+1} + \underline{V})}{\bar{V} - \theta^j + \alpha(\theta^{j+1} + \underline{V})} & \text{if } i \leq j \\ \frac{(1 - \vartheta)\alpha(\theta^{j+1} + \underline{V})}{\theta^j - \alpha(\theta^{j+1} + \underline{V}) - \underline{V}} & \text{if } i = j+1. \\ 0 & \text{if } i > j+1 \end{cases} \tag{28}$$

and M is the largest integer such that $\theta^1 \leq \bar{\xi}$ for θ^i recursively defined by $\theta^{i-1} = \theta^i + \vartheta\alpha(\theta^i + \underline{V})$ with θ^{M+1}

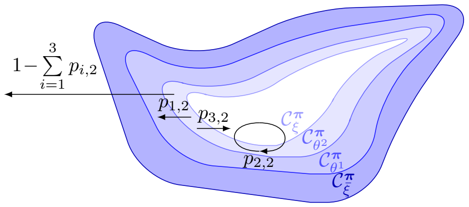

$1-\sum_{i=1}^{3} p_{i,2}$
$p_{1,2}$
$p_{3,2}$
$p_{2,2}$
$\mathcal{C}^{\boldsymbol{\pi}}_{\xi}$
$\mathcal{C}^{\boldsymbol{\pi}}_{\theta^2}$
$\mathcal{C}^{\boldsymbol{\pi}}_{\theta^1}$
$\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$

Fig. 4. In order to certify the K-step δ-safety of a policy $\boldsymbol{\pi}$, we define a sequence of sub-level sets $\mathcal{C}^{\boldsymbol{\pi}}_{\theta^i}$ with decreasing thresholds θ^i. We can bound the probabilities for transitioning to other sub-level sets in each time step using Lemma A.1 as exemplarily illustrated for $\mathcal{C}^{\boldsymbol{\pi}}_{\epsilon^2}$, such that the probability of leaving the $\mathcal{C}^{\boldsymbol{\pi}}_{\bar{\xi}}$ can be bounded using methods for Markov chains.

implicitly defined via $\theta^{M+1} + (\vartheta - 1)\alpha(\theta^{M+1} + \underline{V}) = \xi$ and sufficiently small $\vartheta \in (0, 1)$.

Proof: For proving this proposition, we construct a sequence of sub-level sets $\mathcal{V}_{\theta^j}$ as illustrated in Fig. 4 and bound the transition probabilities between them using Lemma A.1. Given a sub-level set $\mathcal{V}_{\theta^j}$, the probability of transitioning into sub-level set $\mathcal{V}_{\theta^{j+1}}$ can be lower bounded using $\theta_2 = \theta^j - \vartheta\alpha(\theta^{j+1} - \underline{V})$, $\theta_1 = \theta^j - \alpha(\theta^{j+1} - \underline{V})$, which yields

$$p_{j+1,j} = \frac{(1-\vartheta)\alpha(\theta^{j+1} + \underline{V})}{\theta^j - \alpha(\theta^{j+1} + \underline{V}) - C_{\min}}.$$

For transitioning from the sub-level set $\mathcal{V}_{\theta^j}$ to a sub-level set $\mathcal{V}_{\theta^i}$, $i \leq j$, we have

$$\mathbb{P}\left(\boldsymbol{x}_{t+1} \in \mathcal{V}_{\theta^i} \setminus \mathcal{V}_{\theta^{i+1}} | \boldsymbol{x}_t \in \mathcal{V}_{\theta^j}\right) = \mathbb{P}\left(\boldsymbol{x}_{t+1} \in \mathcal{V}_{\theta^i} | \boldsymbol{x}_t \in \mathcal{V}_{\theta^j}\right) - \mathbb{P}\left(\boldsymbol{x}_{t+1} \in \mathcal{V}_{\theta^{i+1}} | \boldsymbol{x}_t \in \mathcal{V}_{\theta^j}\right),$$

such that applying Lemma A.1 to both summands with $\theta_2 = \theta^i$, $\theta_1 = \theta^j - \alpha(\theta^{j+1} - \underline{V})$ and $\theta_2 = \theta^{i+1}$, $\theta_1 = \theta^j - \alpha(\theta^{j+1} - \underline{V})$, respectively, yields

$$p_{i,j} = \frac{\theta^i - \theta^j + \alpha(\theta^{j+1} + \underline{V})}{\theta^i - \underline{V}} - \frac{\theta^{i+1} - \theta^j + \alpha(\theta^{j+1} + \underline{V})}{\bar{V} - \theta^j + \alpha(\theta^{j+1} + \underline{V})}.$$

Note that for $i = M$ we have $\theta_1 = \xi$. Since we cannot guarantee to directly transition from sub-level sets $\mathcal{V}_{\theta^j}$ to sub-level sets $\mathcal{V}_{\theta^i}$ with $i \geq j+2$, we obtain the trivial bound $p_{i,j} = 0$ in this case, which results in (28). Based on the bounds $p_{i,j}$, we can construct a left stochastic matrix similar to the transition matrix of a Markov chain, whose first row corresponds to an absorbing state as shown in (27). Since the first state is absorbing and the transition probabilities to all other states are lower bounds, multiplying this matrix K times with itself and multiplying the initial probability distribution from the right yields the upper bound δ for leaving the sub-level set $\mathcal{V}_{\bar{\xi}}$ within K time steps. ■

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2017.

[2] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *International Conference on Learning Representations*, 2016.

[3] G. Dulac-Arnold, D. Mankowitz, and T. Hester, "Challenges of real-world reinforcement learning," in *ICML Workshop on Real-Life Reinforcement Learning*, 2019.

[4] J. García and F. Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, pp. 1437–1480, 2015.

[5] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, pp. 411–444, 2022.

[6] E. Altman, *Constrained Markov Decision Processes*. CRC Press, 1999.

[7] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *International Conference on Machine Learning*, 2017, pp. 30–47.

[8] S. Paternain, L. F. Chamon, M. Calvo-Fullana, and A. Ribeiro, "Constrained reinforcement learning has zero duality gap," in *Advances in Neural Information Processing Systems*, 2019, pp. 7555–7565.

[9] D. Ding, X. Wei, Z. Yang, Z. Wang, and M. Jovanović, "Provably efficient safe exploration via primal-dual policy optimization," in *International Conference on Artificial Intelligence and Statistics*, 2021, pp. 3304–3312.

[10] S. Dean, S. Tu, N. Matni, and B. Recht, "Safely learning to control the constrained linear quadratic regulator," in *American Control Conference*, 2019, pp. 5582–5588.

[11] U. Rosolia and F. Borrelli, "Learning model predictive control for iterative tasks. A data-driven control framework," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1883–1896, 2018.

[12] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration," in *IEEE Conference on Decision and Control*, 2018, pp. 6059–6066.

[13] A. Capone, A. Lederer, J. Umlauft, and S. Hirche, "Data selection for multi-task learning under dynamic constraints," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 959–964, 2021.

[14] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin, "A general safety framework for learning-based control in uncertain robotic systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 7, pp. 2737–2752, 2019.

[15] A. Taylor, A. Singletary, Y. Yue, and A. Ames, "Learning for safety-critical control with control barrier functions," in *Learning for Dynamics & Control*, 2019, pp. 708–717.

[16] O. Bastani, "Safe reinforcement learning with nonlinear dynamics via model predictive shielding," in *American Control Conference*, 2021, pp. 3488–3494.

[17] K. Wabersich, L. Hewing, A. Carron, and M. Zeilinger, "Probabilistic model predictive safety certification for learning-based control," *IEEE Transactions on Automatic Control*, vol. 76, no. 1, pp. 176–188, 2021.

[18] S. Curi, F. Berkenkamp, and A. Krause, "Efficient model-based reinforcement learning through optimistic policy search and planning," in *Advances in Neural Information Processing Systems*, 2020.

[19] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," *IEEE Control Systems*, vol. 36, no. 6, pp. 30–44, 2016.

[20] K. Kim and R. Braatz, "Generalised polynomial chaos expansion approaches to approximate stochastic model predictive control," *International Journal of Control*, vol. 86, no. 8, pp. 1324–1337, 2013.

[21] Y. Li, W. Zhang, and X. Liu, "Stability of nonlinear stochastic discrete-time systems," *Journal of Applied Mathematics*, vol. 2013, no. 2, 2013.

[22] S. Meyn and R. Tweedie, *Markov Chains and Stochastic Stability*, 1993.

[23] V. Gaitsgory, L. Grüne, M. Höger, C. M. Kellett, and S. R. Weller, "Stabilization of strictly dissipative discrete time systems with discounted optimal control," *Automatica*, vol. 93, pp. 311–320, 2018.

[24] S. Curi, I. Bogunovic, and A. Krause, "Combining pessimism with optimism for robust and efficient model-based deep reinforcement learning," in *International Conference on Machine Learning*, 2021, pp. 2254–2264.

[25] L. Pinto, J. Davidson, R. Sukthankar, and A. Gupta, "Robust adversarial reinforcement learning," in *International Conference on Machine Learning*, 2017, pp. 4310–4319.

[26] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning*, 2018, pp. 1861–1870.

[27] Z. Liu, H. Zhou, B. Chen, S. Zhong, M. Hebert, and D. Zhao, "Constrained model-based reinforcement learning with robust cross-entropy method," *arXiv preprint arXiv:2010.07968*, 2020.

[28] R. Hafner and M. Riedmiller, "Reinforcement learning in feedback control," *Machine learning*, vol. 84, no. 1, pp. 137–169, 2011.

[29] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *IEEE/RSJ international conference on intelligent robots and systems*, 2012, pp. 5026–5033.

[30] Z. I. Botev, D. P. Kroese, R. Y. Rubinstein, and P. L'Ecuyer, "The cross-entropy method for optimization," in *Handbook of Statistics*. Elsevier, 2013, vol. 31, pp. 35–59.