Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

EigenSafe: A Spectral Framework for Learning-Based Probabilistic Safety Assessment

Inkyu Jang*§, Jonghae Park*§, Sihyun Cho†, Chams E. Mballo‡, Claire J. Tomlin‡ and H. Jin Kim§
§Department of Aerospace Engineering, Seoul National University, Seoul, Korea
†Interdisciplinary Program in Artificial Intelligence, Seoul National University, Seoul, Korea
‡Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, USA
*Equal Contribution

***Abstract*—We present EigenSafe, an operator-theoretic framework for safety assessment of learning-enabled stochastic systems. In many robotic applications, the dynamics are inherently stochastic due to factors such as sensing noise and environmental disturbances, and it is challenging for conventional methods such as Hamilton-Jacobi reachability and control barrier functions to provide a well-calibrated safety critic that is tied to the actual safety probability. We derive a linear operator that governs the dynamic programming principle for safety probability, and find that its dominant eigenpair provides critical safety information for both individual state-action pairs and the overall closed-loop system. The proposed framework learns this dominant eigenpair, which can be used to either inform or constrain policy updates. We demonstrate that the learned eigenpair effectively facilitates safe reinforcement learning. Further, we validate its applicability in enhancing the safety of learned policies from imitation learning through robot manipulation experiments using a UR3 robotic arm in a food preparation task.**

Project Webpage: https://eigen-safe.github.io
Code: https://github.com/jonghaepark/eigensafe

I. Introduction

A. Motivation

Ensuring safe operation is a fundamental requirement for the deployment of autonomous robotic systems, yet achieving it is challenging due to the inherent stochasticity of real-world operations. Deployed robots face unpredictable environmental disturbances, sensor noise, and the uncertainties introduced by learning-enabled components. In these stochastic settings, the traditional *binary* view of safety, where a state is strictly classified as either safe or unsafe, becomes unrealistic. Instead, safety is quantified according to the probability of not experiencing any failure over a given time horizon.

Therefore, it is essential to accurately estimate the closed-loop safety probability and understand how it evolves over long time horizons. Achieving this requires a safety critic that can evaluate this probability. Prior approaches based on optimal control, e.g., Hamilton-Jacobi (HJ) reachability analysis [5, 20] and learning-based control barrier functions (CBFs) [3, 53], have successfully demonstrated their capabilities as learning-based safety critics. However, they typically learn surrogate values, such as value functions or barrier scores, which do not inherently correspond to the actual safety probability. Moreover, because these critics are not explicitly calibrated to the closed-loop safety probability, their outputs can be inconsistent or overly conservative, leading to overly cautious policies that unnecessarily sacrifice performance.

These issues can be resolved by using a safety critic that is *well-calibrated* to the evolution of the true closed-loop safety probability over long time horizons. A critic that serves as a consistent proxy for the actual probability would enable a precise trade-off between performance and safety, reducing the unnecessary conservatism inherent in uncalibrated metrics. This necessitates a theoretical framework that directly links the learned safety critic to the probabilistic safety assessment of the system.

B. Summary of Contributions

In this work, we present **EigenSafe**, an operator-theoretic framework for accurate safety assessment of stochastic robotic systems. Within this framework, the evolution of the safety probability is viewed as a repeated application of a linear operator. The operator's dominant eigenpair, consisting of the dominant eigenvalue and eigenfunction, is shown to inherently capture the safety behavior of the closed-loop system over long time horizons. Our contributions are summarized below.

Spectral Safety Assessment. We develop a novel safety assessment method that can serve as an alternative to traditional optimal control-based approaches. Its key advantage is that the derived spectral metrics are explicitly connected to the actual safety probability of the system, which is the most natural measure of safety. This theoretical grounding allows for a rigorous and unified assessment of safety, both locally for states and state-action pairs (via the dominant eigenfunction) and globally for the entire closed-loop system (via the dominant eigenvalue).

Learning Algorithm. We propose a scalable, data-driven learning algorithm to approximate these spectral quantities from trajectory data without requiring an explicit model of the system dynamics. Inspired by the power iteration algorithm, a classical method for finding the dominant eigenpair, we derive a loss functional that is minimized when the estimates converge to it. This allows the spectral safety critic to be learned using standard gradient descent.

Application to Reinforcement Learning. We demonstrate the utility of EigenSafe in safe reinforcement learning (RL) tasks by formulating a safety-constrained policy optimization. This allows one to find a policy that balances well

between maximizing the task rewards and maintaining long-term safety. Validation experiments are carried out in Gym [7] environments, showing comparable results both in terms of performance and safety to modern baselines.

Application to Imitation Learning. We apply EigenSafe to imitation learning (IL) by introducing a test-time safety filtering mechanism for learned stochastic policies. This ensures that only sufficiently safe input among multiple candidates is applied into the system, thereby achieving enhanced safety of the overall closed loop. We validate the approach in a real-world food preparation task using a UR3 robotic arm.

II. Related Work

We review relevant works on safety-critical control and their extensions to stochastic systems, as well as learning-based methods for safety critics and safe policies.

Safety-Critical Control Theory. HJ reachability [5, 39] and CBFs [3] are widely studied control methodologies for the safety of dynamical systems. In these frameworks, the reachability value function and the barrier function act as quantitative safety critics that measure the margin of safety for a given state with respect to a prescribed risk measure. Closest to EigenSafe are stochastic extensions of CBFs. These extensions provide time-varying lower bounds on the safety probability, often in the form of exponentially decaying bounds [13, 14, 29, 40, 46, 49, 54, 55, 58]. However, they are typically derived from conservative martingale inequalities, which can be overly conservative and often require specialized techniques to tighten the gap [15, 30]. In practice, these safety-critical control methods are often deployed in the form of a *safety filter* [56], whose role is to render a potentially unsafe reference policy safe by minimally modifying it [2].

Safety Critic Learning. A growing body of work combines these control-theoretic methods with machine learning to synthesize safety critics [8, 16, 20, 22]. Learning CBFs or HJ reachability value functions is relatively straightforward when the dynamics model is at least partially known [4, 11, 37, 47, 48, 53]. However, this assumption can be unrealistic for complex or high-dimensional systems, such as systems subject to implicit safety constraints [10, 12, 31, 32, 43, 50] or systems whose dynamics are represented through learned world models [24, 25, 42, 60]. In these settings, model-free approaches based on RL [18, 27, 35] are better suited. Nevertheless, these approaches typically rely on optimal control formulations for deterministic systems and fail to capture the stochastic nature of real robotic systems. Moreover, because the learned critics are often tied to specific risk measures, they serve only as surrogates and are not calibrated to the actual safety probability. As a result, these safety critics tend to yield inconsistent safety assessments. A few works propose HJ reachability formulations for safety probability, thereby correctly handling these issues [19, 28]. However, the infinite-horizon safety probability they consider is typically zero everywhere unless the system can be made almost-surely safe, which makes it difficult to compare relative safety across different states.

Safe Decision Making. The goal of learning safety critics is to enable safe decision making. Safety filters are frequently employed with learned safety critics, often in the form of a switching filter, where the reference input is applied directly until the critic indicates a low safety level, after which the system falls back to a safe backup policy [43, 50]. However, this approach may lead to suboptimal task performance because it lacks a principled mechanism to balance safety and performance. In safe RL, on the other hand, the decision-making problem is formulated as a constrained Markov decision process (CMDP), where the objective is to maximize rewards subject to safety constraints [8, 21, 22]. These safety constraints are commonly enforced by the use of Lagrange multipliers [1, 19] or epigraph reformulations [52, 59]. Since the constraints in such optimizations are naturally defined with respect to the learned safety critic, their performance heavily depends on its quality. When the safety metric is poorly calibrated, these algorithms often produce overly conservative behavior.

III. Problem Setup

A. Safety-Critical Dynamics

The robotic system considered in this paper is modeled as a discrete-time Markov decision process (MDP) without the reward term, a 3-tuple (S, A, P), where S and A are the state and action spaces, respectively, and P is the transition probability that describes the dynamics:

$$s_{t+1} \sim P(\cdot|s_t, a_t). \tag{1}$$

Here, $s_t, s_{t+1} \in S$ are the system's state at time t and $t+1$, respectively, and $a_t \in A$ is the action taken at time t.

We consider a safety constraint $s_t \in C$, where $C \subsetneq S$ is the *safe set*, i.e., the set of allowable states. For example, C could represent the set of states satisfying a collision avoidance constraint. Once the system leaves C for the first time, a permanent safety failure is recorded, and the subsequent behavior is considered unsafe. We model this using the notion of a *termination state* $K \notin C$. If $s_t \notin C$, then the trajectory is marked as stopped and terminated by setting $s_{t'} = K$ for all $t' \geq t$, regardless of the action taken thereafter. In this paper, we do not distinguish between different failure modes, so we can assume there exists only one unsafe state and write the state space as $S = C \cup \{K\}$.

The safety probability of the system is defined as the probability of the system not reaching the termination state K throughout the given horizon, i.e., $\mathbb{P}[s_t \neq K]$. This probability is naturally non-increasing in time (i.e., $\mathbb{P}[s_{t+1} \neq K] \leq \mathbb{P}[s_t \neq K]$) and depends on the chosen (feedback) policy and the initial state. The problem of interest in this paper is to derive a measure that quantitatively assesses the safety of a given state, action, or a feedback policy, in terms of how this safety probability evolves over a long time horizon.

B. Dynamic Programming for Safety Probability

Let π be either a deterministic or a stochastic policy satisfying the Markov property. This means that the probability

distribution of the action taken at time t depends only on the current state s_t, i.e., $a_t \sim \pi(\cdot|s_t)$, for all $t \geq 0$. Under this policy, we define $Z_\pi(t,x)$ as the probability that the system stays in the safe set until time t given the initial state $s_0 = x \in S$:

$$Z_\pi(t,x) \coloneqq \mathbb{P}_\pi[s_t \neq K | s_0 = x], \tag{2}$$

where $\mathbb{P}_\pi$ denotes the probability measure induced by the closed-loop dynamics under the policy π.

It follows directly from the definition that $Z_\pi(0,x) = 1_C(x)$, where $1_C : S \to \{0,1\}$ is the indicator function of C. From the law of total probability and the Markov property, one can easily see that the $(t+1)$-step safety probability from the initial condition $s_0 = x$, denoted $Z_\pi(t+1,x)$, can be expressed as the expectation of the t-step safety probability from the next state $s_1 = x'$, $Z_\pi(t,s_1)$. This yields the following dynamic programming principle: for any policy π, $t \in \{0,1,\cdots,\}$, and $x \in S$,

$$Z_\pi(t+1,x) = \mathbb{E}_{u\sim\pi(\cdot|x),\ x'\sim P(\cdot|x,u)}\left[Z_\pi(t,x')\right]. \tag{3}$$

In this paper, we view this dynamic programming principle from the perspective of an *operator* acting on the space of scalar functions on S. Formally, we start by letting $(\mathcal{F}, \|\cdot\|_\infty)$ be the Banach space of bounded and measurable scalar functions on S, where $\|\cdot\|_\infty$ is the supremum norm defined as $\|\beta\|_\infty = \sup_{x\in S}|\beta(x)|$. The operator $T_\pi : D_T \subset \mathcal{F} \to \mathcal{F}$, which describes this dynamic programming principle, is defined as

$$T_\pi\beta(x) \coloneqq \mathbb{E}_{u\sim\pi(\cdot|x),\ x'\sim P(\cdot|x,u)}[\beta(x')], \tag{4}$$

where the domain of T_π is

$$D_T \coloneqq \left\{\beta \in \mathcal{F} \;\middle|\; \beta(K) = 0\right\}, \tag{5}$$

which inherits the Banach space structure from $\mathcal{F}$. This T_π admits an integral[1] form

$$T_\pi\beta(x) = \int_S \beta(y)p_\pi(y|x)dy = \int_C \beta(y)p_\pi(y|x)dy, \tag{6}$$

where $p_\pi(\cdot|x)$ is the probability measure on S induced by the policy π, with probability density function $p_\pi(y|x) \coloneqq \int_A \pi(u|x)P(y|x,u)du$, which can also be understood as the density of the next state y given the current state x under policy π. Note that the second equality holds since $\beta(x) = 0$ for any x outside C.

One can also find that the image of T_π is a subset of D_T, making T_π an endomorphism (a mapping from D_T onto itself). This follows from the fact that the system cannot recover from an unsafe state; hence, $\beta \in D_T$ implies $T_\pi\beta(x) = 0$ for all $x \notin C$. Combined with the initial condition $Z_\pi(0,x) = 1_C(x)$, the dynamic programming principle (3) can now be expressed as a t-time repeated application of the operator T_π to the indicator function:

$$Z_\pi(t,x) = T_\pi^t 1_C(x) = \underbrace{T_\pi \circ \cdots \circ T_\pi}_{t \text{ times}} 1_C(x). \tag{7}$$

[1]The integrals hereafter should be summations in case of discrete state or action spaces.

IV. Spectral Theory for Safety Assessment

A. Spectral Properties of T_π

The operator T_π can be understood as the restriction of the Koopman operator [9, 51] of the closed-loop system to the domain D_T which is a vector subspace of the set of all measurable scalar functions on S. It is evident from the definition (4) that regardless of π, T_π is a *linear* operator. That is, if β_1 and β_2 are two scalar functions on S and c_1 and c_2 are scalars, then $T_\pi(c_1\beta_1 + c_2\beta_2) = c_1T_\pi\beta_1 + c_2T_\pi\beta_2$. This allows one to examine its spectral properties. Let the complex extension of the domain D_T of T_π be defined as $\overline{D}_T \coloneqq D_T + \mathrm{i}D_T = \{\beta_r + \mathrm{i}\cdot\beta_i \mid \beta_r, \beta_i \in D_T\}$. We define an *eigenpair* (γ, ϕ) as a pair of a scalar $\gamma \in \mathbb{C}$ and a nonzero function $\phi \in \overline{D}_T \setminus \{0\}$ satisfying

$$T_\pi\phi(x) = \gamma\phi(x), \quad \forall x \in S. \tag{8}$$

Since the domain $\overline{D}_T$ is typically infinite-dimensional (unless S is a discrete set), there are typically infinitely many such pairs. We denote the eigenvalue with the biggest absolute value as the *dominant eigenvalue*, γ_π, and its associated eigenfunction ϕ_π as the *dominant eigenfunction*. The eigenpair (γ_π, ϕ_π) is referred to as the *dominant eigenpair*.

One key characteristic of T_π is that it is a *positive operator*; that is, it maps nonnegative real functions to nonnegative real functions. This allows the use of Perron-Frobenius theory (and its infinite-dimensional generalization by Krein and Rutman [33]) to certify that γ_π is a positive real number and that ϕ_π can be chosen to be nonnegative real everywhere on S.[2]

Moreover, T_π is non-expansive with respect to the supremum norm $\|\cdot\|_\infty$. For any $x \in C$, we observe

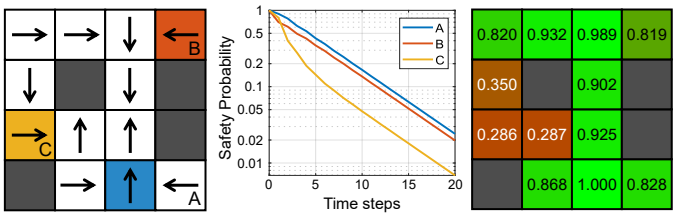
$$|T_\pi\beta(x)| = \left|\int_C \beta(y)p_\pi(y|x)dy\right| \leq \|\beta\|_\infty. \tag{9}$$

This inequality implies that the spectral radius of T_π is bounded by 1; thus, all eigenvalues, including γ_π, should have absolute values less than or equal to 1. Combining this with the positivity property mentioned above, we conclude that the dominant eigenvalue is a real number satisfying $0 \leq \gamma_\pi \leq 1$.

B. The Dominant Eigenpair as a Safety Measure

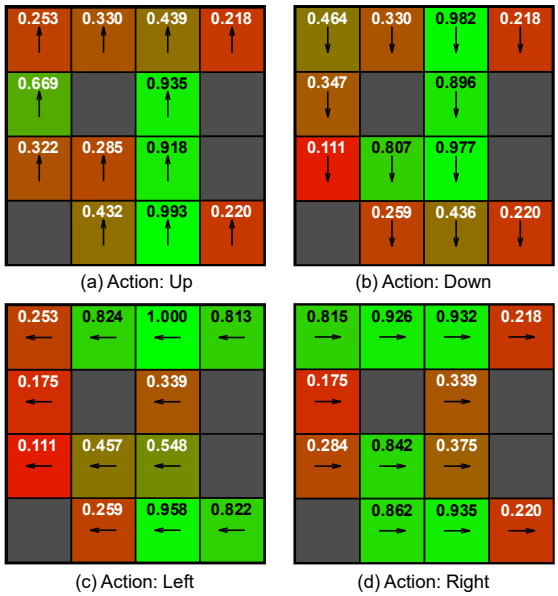
Recall that the primary objective of this paper is to analyze the long-term safety of the closed-loop system, specifically, how the safety probability Z_π of a given system evolves over long time horizons. Once the dynamic programming recursion (7) is expressed as repeated application of a linear operator, it is governed by the spectral structure of T_π, with eigenmodes propagated multiplicatively according to their associated eigenvalues. In this context, the dominant eigenmode is of particular significance because the influence of non-dominant modes decays over time and eventually become negligible relative to the dominant eigenmode.

[2]This requires the assumption that T_π is compact and the process is irreducible on the safe set C (i.e., any state in C is reachable from any other state in C at a nonzero probability prior to arriving at K), which usually holds in realistic robotic systems.



(a) The policy π (b) Safety Probability (c) ϕ_π values

Fig. 1. A toy example describing the meaning of the dominant eigenpair of T_π. (a) This toy example consists of a finite state space represented by square cells, and a finite action space represented by an arrow on each of them. At each time step, the system moves to the adjacent cell in the arrow direction with probability 0.6, and moves to each of the other adjacent cells or remains in the current cell with probability 0.1. If the system enters a gray cell or leaves the map, it is deemed unsafe and is therefore considered to have reached the unsafe state K. (b) The safety probability given initial conditions A, B, C, corresponding to the colored cells in (a). Note that the vertical axis is log-scale and the slopes of all three curves converge to the same value, which corresponds to the dominant eigenvalue γ_π. (c) The dominant eigenfunction values. It can be seen that cells with higher values correspond to safer points. For finite-state systems, one can directly perform eigendecomposition of the finite-dimensional matrix representation of T_π to obtain the eigenpair.



(a) Action: Up (b) Action: Down

(c) Action: Left (d) Action: Right

Fig. 2. The dominant eigenfunction ψ_π of A_π, defined with respect to the same dynamics and π as in Fig. 1. Similar to Fig. 1 (c), the eigenfunction can be computed by directly eigendecomposing the matrix representation of A_π. It can be seen that the eigenfunction assigns higher values to safer transitions.

Usually, the dominant eigenvalue γ_π has multiplicity 1, meaning that the dominant eigenfunction ϕ_π is unique up to scalar multiplication. Assuming a nonzero spectral gap (separation between the modulus of the dominant eigenvalue and the rest of the spectrum), the influence of non-dominant modes decays geometrically faster than γ_π^t as t grows. Thus, for sufficiently large t, $Z_\pi(t,x)$ can be approximated as

$$Z_\pi(t,x) = c \cdot \phi_\pi(x) \cdot \gamma_\pi^t + o(\gamma_\pi^t), \tag{10}$$

where $c \in \mathbb{R}$ is a constant that does not depend on t. The additive term $o(\gamma_\pi^t)$ indicates that the approximation error decays strictly faster than γ_π^t. The approximation (10) is therefore the tightest among exponential ones possible for the safety probability Z_π. In Appendix A, we prove that, under a mild set of assumptions, the dominant eigenpair is unique up to scaling and has a nonzero spectral gap, hence the mentioned probability approximation (10) holds. We also discuss the practicality of the underlying assumptions in Appendix A-I.

This approximation (10) implies that the asymptotic decay rate of Z_π is equal to γ_π and is uniform across the entire state space. Therefore, the dominant eigenvalue serves as a global metric for the safety of the closed-loop system under policy π, with a value closer to 1 indicating a safer policy. The value of the dominant eigenfunction $\phi_\pi(x)$ quantifies the relative safety of a specific state point $x \in C$ compared to other states in C. If this ϕ_π is chosen to be a nonnegative function, then a larger value of $\phi_\pi(x)$ corresponds to a higher degree of safety for that state x.

Collectively, these properties establish the dominant eigenpair (γ_π, ϕ_π) as a quantitative safety measure for the closed-loop system. These spectral quantities are intrinsically tied to the actual safety probability, which is the most natural indicator for safety. The eigenvalue γ_π is a safety measure for the closed-loop system itself or the policy, ϕ_π is the safety measure for the given state under π. See Fig. 1 (a) for a simple illustrative example using a simple finite-state system.

C. The Safety Q Function

The dominant eigenfunction ϕ_π, which depends only on the state, provides a state-wise safety metric and, as such, plays a role analogous to a *safety value function*. However, it cannot capture how safety varies with the action taken at a given state. In this subsection, we aim to derive a safety measure that provides a *directional guide* for policy improvement. Achieving this requires a safety metric defined over state-action pairs, that is, a *Q function* for safety, which extends the utility of ϕ_π.

To this end, we broaden our discussion from the operator T_π to assess safety at the level of state-action pairs, rather than solely at the state level. We introduce $H_\pi : S \times A \to \mathbb{R}$ as the safety probability conditioned on the initial state and action:

$$H_\pi(t,x,u) \coloneqq \mathbb{P}_\pi\left[s_t \neq K | s_0 = x, a_0 = u\right], \tag{11}$$

where $\mathbb{P}_\pi$ here measures the probability of the event assuming $a_k \sim \pi(\cdot|s_k)$ for all subsequent steps $k \in \{1, \cdots, t-1\}$. Similar to Z_π, H_π takes nonnegative values upper-bounded by 1 and satisfies the consistency condition

$$\mathbb{E}_{u\sim\pi(\cdot|x)}[H_\pi(t,x,u)] = Z_\pi(t,x). \tag{12}$$

Functionally, H_π represents the safety probability of an *augmented autonomous system* with state $\hat{s}_t \coloneqq (s_t, a_t)$, governed by the dynamics

$$\begin{aligned} s_{t+1} &\sim P(\cdot|s_t, a_t), \\ a_{t+1} &\sim \pi(\cdot|s_{t+1}), \end{aligned} \tag{13}$$

whose safety characteristics are the same as those of the original closed-loop system under policy π, except for the initial step $t = 0$.

Similar to T_π, we define the operator

$$A_\pi \beta(x,u) \coloneqq \mathbb{E}_{x' \sim P(\cdot|x,u), u' \sim \pi(\cdot|x')} \left[\beta(x',u')\right] \tag{14}$$

on the domain $D_A \coloneqq \left\{\beta \in \mathcal{G} \;\middle|\; \beta(K,u) = 0,\ \forall u \in A\right\}$, where $(\mathcal{G}, \|\cdot\|_\infty)$ is the Banach space of bounded and measurable scalar functions on $S \times A$, equipped with the supremum norm. This allows us to follow the same procedure in Section IV-B to derive the dynamic programming principle for H_π:

$$H_\pi(t+1,\cdot,\cdot) = A_\pi H_\pi(t,\cdot,\cdot), \tag{15}$$

which implies $H_\pi(t,x,u) = A_\pi^t 1_{C\times A}(x,u)$.

It is evident that A_π is also a nonnegative and non-expansive linear operator. Since the augmented system shares the underlying dynamics of the original system, the eigenstructure of A_π is closely related to that of T_π. Specifically, there exists a dominant eigenpair (γ_π, ψ_π) such that $A_\pi \psi_\pi(x,u) = \gamma_\pi \psi_\pi(x,u)$ for all $(x,u) \in S \times A$. The dominant eigenvalue γ_π is identical to that of T_π and the dominant eigenfunction ψ_π is a nonnegative function on $S \times A$. Furthermore, ψ_π relates to ϕ_π through the expectation over the closed-loop actions

$$\mathbb{E}_{u\sim\pi(\cdot|x)} \psi_\pi(x,u) = c \cdot \phi_\pi(x), \quad \forall x \in C, \tag{16}$$

for an appropriate constant $c \in \mathbb{R}$ not depending on the state.

Similarly to (10), one can approximate H_π as

$$\begin{aligned} H_\pi(t,x,u) &= A_\pi^t 1_{C\times A}(x,u) \\ &= c \cdot \psi_\pi(x,u) \cdot \gamma_\pi^t + o(\gamma_\pi^t), \end{aligned} \tag{17}$$

where $c \in \mathbb{R}$ is an appropriate constant. This suggests that the eigenfunction ψ_π quantifies the safety level of a given state-action pair (x,u). Just as ϕ_π plays a role as a safety value function, ψ_π serves as a safety Q function, providing a metric to guide policy improvements during training. We illustrate the role of the eigenfunction ψ_π in Fig. 2 using the same tabular dynamics as in Fig. 1.

V. EigenSafe: Learning the Eigenpair

In this section, we present a practical methodology for learning the safety Q function ψ_π and its associated eigenvalue from transition data, building upon the theoretical framework established in the previous section.

A. Eigenpair Learning

Consider a dataset, batch, or a replay buffer $\mathcal{D}$ comprising transition tuples (x,u,x') of state, action, and the next state. In its simplest form, EigenSafe parametrizes the eigenvalue γ as a learnable real scalar and approximates the eigenfunction ψ using a function approximator, such as a lookup table for discrete spaces or a neural network for continuous spaces. The parameters are optimized by minimizing the following loss functional:

$$\begin{aligned} &\mathcal{J}_{\text{eig}}[\gamma,\psi] \coloneqq \\ &\frac{W_{\text{eig}}}{|\mathcal{D}|} \sum_{(x,u,x')\in\mathcal{D}} \begin{pmatrix} 1[x \in C \wedge x' \in C] \cdot \psi(x',u') \\ - \gamma \cdot 1[x \in C] \cdot \psi(x,u) \end{pmatrix}^2 \\ &\qquad + W_n \cdot \left(\max_{(x,u,\cdot)\in\mathcal{D}} \psi(x,u) - 1 \right)^2, \end{aligned} \tag{18}$$

where u' in the summation is sampled from the policy conditioned on the next state x' from the dataset, and $W_{\text{eig}},\ W_n > 0$ are tunable hyperparameters. The indicator function $1[\cdot]$ evaluates to 1 if the condition inside the bracket holds, and 0 otherwise. The first term in (18) constrains ψ to be zero outside the safe set C, while ensuring it satisfies the eigenvalue equation $A_\pi \psi(x,u) = \gamma \psi(x,u)$ within C. The second term serves as a soft normalization constraint that ensures the learned eigenfunction maintains a unit supremum norm, $\|\psi\|_\infty = 1$, and prevents converging to a degenerate solution $\psi = 0$.

While standard Q learning algorithms from RL utilize a discount factor strictly smaller than 1 to ensure a unique solution via contraction mapping theory, the loss functional (18) does not theoretically admit a unique minimizer. This is because *any* valid eigenpair of the operator A_π, including the non-dominant ones, satisfies the eigenvalue equation enforced by (18). Instead, the convergence of the proposed approach is grounded in the principles of the *power iteration* algorithm, also known as the von Mises iteration. To determine the dominant eigenpair of a positive linear operator $L : V \to V$, power iteration initializes with a nonzero initial guess $v \in V \setminus \{0\}$ and recursively applies the update

$$\begin{aligned} \lambda &\leftarrow \|Lv\|, \\ v &\leftarrow Lv/\|Lv\|. \end{aligned} \tag{19}$$

Although this update rule is not a contraction mapping in general, the pair (λ, v) converges to the dominant eigenpair of L with probability 1, provided that the dominant eigenvalue has multiplicity 1 and is strictly bigger than the modulus of every other eigenvalue. The normalization step ensures that the magnitudes of non-dominant modes eventually decay relative to the dominant mode, isolating the dominant eigenpair. While RL-based approaches to learning safety critics rely on artificial discount factors (which can undermine the theoretical validity of the guarantees they provide) to ensure convergence, EigenSafe achieves robust empirical convergence without the need for such modifications.

B. Remarks on Implementation

Empirically, minimizing $\mathcal{J}_{\text{eig}}$ alone is usually sufficient for convergence, regardless of initialization. However, to accelerate training during the initial phase, one may optionally choose to employ the following additional loss term

$$\mathcal{J}_+[\psi] = \frac{W_+}{|\mathcal{D}|} \sum_{(x,u,\cdot)\in\mathcal{D}} \text{ReLU}\left(-\psi(x,u)\right) \tag{20}$$

in addition to $\mathcal{J}_{\text{eig}}$, where W_+ is a tunable nonnegative weight. This term penalizes negative values and thereby biases the optimization toward nonnegative ψ, reducing the influence of sign-oscillatory components associated with non-dominant modes. Once the dominant eigenmode becomes prominent, this auxiliary loss quickly vanishes, not affecting the overall learned result.

In cases where the policy π is only implicitly known through the trajectory dataset itself, and thus the next action u' cannot be actively sampled from $\pi(\cdot|x')$, one can substitute it with

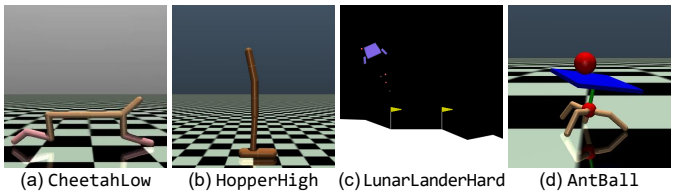


Fig. 3. The Gym environments used in the safe RL demonstration.

the action u' recorded in the dataset trajectory (x, u, x', u'). In this case, γ and ψ will converge to the dominant eigenpair corresponding to the *behavior policy* that generated the dataset.

Finally, consistent with standard practices in deep RL, we employ a semi-gradient approach to enhance training stability. Specifically, during training, we treat the term $\psi(x', u')$ in (18) as a fixed regression target during the optimization step. This is achieved by detaching the calculated values of $\psi(x', u')$ from the computational graph where gradient is propagated, effectively mitigating the moving target problem.

VI. Safe Reinforcement Learning using EigenSafe

In this section, we demonstrate the application of the learned eigenfunction as a safety Q function in an online safe RL framework. We present experimental results obtained from Gym environments [7]. For details regarding the safe RL experiments, the readers are referred to Appendix B.

A. Setup

The objective of this safe RL setting is to find a policy π that maximizes the expected return (the discounted sum of rewards) while satisfying the safety constraint, ensuring that the agent remains within the safe set C with a probability above a specified threshold. To achieve this, we impose a spectral constraint on the policy:

$$\gamma_\pi \geq \gamma_0, \tag{21}$$

where $\gamma_0 \in [0, 1]$ is the target eigenvalue. As γ_π is a global indicator for the safety of π, this informally interprets to: the policy π should be *safer* than the given threshold. Practically, γ_0 should be chosen as exactly 1 or very close to it, as it represents the exponential decay rate of the safety probability. For instance, a policy with $\gamma_\pi = 0.9$ would cause the system to reach an unsafe state in a few dozen timesteps.

We adopt the soft actor-critic (SAC) [23] framework which utilizes a stochastic policy that facilitates exploration. The proposed EigenSafe framework is naturally compatible with the stochastic transitions inherent in SAC. The task critic Q_π is trained via the standard Bellman loss to match the entropy-regularized expected return. Simultaneously, the eigenpair (γ_π, ψ_π) is learned by minimizing the loss functional introduced in (18).

The challenge in training the policy π is that the eigenvalue γ_π is a global property of the operator A_π. It depends on the policy only implicitly and does not directly relate to the local state-action inputs, making gradient-based optimization

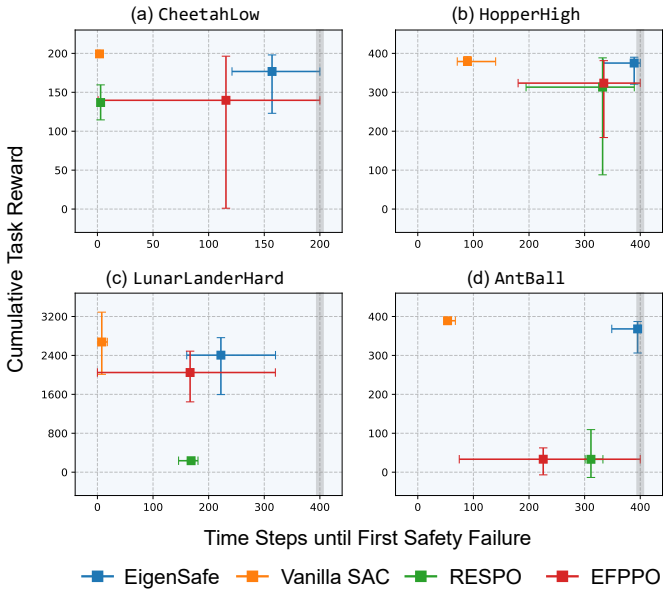


Fig. 4. Baseline comparison results for safe RL. The horizontal axis denotes the number of steps taken until a safety failure or the agent has reached the maximum episode limit, while the vertical axis represents the total reward accumulated throughout the episode, regardless of the safety outcome. EigenSafe consistently appears in the upper-right region across all environments tested, indicating a better balance between reward and safety. Squares indicate mean values, and error bars denote maximum and minimum values over the rollouts. Gray vertical bars indicate the maximum episode length. Each performance is evaluated over five rollout episodes during the final three evaluation epochs, across four training seeds.

intractable. To address this, we replace the global constraint (21) with an equivalent local condition

$$\mathbb{E}_{x' \sim P(\cdot|x,u), u' \sim \pi(\cdot|x')} \psi_\pi(x', u') \geq \gamma_0 \psi_\pi(x, u), \tag{22}$$

which should be satisfied for all $(x, u) \in S \times A$. If a policy π satisfies (22) for all state-action pairs, then the global eigenvalue condition is satisfied, and vice versa.[3] The policy optimization can therefore be formulated as the following constrained problem:

$$\begin{aligned} \max_{\pi} \quad & \mathcal{J}_{\mathrm{RL}}[\pi] := \mathbb{E}_{(x,\cdot,\cdot)\sim\mathcal{D}, u\sim\pi(\cdot|x)} \left[Q_\pi(x, u) \right] \\ \text{s.t.} \quad & \mathbb{E}_{x' \sim P(\cdot|x,u), u' \sim \pi(\cdot|x')} \psi_\pi(x', u') \geq \gamma_0 \psi_\pi(x, u), \end{aligned} \tag{23}$$

where the constraint should be satisfied for all $(x, u, \cdot)$ drawn from the replay buffer $\mathcal{D}$.

We solve this constrained optimization by reformulating its objective into an unconstrained augmented one using the method of Lagrange multipliers [6, 17]. The augmented objective is defined as

$$\begin{aligned} \hat{\mathcal{J}}_{\mathrm{RL}}[\Lambda, \pi] := \mathcal{J}_{\mathrm{RL}}[\pi] + \mathbb{E}_{(x,u,x')\sim\mathcal{D}, u'\sim\pi(\cdot|x')} \big[\\ \Lambda(x, u) \cdot \min \left\{ \epsilon, \psi_\pi(x', u') - \gamma_0 \psi_\pi(x, u) \right\} \big]. \end{aligned} \tag{24}$$

Here, $\Lambda(\cdot, \cdot)$ represents the state- and action-dependent Lagrange multiplier, and $\epsilon > 0$ is a positive but very small constant employed to prevent Λ from exploding towards

[3] This equivalence is the infinite-dimensional analogue of the Collatz-Wielandt formula for positive matrices [41].

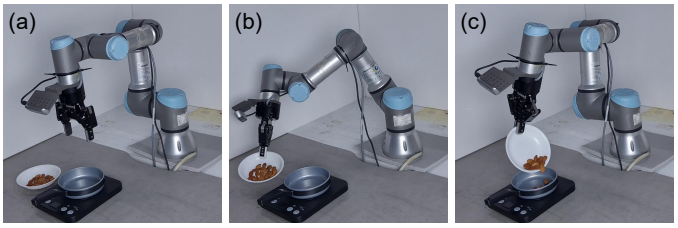


Fig. 5. The task for the hardware experiment in Section VII is to safely pick up the bowl of almonds and pour the almonds into the pan, without spilling them or causing a collision.

positive infinity. The optimization is performed via the dual gradient descent algorithm. It simultaneously updates π and Λ to maximize and minimize, respectively, the augmented objective $\hat{\mathcal{J}}_{\text{RL}}$. This process yields a policy that maximizes the task reward while adhering to the safety constraints.

B. Results

We demonstrate this online safe RL framework in four Gym environments as shown in Fig. 3.

(a) `CheetahLow` is based on the standard `HalfCheetah` environment. The agent receives reward based on its forward velocity only, and the constraint is to maintain its torso below a certain level.

(b) `HopperHigh` is based on the standard `Hopper` environment. The agent receives reward based on its forward velocity only, and the constraint is to maintain its torso above a certain level.

(c) `LunarLanderHard` is a more difficult variant of the standard `LunarLander` environment, with diversified initial reset states. The reward is given only for reaching the goal, and the safety constraints are imposed on landing velocity, body orientation, and horizontal position.

(d) `AntBall` is a modified version of the `Ant` environment. The agent receives reward based on its forward velocity only, and the safety constraint is to not drop the ball from the plate attached to the torso.

We compare with three baselines. Vanilla SAC [23] is included as an unconstrained RL baseline that purely maximizes the reward without accounting for safety constraints. RESPO [19] learns a reachability value function that predicts future constraint violations. The policy is trained through dual gradient descent using Lagrange multipliers. EFPPO [52] instead reformulates the safety-constrained RL problem into an epigraph form. It searches over a cost-budget variable to satisfy constraints, without using Lagrange multipliers. As shown in Fig. 4, the proposed method achieves a better safety-performance balance than the baselines, which we attribute to EigenSafe's improved accuracy of safety assessment.

VII. Safety-Filtered Inference for Imitation Learning using EigenSafe

We demonstrate another practical application of the learned eigenfunction ψ_π, utilizing it as a safety Q function to perform test-time safety filtering on a learned stochastic policy. Detailed information regarding the imitation learning experiment is presented in Appendix C.

Algorithm 1 Safety-Filtered Inference for IL

Input: Pretrained stochastic policy π_{BC}, Pretrained eigenfunction ψ_π, Hyperparameters n, k
for every time step t **do**
  Observe the current state s_t.
  Sample n candidate actions $\{u^{(i)}\}_{i=1}^n \sim \pi_{\text{BC}}(\cdot|s_t)$.
  Evaluate safety scores $v^{(i)} \leftarrow \psi_\pi(s_t, u^{(i)})$ for all i.
  Select action $a_t = u^{(j)}$ such that $v^{(j)}$ is the k-th largest value among the safety scores.
  Execute action a_t.
end for

A. Setup

We consider a standard behavior cloning (BC) setup. The objective is to filter the output of a stochastic BC policy, selecting only those actions deemed relatively safe among multiple candidates. This approach consists of two key components: (a) a learned safety metric (the eigenpair) that evaluates the safety of the chosen action, and (b) an expressive stochastic policy that generates diverse action candidates reflecting the multi-modality of the demonstration data. Unlike the online RL setting where the policy itself is updated, this task focuses on test-time safety improvement. Instead of updating the policy weights, we bias the action sampling process of a pre-trained stochastic policy π_{BC} to favor safer transitions.

To learn the eigenfunction we utilize trajectory tuples (x, u, x', u') consisting of state, action, next state, and next action from the dataset. The EigenSafe loss function (18) is minimized using the next action u' recorded in the dataset, resulting in the learned eigenpair reflecting the safety characteristics of the behavior policy that generated the data. At the same time, the stochastic policy $\pi_{\text{BC}}(\cdot|x)$ is trained using flow matching [36, 38, 45], a generative modeling technique that ensures sampled actions cover the multi-modality of the training data.

During inference, we employ a safety-filtering mechanism to select the most appropriate action from the stochastic policy. At each time step t, given the current state s_t, we sample n candidate actions $\{u^{(i)}\}_{i=1}^n$ from the trained BC policy $\pi_{\text{BC}}(\cdot|s_t)$. We then evaluate the safety score of each candidate using the learned eigenfunction $v^{(i)} = \psi_\pi(s_t, u^{(i)})$. The action corresponding to the k-th largest value among $v^{(i)}$-s is chosen as the input a_t and applied to the system. Here, n and k are tunable parameters. For best performance, we avoid using the highest-value action ($k = 1$) because it increases the risk of out-of-distribution (OOD) actions relative to the behavior policy. Empirically, setting $k \approx n/5$ yields the best performance in both task completion and safety. The overall procedure is summarized in Algorithm 1.

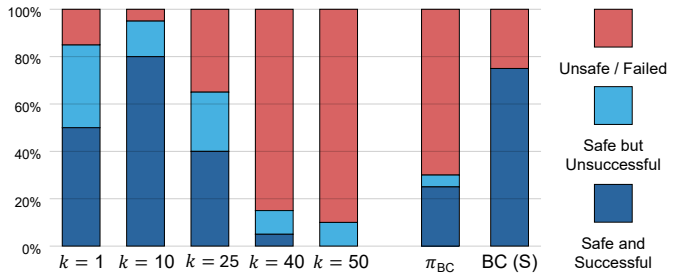


Fig. 6. Success/safety rates in safety-filtered IL, with $n = 50$. Except for the $k = 1$ case, there is a positive correlation between selecting actions with higher ψ_π values and the resulting success and safety rate. Notably, the proposed safety-filtered approach achieves higher safety performance than even the flow BC policy trained only on successful demonstrations. Dark blue bars represent full task success (pouring almonds safely), while light blue indicates episodes that remained safe (no violations) but failed to complete the task. BC (S) refers to the policy trained solely on successful demonstrations. The success and safety rates are estimated by conducting 20 repeated trials for each scenario.

B. Results

We validate the method on a food preparation task depicted in Fig. 5. The objective is to control the UR3 robot equipped with a Robitq 2F-85 gripper to grasp a bowl of almonds and pour them into a pan on a toy stove. Any spill of almonds outside the pan or collision between the robot and the environment is considered a safety violation. The robot's perception consists of two RGB cameras: one attached to the wrist of the arm, and one on an external mount that provides the third-person view of the workspace. The images are processed by a DINOv2 encoder [44] to extract latent features, which are then concatenated with the robot's joint angles and gripper status, forming a 776-dimensional state. The dataset was collected using a GELLO teleoperation device [57]. The dataset comprises 150 successful demonstrations, where the task was completed without safety violations, and 203 failed trajectories, which terminated due to spillage or collision.

In the experiments, we set the number of action candidates $n = 50$ and evaluated performance across different selection of k, by recording the success and safety rates over 20 trials each. We also compare with the naive BC baseline π_{BC} which is trained on the whole dataset, and another BC policy trained only on successful demonstrations. These rates are summarized in Fig. 6. It can be seen that there is a clear positive correlation between the eigenfunction value and both the success and safety rates, with the only exception being the $k = 1$ case, which is affected by the OOD issue discussed above. Notably, the optimal $k = 10$ case exhibits better performance even compared to the BC trained only on successful demonstration, which is not aware of the safety boundary. Fig. 7 visualizes the evolution of ψ_π values for (a) our safety-filtered approach using an optimal rank $k = 10$, and (b) the naive BC baseline. It can be seen from the snapshots that the learned ψ_π serves as an effective safety metric, assigning higher values to safer states. Moreover, it also exhibits *prediction* capabilities. For instance, at point Ⓐ, the eigenfunction value drops significantly upon detecting a precarious grasp, accurately anticipating the eventual failure before it physically occurs.

VIII. Conclusion

A. Summary

This paper presents EigenSafe, a novel theoretical framework for learning safety value functions based on linear operator theory rather than optimal control. We derive a linear Bellman-like operator for dynamic programming for safety probability, whose dominant eigenpair provides a safety measure that is directly tied to the safety probability. In particular, the dominant eigenvalue serves as a scalar metric quantifying the global decay rate of the system's safety probability, while the corresponding dominant eigenfunction serves as a safety Q function that quantifies the relative safety of state-action pairs and provides a directional guidance for safety improvement. This dominant eigenpair can be learned via the proposed loss function inspired by the power iteration algorithm.

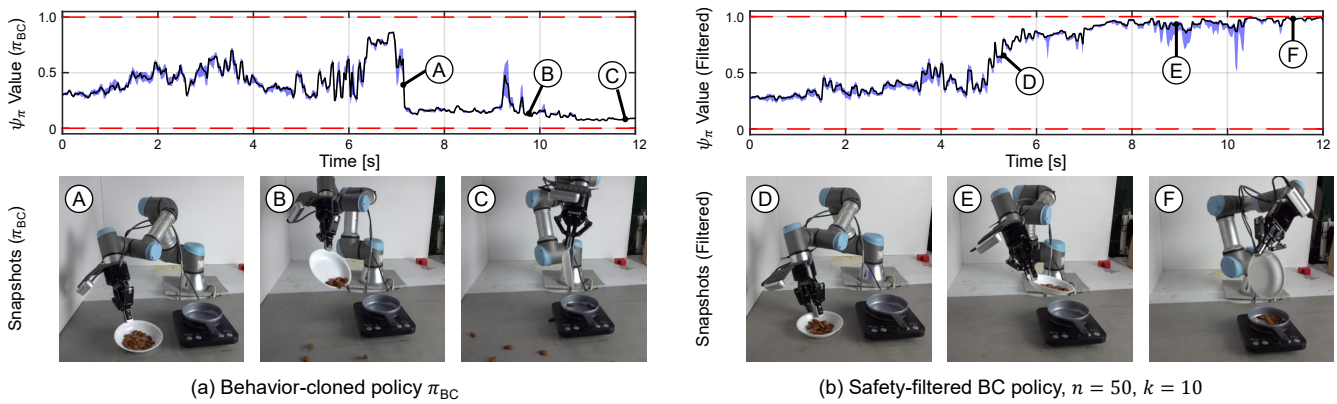
We demonstrate the versatility of EigenSafe in safe RL and IL tasks. For safe RL, we formulate a policy optimization problem that maximizes the expected return (the discounted sum of task rewards) subject to a lower-bound constraint on the dominant eigenvalue. Empirical results across various Gym environments confirm that the approach is effective in achieving both safety compliance and reward maximization. For IL, we propose a test-time safety-filtering technique for a learned stochastic BC policy that evaluates the safety of multiple candidate actions using the learned eigenfunction and executes only those deemed safe. Experiments on a food preparation task using a UR3 robotic arm demonstrate that higher safety and success rates can be achieved with this filtering technique, compared to both naive BC policies and policies trained exclusively on successful demonstrations.

B. Limitations and Future Work

We conclude this paper by summarizing the limitations of the proposed approach and future research directions.

Overestimation Bias in Online Learning. In the online RL setting, where the policy and the eigenfunction are learned simultaneously, EigenSafe may overestimate the safety of certain areas before the policy learning has stabilized. While this is well-known and common in RL [26], it becomes particularly problematic in terms of safety. In the future work, we plan to apply techniques such as [26, 34] to mitigate this issue.

Dependency on Failure Data and Out-of-Distribution Risks. The current framework fundamentally relies on failure labels, i.e., transitions leaving the safe set C. Consequently, the method might struggle to distinguish between the true safe regions and OOD regions where data is scarce, especially in the offline learning setting. Future work will address this by integrating OOD avoidance into the safety constraints, e.g., [10, 31, 50], or by introducing additional penalties near the boundary of the training dataset.



(a) Behavior-cloned policy π_{BC}

(b) Safety-filtered BC policy, $n = 50$, $k = 10$

Fig. 7. Temporal evolution of the eigenfunction ψ_π value during task execution. The blue shaded regions represent the range of eigenfunction values across the sampled action candidates at each time step. The black curves represent that of the chosen action. (a) The naive BC policy π_{BC} leads to a safety violation. It can be seen that the eigenfunction value drops (at Ⓐ) ahead of the actual spillage (see Ⓑ), demonstrating the capability of the eigenfunction to predict future failures. (b) The safety-filtered policy (using $n = 50$ and $k = 10$) consistently selects actions with high safety scores. This ensures a more stable grasp Ⓓ compared to Ⓐ, allowing the robot to convey (Ⓔ) and pour (Ⓕ) the almonds safely.

Future Research Directions. In addition to addressing these limitations, future research will focus on exploiting prior information to enhance sample efficiency and reduce the reliance on failure experiences. We believe one can leverage large foundation models to infer common-sense safety constraints and physical priors, enabling the agent to recognize unsafe areas with less failure experiences. In addition, acknowledging that the definition of safety might vary across domains and may not be fully captured by probability alone, we plan to generalize the framework to more general risk metrics, for example, conditional value at risk, allowing the framework to address different severity of failure modes.

Acknowledgments

The authors would like to thank Prof. Jason J. Choi (UCLA) and Prof. Namhoon Cho (Seoul National University) for insightful discussions, Jonghun Shin, Jusuk Lee, and Seungwon Roh (Seoul National University) for their effort in the hardware experiment setup. This work was supported partially by Samsung Research Funding & Incubation Center of Samsung Electronics under Project Number SRFC-IT2402-17, and partially by the US National Science Foundation's Safe Learning Enabled Systems program and DARPA's Assured Autonomy program. Inkyu Jang received support from the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (RS-2024-00407121).

References

[1] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pages 22–31. PMLR, 2017.

[2] Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2016.

[3] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pages 3420–3431. IEEE, 2019.

[4] Somil Bansal and Claire J Tomlin. DeepReach: A deep learning approach to high-dimensional reachability. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1817–1824. IEEE, 2021.

[5] Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J Tomlin. Hamilton-Jacobi reachability: A brief overview and recent advances. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2242–2253. IEEE, 2017.

[6] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

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

[8] Lukas Brunke, Melissa Greeff, Adam W Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(1):411–444, 2022.

[9] Steven L. Brunton, Marko Budišić, Eurika Kaiser, and J. Nathan Kutz. Modern Koopman theory for dynamical systems. *SIAM Review*, 64(2):229–340, 2022. doi: 10.1137/21M1401243.

[10] Fernando Castañeda, Haruki Nishimura, Rowan Thomas McAllister, Koushil Sreenath, and Adrien Gaidon. In-distribution barrier functions: Self-supervised policy filters that avoid out-of-distribution states. In *The 5th Annual Learning for Dynamics and Control Conference*, volume 211, pages 286–299. PMLR, 15–16 Jun 2023. URL https://proceedings.mlr.press/v211/castaneda23a.html.

[11] Jason Choi, Fernando Castañeda, Claire J Tomlin, and Koushil Sreenath. Reinforcement learning for safety-critical control under model uncertainty, using control lyapunov functions and control barrier functions. In *Robotics: Science and Systems (RSS)*, 2020.

[12] Glen Chou, Dmitry Berenson, and Necmiye Ozay. Learning constraints from demonstrations. In *International Workshop on the Algorithmic Foundations of Robotics*, pages 228–245. Springer, 2018.

[13] Andrew Clark. Control barrier functions for complete and incomplete information stochastic systems. In *2019 American Control Conference (ACC)*, pages 2928–2935. IEEE, 2019.

[14] Ryan Cosner, Preston Culbertson, Andrew Taylor, and Aaron Ames. Robust Safety under Stochastic Uncertainty with Discrete-Time Control Barrier Functions. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.084.

[15] Ryan K. Cosner, Preston Culbertson, and Aaron D. Ames. Bounding stochastic safety: Leveraging freedman's inequality with discrete-time control barrier functions. *IEEE Control Systems Letters*, 8:1937–1942, 2024. doi: 10.1109/LCSYS.2024.3409105.

[16] Charles Dawson, Sicun Gao, and Chuchu Fan. Safe control with learned certificates: A survey of neural Lyapunov, barrier, and contraction methods for robotics and control. *IEEE Transactions on Robotics*, 39(3):1749–1767, 2023. doi: 10.1109/TRO.2022.3232542.

[17] Ben Eysenbach, Russ R Salakhutdinov, and Sergey Levine. Robust predictable control. *Advances in neural information processing systems*, 34:27813–27825, 2021.

[18] Jaime F. Fisac, Neil F. Lugovoy, Vicenç Rubies-Royo, Shromona Ghosh, and Claire J. Tomlin. Bridging Hamilton-Jacobi safety analysis and reinforcement learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8550–8556, 2019. doi: 10.1109/ICRA.2019.8794107.

[19] Milan Ganai, Zheng Gong, Chenning Yu, Sylvia Herbert, and Sicun Gao. Iterative reachability estimation for safe reinforcement learning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 69764–69797, 2023.

[20] Milan Ganai, Sicun Gao, and Sylvia L. Herbert. Hamilton-Jacobi reachability in reinforcement learning: A survey. *IEEE Open Journal of Control Systems*, 3:310–324, 2024. doi: 10.1109/OJCSYS.2024.3449138.

[21] Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

[22] Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, and Alois Knoll. A review of safe reinforcement learning: Methods, theories, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):11216–11235, 2024. doi: 10.1109/TPAMI.2024.3457538.

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

[24] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019.

[25] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025.

[26] Hado Hasselt. Double Q-learning. *Advances in neural information processing systems*, 23, 2010.

[27] Kai-Chieh Hsu, Duy Phuong Nguyen, and Jaime Fernàndez Fisac. Isaacs: Iterative soft adversarial actor-critic for safety. In *The 5th Annual Learning for Dynamics and Control Conference*, volume 211, pages 90–103. PMLR, 15–16 Jun 2023. URL https://proceedings.mlr.press/v211/hsu23a.html.

[28] Subin Huh and Insoon Yang. Safe reinforcement learning for probabilistic reachability and safety specifications: A Lyapunov-based approach. *arXiv preprint arXiv:2002.10126*, 2020.

[29] Pushpak Jagtap, Sadegh Soudjani, and Majid Zamani. Formal synthesis of stochastic systems via control barrier certificates. *IEEE Transactions on Automatic Control*, 66 (7):3097–3110, 2021. doi: 10.1109/TAC.2020.3013916.

[30] Inkyu Jang and H. Jin Kim. Upper bound on escape probability for stochastic control barrier functions. *IEEE Transactions on Automatic Control*, 71(5):3416–3423, 2026. doi: 10.1109/TAC.2025.3638059.

[31] Katie Kang, Paula Gradu, Jason J Choi, Michael Janner, Claire Tomlin, and Sergey Levine. Lyapunov density models: Constraining distribution shift in learning-based control. In *International Conference on Machine Learning*, pages 10708–10733. PMLR, 2022.

[32] Konwoo Kim, Gokul Swamy, Zuxin Liu, Ding Zhao, Sanjiban Choudhury, and Steven Z Wu. Learning shared safety constraints from multi-task demonstrations. *Advances in Neural Information Processing Systems*, 36:5808–5826, 2023.

[33] M. G. Kreĭn and M. A. Rutman. Linear operators leaving invariant a cone in a Banach space. *Amer. Math. Soc. Translation*, 1950(26):128, 1950.

[34] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. *Advances in neural information processing systems*, 33:1179–1191, 2020.

[35] Jingqi Li, Donggun Lee, Jaewon Lee, Kris Shengjun Dong, Somayeh Sojoudi, and Claire Tomlin. Certifiable reachability learning using a new lipschitz continuous value function. *IEEE Robotics and Automation Letters*, 10(4):3582–3589, 2025. doi: 10.1109/LRA.2025.3535183.

[36] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for generative modeling. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.

[37] Simin Liu, Changliu Liu, and John Dolan. Safe control under input limits with neural control barrier functions. In *Proceedings of The 6th Conference on Robot Learning*, volume 205, pages 1970–1980. PMLR, 14–18 Dec 2023.

[38] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.

[39] Kostas Margellos and John Lygeros. Hamilton-Jacobi formulation for reach-avoid differential games. *IEEE Transactions on automatic control*, 56(8):1849–1861, 2011.

[40] Frederik Baymler Mathiesen, Simeon C. Calvert, and Luca Laurenti. Safety certification for stochastic systems via neural barrier functions. *IEEE Control Systems Letters*, 7:973–978, 2023. doi: 10.1109/LCSYS.2022.3229865.

[41] Carl D. Meyer. *Matrix Analysis and Applied Linear Algebra, Second Edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2023. doi: 10.1137/1.9781611977448. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611977448.

[42] Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *The Eleventh International Conference on Learning Representations*, 2023.

[43] Kensuke Nakamura, Lasse Peters, and Andrea Bajcsy. Generalizing safety beyond collision-avoidance via latent-space reachability analysis. In *Robotics: Science and Systems (RSS)*, 2025.

[44] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[45] Seohong Park, Qiyang Li, and Sergey Levine. Flow Q-learning. In *International Conference on Machine Learning (ICML)*, 2025.

[46] Maria Prandini, Jianghai Hu, C Cassandras, and J Lygeros. Stochastic reachability: Theory and numerical approximation. *Stochastic hybrid systems, Automation and Control Engineering Series*, 24:107–138, 2006.

[47] Zengyi Qin, Dawei Sun, and Chuchu Fan. Sablas: Learning safe control for black-box dynamical systems. *IEEE Robotics and Automation Letters*, 7(2):1928–1935, 2022.

[48] Alexander Robey, Haimin Hu, Lars Lindemann, Hanwen Zhang, Dimos V. Dimarogonas, Stephen Tu, and Nikolai Matni. Learning control barrier functions from expert demonstrations. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 3717–3724, 2020. doi: 10.1109/CDC42340.2020.9303785.

[49] Cesar Santoyo, Maxence Dutreix, and Samuel Coogan. A barrier function approach to finite-time stochastic system verification and control. *Automatica*, 125:109439, 2021.

[50] Junwon Seo, Kensuke Nakamura, and Andrea Bajcsy. Uncertainty-aware latent safety filters for avoiding out-of-distribution failures. *Conference on Robot Learning (CoRL)*, 2025.

[51] Lu Shi, Masih Haseli, Giorgos Mamakoukas, Daniel Bruder, Ian Abraham, Todd Murphey, Jorge Cortés, and Konstantinos Karydis. Koopman operators in robot learning. *IEEE Transactions on Robotics*, pages 1–20, 2026. doi: 10.1109/TRO.2026.3654384.

[52] Oswin So and Chuchu Fan. Solving Stabilize-Avoid via Epigraph Form Optimal Control using Deep Reinforcement Learning. In *Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.085.

[53] Oswin So, Zachary Serlin, Makai Mann, Jake Gonzales, Kwesi Rutledge, Nicholas Roy, and Chuchu Fan. How to train your neural control barrier function: Learning safety filters for complex input-constrained systems. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11532–11539. IEEE, 2024.

[54] Jacob Steinhardt and Russ Tedrake. Finite-time regional verification of stochastic non-linear systems. *The International Journal of Robotics Research*, 31(7):901–923, 2012.

[55] Abraham P Vinod and Meeko MK Oishi. Stochastic reachability of a target tube: Theory and computation. *Automatica*, 125:109458, 2021.

[56] Kim P. Wabersich, Andrew J. Taylor, Jason J. Choi, Koushil Sreenath, Claire J. Tomlin, Aaron D. Ames, and Melanie N. Zeilinger. Data-driven safety filters: Hamilton-Jacobi reachability, control barrier functions, and predictive methods for uncertain systems. *IEEE Control Systems Magazine*, 43(5):137–177, 2023. doi: 10.1109/MCS.2023.3291885.

[57] Philipp Wu, Yide Shentu, Zhongke Yi, Xingyu Lin, and Pieter Abbeel. GELLO: A general, low-cost, and intuitive teleoperation framework for robot manipulators. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 12156–12163. IEEE, 2024.

[58] Shakiba Yaghoubi, Keyvan Majd, Georgios Fainekos, Tomoya Yamaguchi, Danil Prokhorov, and Bardh Hoxha. Risk-bounded control using stochastic barrier functions. *IEEE Control Systems Letters*, 5(5):1831–1836, 2020.

[59] Songyuan Zhang, Oswin So, Mitchell Black, Zachary Serlin, and Chuchu Fan. Solving multi-agent safe optimal control with distributed epigraph form MARL. In *Proceedings of Robotics: Science and Systems*, 2025.

[60] Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.