ISAACS: Iterative Soft Adversarial Actor-Critic for Safety

Kai-Chieh Hsu*   KAICHIEH@PRINCETON.EDU

Duy P. Nguyen*   DUYN@PRINCETON.EDU

Jaime F. Fisac   JFISAC@PRINCETON.EDU

Department of Electrical and Computer Engineering, Princeton University, NJ, USA

Abstract

The deployment of robots in uncontrolled environments requires them to operate robustly under previously unseen scenarios, like irregular terrain and wind conditions. Unfortunately, while rigorous safety frameworks from robust optimal control theory scale poorly to high-dimensional nonlinear dynamics, control policies computed by more tractable "deep" methods lack guarantees and tend to exhibit little robustness to uncertain operating conditions. This work introduces a novel approach enabling scalable synthesis of robust safety-preserving controllers for robotic systems with general nonlinear dynamics subject to bounded modeling error, by combining game-theoretic safety analysis with adversarial reinforcement learning in simulation. Following a soft actor-critic scheme, a safety-seeking fallback policy is co-trained with an adversarial "disturbance" agent that aims to invoke the worst-case realization of model error and training-to-deployment discrepancy allowed by the designer's uncertainty. While the learned control policy does not intrinsically guarantee safety, it is used to construct a real-time safety filter with robust safety guarantees based on forward reachability rollouts. This safety filter can be used in conjunction with a safety-agnostic control policy, precluding any task-driven actions that *could* result in loss of safety. We evaluate our learning-based safety approach in a 5D race car simulator, compare the learned safety policy to the numerically obtained optimal solution, and empirically validate the robust safety guarantee of our proposed safety filter against worst-case model discrepancy.

Keywords: Adversarial Reinforcement Learning, Model Predictive Safety Filter, Hamilton Jacobi Reachability Analysis

1. Introduction

Recent years have seen a rapid increase in the deployment of robotic systems beyond their traditional industrial settings, with emerging applications including home robots, autonomous driving, and a range of drone services. These new opportunities are tied to open, uncontrolled environments, where safe robot operation is at once critical and hard to ensure. Safety guarantees in these open-world settings face the coupled challenges of *scalability* and *robustness*. Many modern robotic systems present high-order nonlinear dynamics, making safety analysis computationally demanding. Even when the analysis is tractable (usually for lower-fidelity models), discrepancies between the modeled and physical system can result in degraded performance and even catastrophic failures.

To ensure safe autonomous operation, a range of engineering efforts seek to automatically *filter* robots' task-oriented control policies to preclude unsafe actions. One important family of methods is built on Hamilton-Jacobi (HJ) reachability analysis, formulating a zero-sum dynamic game where the robot's *controller* aims to keep the state away from all known failure conditions despite the
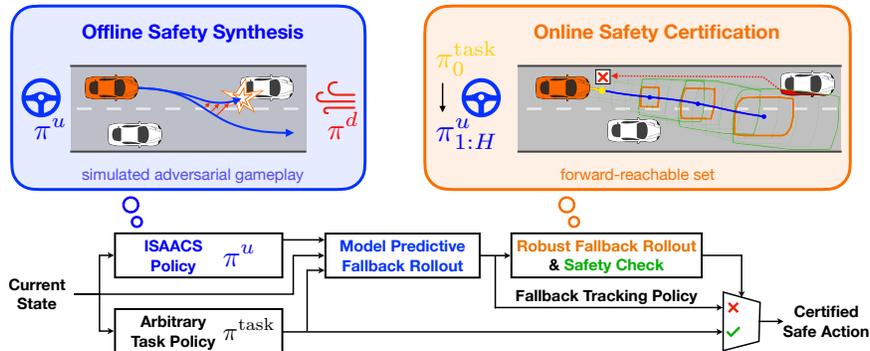
---

* Equal contribution.

Figure 1: ISAACS is a game-theoretic reinforcement learning scheme whose best-effort learned safety policy can be converted into effective robust safety-certified strategies at runtime. *Offline Safety Synthesis:* adversarial reinforcement learning approximately solves the robust safety problem, jointly training the safety policy π^u and worst-case disturbance π^d. *Online Safety Certification:* the learned safety policy is rolled out under all disturbance realizations. Here, the forward-reachable sets (orange) are safe if their footprint-augmented counterparts (green) remain collision-free. *Robust Safety Filter:* control actions proposed by an arbitrary task policy π^{task} are allowed if a subsequent safety policy rollout ("fallback") is certified safe; otherwise, the (already certified) fallback tracking policy is used.

adversarial inputs of a bounded *disturbance* representing unknown model error and exogenous perturbations (Mitchell et al., 2005). While powerful, HJ methods become computationally prohibitive beyond 5 state dimensions (Bansal et al., 2017). Recent research with neural representations shows promise in scaling safety analysis to high-dimensional systems (Fisac et al., 2019; Bansal and Tomlin, 2021; Hsu et al., 2023). Unfortunately, the learned policy and value function may not be accurate everywhere, and therefore carry a risk of catastrophic outcomes if directly relied upon for safety.

This paper introduces Iterative Soft Adversarial Actor-Critic for Safety (ISAACS), a novel game-theoretic reinforcement learning (RL) scheme for *approximate safety analysis*, whose outputs can be efficiently converted at runtime into *robust safety-certified control strategies* (Figure 1). ISAACS is first used in an *offline synthesis* stage that jointly trains a best-effort safety controller and a worst-case failure-seeking disturbance through many iterations of simulated zero-sum gameplay (Silver et al., 2017; Pinto et al., 2017). The learned control policy can then be treated as an "untrusted oracle" and used in *online safety certification* by guiding a robust predictive rollout that accounts for all admissible realizations of model uncertainty. This "rollout check" enables a *recursively safe* runtime control filter that preemptively overrides any candidate control action that could otherwise drive the state into an unrecoverable configuration.

We demonstrate this framework on a dynamical system at the boundary of computability for numerical dynamic programming (Bui et al., 2022), namely a 5-dimensional race car simulator. Rather than substantiating scalability (already established in prior work, cf. Haarnoja et al. (2018); Fisac et al. (2019)), we focus on showing the framework's ability to synthesize provably correct robust safety filters using deep reinforcement learning, and use the still-tractable numerical 5-D solution as a reference oracle. In our experiments, the ISAACS robust rollout safety filter maintained a perfect (zero violation) safety rate under the oracle worst-case disturbance, with moderate conservativeness relative to the optimal Hamilton-Jacobi solution. Using the ISAACS model directly performed well for the most part, but it lacked theoretical guarantees and the safety rate was indeed nonzero.[1]

---

1. See https://saferobotics.princeton.edu/research/isaacs/ for supplementary material.

1.1. Related Work

Safety guarantees in robotics have their origins in robust control. Robust "tube" model predictive control (MPC) approaches (Langson et al., 2004) allow enforcing state constraints like collision avoidance in the presence of bounded uncertainty. Hamilton-Jacobi-Isaacs (HJI) theory handles general nonlinear dynamics and control objectives by posing the problem as a two-player zero-sum differential game between the controller and an adversarial disturbance (Isaacs, 1954; Mitchell et al., 2005; Bansal et al., 2017). Similar approaches pose the game in temporal logic form using formal methods (Mattila et al., 2015; Alpern and Schneider, 1985). While these techniques enjoy strong theoretical properties, their use in robotics is limited by their poor computational scalability.

Control barrier function (CBF) approaches aim to circumvent numerical intractability by finding a smaller (more conservative) invariant set that can be encoded in closed form (e.g., sum-of-squares) (Ames et al., 2014, 2019); unfortunately there are no systematic constructive methods, so system-specific hand design is often needed (Nguyen and Sreenath, 2016; Squires et al., 2018). Further, CBF guarantees have limited robustness to disturbances (Xu et al., 2015) and may be lost entirely if the robot reaches its actuation limits (Zeng et al., 2021; Choi et al., 2021).

Finally, deep self-supervised learning (Bansal and Tomlin, 2021) and reinforcement learning (Fisac et al., 2019; Bharadhwaj et al., 2021; Thananjeyan et al., 2021; Hsu et al., 2021) can synthesize approximate control policies and value functions ("safety critics"), but offer no intrinsic safety guarantees, and robustness to modeling and learning error is not yet well understood. Domain randomization (DR) approaches (Tobin et al., 2017; Mehta et al., 2020) conduct training under a family of environments with randomly sampled parameter values, targeting expected performance over the hypothesized parameter distribution, whereas adversarial training approaches formulate a zero-sum game and jointly train the control policy and the worst-case environment realization via simulated gameplay (Pinto et al., 2017). While most closely aligned with Fisac et al. (2019); Hsu et al. (2021), our work introduces robustness through an adversarial reinforcement learning scheme grounded in the game-theoretic HJI formulation and further recovers robust guarantees by rolling out the learned policies in an online receding horizon framework.

2. Preliminaries

2.1. Hamilton-Jacobi-Isaacs Reachability Analysis and Safety Filters

We consider fully observable robotic system governed by discrete-time dynamics with unknown but bounded model error:

$$x_{t+1} = f(x_t, u_t, d_t), \qquad (1)$$

where, at each time step $t \in \mathbb{N}$, $x_t \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$ is the state, $u_t \in \mathcal{U} \subset \mathbb{R}^{n_u}$ is the *controller* input, and $d_t \in \mathcal{D} \subset \mathbb{R}^{n_d}$ is the *disturbance* input, unknown *a priori*. We assume we are given a specification of the *failure set* $\mathcal{F} \subseteq \mathcal{X}$ that the system must be prevented from entering. By convention, we assume $\mathcal{F}$ to be open. Safety analysis seeks to determine the *safe set* $\Omega \subseteq \mathcal{X}$, consisting of all initial states from which there exists a control policy that can keep the system away from the failure set at all times for *any* realization of the uncertainty:

$$\Omega := \left\{ x \in \mathcal{X} \mid \exists \pi^u : \mathcal{X} \to \mathcal{U}, \ \forall \pi^d : \mathcal{X} \to \mathcal{D}, \ \forall t > 0, \ \mathbf{x}_x^{\pi^u, \pi^d}(t) \notin \mathcal{F} \right\}, \qquad (2)$$

where $\mathbf{x}_x^{\pi^u, \pi^d} : \mathbb{N} \to \mathcal{X}$ denotes the trajectory starting from state $x_0 = x$ following dynamics (1)

under control policy π^u and disturbance policy π^d. Note that the order of quantifiers in (2) is crucial: there must be one (same) control policy π^u that maintains safety under all disturbance policies π^d.

Hamilton-Jacobi-Isaacs (HJI) reachability analysis formulates the robust safety problem as a zero-sum game between the controller and the disturbance, introducing a Lipschitz continuous *safety margin* $g \colon \mathcal{X} \to \mathbb{R}, g(x) < 0 \iff x \in \bar{\mathcal{F}}$ to further transform the safety "game of kind" with a binary outcome (whether the system enters $\mathcal{F}$) into a "game of degree" with continuous payoff

$$J^{\pi^u, \pi^d}(x) := \min_{t \in \mathbb{N}} g\Big(\mathbf{x}_x^{\pi^u, \pi^d}(t)\Big). \tag{3}$$

Consistent with the order of quantifiers in (2), which gives the disturbance the instantaneous informational advantage (Isaacs, 1954), we define the lower value function of the safety game as $V(x) := \max_{\pi^u} \min_{\pi^d} J^{\pi^u, \pi^d}(x)$, which encodes the minimal safety margin g that our controller can maintain at all times under the worst-case disturbance. This value $V(x)$ satisfies the two-player dynamic programming Isaacs equation

$$V(x) = \max_u \min_d \min \Big\{ g(x),\, V\big(f(x, u, d)\big) \Big\}. \tag{4}$$

If the value function can be computed, the safe set (2) can be obtained by $V(x) \geq 0 \iff x \in \Omega$, and the optimal policies $\pi^{u*}(x)$, $\pi^{d*}(x)$ are given by the optimizers of (4) at each state x. We can then *filter* an arbitrary task-oriented policy π^{task} through the least-restrictive law (Fisac et al., 2019):

$$\phi(x; \pi^{\text{task}}) = \begin{cases} \pi^{\text{task}}(x), & V(x) \geq \epsilon \\ \pi^{u*}(x), & \text{otherwise} \end{cases} \tag{5}$$

where $\epsilon \geq 0$ is the value threshold, typically chosen slightly larger than zero to account for numerical errors and control delays. The safety filter ϕ enforces an important invariance property: from any state in the safe set Ω the system is guaranteed to remain in the safe set perpetually.

2.2. Reachability Analysis through Reinforcement Learning

Level-set methods solve for the HJI value function in (4) with vanishing approximation error as the grid resolution increases. However, the memory and computation complexity grows exponentially with the state dimension, which limits practical applicability to dynamical systems with at most 6 continuous state dimensions (Bui et al., 2022). Fisac et al. (2019) use reinforcement learning algorithms to more tractably find approximate solutions to high-dimensional reachability problems (demonstrated on up to 18 state dimensions) by replacing the usual reinforcement learning Bellman equation for a cumulative reward with the time-discounted (single-player) counterpart of (4):

$$V_\omega(x) = (1 - \gamma)g(x) + \gamma \min \Big\{ g(x),\, \max_{u \in \mathcal{U}} Q_\omega(x, u) \Big\}, \quad Q_\omega(x, u) := V_\omega\big(\bar{f}(x, u)\big), \tag{6}$$

where the nominal dynamics $\bar{f} \colon \mathcal{X} \times \mathcal{U} \to \mathcal{X}$, which can be seen as a special case of f in (1) assuming no modeling error ($d = 0$); $\gamma \in (0, 1)$ is the time discount rate for future safety margins; and Q_ω is the state-action safety value function, parameterized by ω.[2] An analogous safety filter to (5) can then be constructed by replacing V with the learned V_ω and $\pi^{u*}(x)$ with $\operatorname{argmax}_{u \in \mathcal{U}} Q_\omega(x, u)$. Unfortunately, due to the approximate nature of Q_ω and V_ω, this is only a *best-effort* safety filter: unlike (5), it comes with no invariance guarantees and it cannot generally prevent safety violations.

---

2. In the deep reinforcement learning literature, ω are neural network weights and Q_ω is called a Q-network or critic.

3. ISAACS: Iterative Soft Adversarial Actor-Critic for Safety

To harness the scalability of neural representations without renouncing the robust safety guarantees of model-based analysis, we propose Iterative Soft Adversarial Actor-Critic for Safety (ISAACS), a game-theoretic reinforcement learning scheme that approximates the HJI solution to a reachability game and learns a safety policy that can be used to construct a provably safe runtime control strategy. ISAACS uses repeated rounds of simulated zero-sum gameplay to jointly train a safety control policy and a failure-seeking disturbance policy, consistent with the Isaacs equation (4). Once trained, the ISAACS safety policy can be used as the receding-horizon reference for a robust fallback control strategy, defining a compact forward-reachable set that can be checked for safety violations. This results in a recursive safety filter with an equivalent invariance property to the accurate but less scalable counterpart (5) enabled by numerical HJI methods.

3.1. Adversarial Actor-Critic Reinforcement Learning for Safety Policy Synthesis

Analogous to the single-player reachability reinforcement learning formulation of Fisac et al. (2019), we consider a time-discounted counterpart of the reachability payoff (3). In this case, however, we have a zero-sum game whose value function is characterized by a two-player Isaacs equation (rather than a Bellman equation). In the space of soft actor-critic policies, the Isaacs equation can be written

$$V(x) = (1-\gamma)g(x) + \gamma \max_{\pi^u} \min_{\pi^d} \mathbb{E}_{u,d} \min\Big\{g(x), Q(x,u,d)\Big\}, \quad Q(x,u,d) := V\big(f(x,u,d)\big) \quad (7)$$

where π^u, π^d are *stochastic* controller and disturbance policies, and $u \sim \pi^u(\cdot \mid x), d \sim \pi^d(\cdot \mid x)$. Note that as the time discount factor $\gamma \in [0,1)$ goes to 1, we recover the undiscounted problem (4).

The ISAACS offline synthesis scheme solves the Isaacs equation (7) approximately by training three neural networks, with parameters ω, θ, ϕ, that encode a critic, a control policy, and a disturbance policy, respectively. The learning scheme updates the critic and disturbance policy $\tau \in \mathbb{N}$ more often than the control policy. This effectively makes the disturbance policy a *follower* to the control policy (Zrnic et al., 2021)—thereby maintaining the disturbance's informational advantage

---

Algorithm 1 ISAACS: Iterative Soft Adversarial Actor-Critic for Safety (Offline Safety Synthesis)

1: **for** each tournament round **do**
2:     **for** each episode **do**
3:         $x_0 \sim P_0, \tilde{\pi}^d \sim P_{\Pi^d}$ ▷ Sample initial state and disturbance policy for this episode
4:         **for** each time step **do**
5:             $u_t \sim \pi_\theta^u(\cdot|x_t), d_t \sim \tilde{\pi}^d(\cdot|x_t)$ ▷ Sample control and disturbance from policies
6:             $x_{t+1} = f(x_t, u_t, d_t)$ ▷ Get transition from the environment
7:             $\mathcal{B} \leftarrow \mathcal{B} \cup \big\{\big(x_t, u_t, d_t, x_{t+1}, g(x_{t+1})\big)\big\}$ ▷ Store the transition in the replay buffer
8:         **for** each gradient step **do**
9:             $\omega \leftarrow \omega - \lambda_\omega \nabla_\omega L(\omega)$ ▷ Update critic parameters
10:             $\omega' \leftarrow \lambda_{\omega'}\omega + (1 - \lambda_{\omega'})\omega'$ ▷ Update target critic parameters
11:             $\phi \leftarrow \lambda_\phi \nabla_\phi L(\phi)$ ▷ Update disturbance policy parameters
12:             **if** gradient step is a multiple of τ **then**
13:                 $\theta \leftarrow \lambda_\theta \nabla_\theta L(\theta)$ ▷ Update safety policy parameters at a slower rate
14:     Update leaderboard $\Pi^u \bigcup\{\pi_\theta^u\}$ vs. $\Pi^d \bigcup\{\pi_\phi^d\}$ and keep best k^u in Π^u, best k^d in Π^d
15:     $m \leftarrow (m_1, \cdots, m_{|\Pi^d|})$ ▷ m_i is the total win rate of π_i^d across all $\tilde{\pi}^u \in \Pi^u$
16:     $P_{\Pi^d} \leftarrow \text{softmax}(m)$ ▷ Update disturbance policy distribution

---

from (2) and (7)—and has the advantage of optimizing against a static target within each update epoch of the controller's policy.

We additionally maintain a finite *leaderboard* of controller and disturbance policies from past stages of training, $\Pi^j = \{\pi_1^j, ...\pi_{k^j}^j\}$, $j \in \{u, d\}$. At the start of each episode, ISAACS samples an initial state from a preset distribution P_0 and selects a disturbance policy $\tilde{\pi}^d$ from Π^d and simulates the gameplay between π_θ^u and the sampled $\tilde{\pi}^d$, which discourages the control policy updates from overfitting to a single disturbance policy (Vinitsky et al., 2020). Periodically during training, the leaderboard is updated by incorporating the current controller and disturbance policies into Π^u, Π^d and simulating multiple gameplay episodes for each new pair of controller-disturbance policies, recording the fraction of episodes that result in safety failures. Policies in Π^u are ranked based on their overall win rate against all opponent policies in Π^d, and vice versa, and the worst-performing one is dropped from each leaderboard (if the total count exceeds the preset capacity k^u, k^d).

We update all neural networks based on Soft Actor-Critic (SAC) (Haarnoja et al., 2018). At every time step we store the transition (x, u, d, x', g') in the replay buffer $\mathcal{B}$, with $x' = f(x, u, d)$ and $g' = g(x')$. We update the critic to reduce the deviation from the Isaacs target (7),[3]

$$L(\omega) := \underset{(x,u,d,x',g')\sim\mathcal{B}}{\mathbb{E}}\left[(Q_\omega(x, u, d) - y)^2\right], \quad y = (1-\gamma)g' + \gamma\min\{g', \ Q_{\omega'}(x', u', d')\}, \quad \text{(8a)}$$

with $u' \sim \pi_\theta^u(\cdot \mid x')$, $d' \sim \pi_\phi^d(\cdot \mid x')$. We update both policies following the policy gradient induced by the critic and entropy loss terms:

$$L(\theta) := \underset{(x,d)\sim\mathcal{B}}{\mathbb{E}}\left[-Q_\omega(x, \tilde{u}, d) + \alpha_u\log\pi_\theta^u(\tilde{u}|x)\right], \quad L(\phi) := \underset{(x,u)\sim\mathcal{B}}{\mathbb{E}}\left[Q_\omega(x, u, \tilde{d}) + \alpha_d\log\pi_\phi^d(\tilde{d}|x)\right], \quad \text{(8b)}$$

where $\tilde{u} \sim \pi_\theta^u(\cdot \mid x)$, $\tilde{d} \sim \pi_\phi^d(\cdot \mid x)$, and α_u, α_d are hyperparameters encouraging higher entropy in the stochastic policies (more exploration), which decay gradually in magnitude through the ISAACS training. We summarize the ISAACS scheme in Algorithm 1 (where $\lambda_\omega, \lambda_{\omega'}, \lambda_\theta, \lambda_\phi$ are the learning rate hyperparameters for $\omega, \omega', \theta, \phi$).

3.2. Runtime Safety Filter through Robust Policy Rollout

It may seem tempting to use the value and policies computed by ISAACS directly in an online safety solution. While a learned value-based safety filter in the form of (5) can work well in practice (Hsu et al., 2023), it comes without guarantees, and it may not always prevent catastrophic failures. A similar issue arises when directly rolling out the learned controller and disturbance policies at runtime and checking the resulting trajectory for future collisions: while a suboptimal controller policy would merely result in a more conservative filter, a suboptimal disturbance policy may lead us to erroneously conclude that a state is safe, when in reality there exists a different uncertainty realization that could drive the state to the failure set. To obtain a robust safety guarantee under possibly suboptimal ISAACS learning, we therefore treat the controller and the disturbance differently. The learned controller policy is used as a best-effort "untrusted oracle" to obtain a reference rollout trajectory; conversely, for the disturbance, we consider *all* possible inputs within the bounded set $\mathcal{D}$, which induce a forward-reachable set (FRS) containing a continuum of possible futures.

---

3. Deep RL usually trains an auxiliary target critic $Q_{\omega'}$, whose parameters ω' change slowly to match the critic parameters ω to stabilize the regression (a fixed target in a short period of time).

At each control cycle, given a proposed control from the task policy, we start by rolling out a nominal reference trajectory with zero disturbance input. Similar to other model predictive safety filtering approaches Bastani and Li (2021), we simulate a single step using the proposed control and subsequently switch to the trained safety policy for the remaining H steps:

$$x_{\tau+1|t} = f(x_{\tau|t}, u_{\tau|t}, 0), \qquad u_{\tau|t} = \begin{cases} \pi^{\text{task}}\left(x_{\tau|t}\right), & \tau = 0 \\ \pi^u_\theta\left(x_{\tau|t}\right), & \tau \in \{1, \ldots, H\}, \end{cases} \qquad x_{0|t} = x_t, \quad (9)$$

where $(\cdot)_{\tau|t}$ denotes variables at step τ of a plan computed at time t. The blue polyline in Figure 1 (*right*) shows the nominal trajectory. Since the neural network safety policy is not guaranteed to be stabilizing, we utilize the time-varying linear quadratic regulator (LQR) approach to derive (local) linear tracking policies for the time horizon H.

To compute the tracking policy, we first linearize the dynamics around the nominal trajectory $\delta x_{\tau+1} = f_{x,\tau} \delta x_\tau + f_{u,\tau} \delta u_\tau + f_{d,\tau} d_{t+\tau} + e_\tau$, at each $\tau \in \{0, \ldots, H\}$, where $\delta x_\tau := x_{t+\tau} - x_{\tau|t}$, $\delta u_\tau := u_{t+\tau} - u_{\tau|t}$, and $f_{x,\tau}, f_{u,\tau}, f_{d,\tau}$ are the Jacobians of the dynamics evaluated at prediction step τ, with e_τ denoting the associated linear approximation error ("Taylor remainder") at that time. Using time-varying LQR, we compute the *fallback tracking policy* $\delta u_\tau = K_{\tau|t} \delta x_\tau, \tau \in \{1, \ldots, H\}$, which aims to efficiently track the nominal rollout trajectory from (9). The closed-loop linear error dynamics under this policy (with the unknown disturbance as the only exogenous input) are

$$\delta x_{\tau+1} = A_\tau \delta x_\tau + B_\tau d_{t+\tau} + e_\tau, \ \tau \in \{1, \ldots, H\}, \tag{10}$$

where $A_\tau := f_{x,\tau} + f_{u,\tau} K_{\tau|t}$ and $B_\tau := f_{d,\tau}$. Similarly letting $B_0 := f_{d,0}$, the forward-reachable set containing all possible (under $\mathcal{D}$) tracking errors δx_τ at each step τ can be computed by

$$\mathcal{R}_{\tau+1} = A_\tau \mathcal{R}_\tau \oplus B_\tau \mathcal{D} \oplus \mathcal{E}_\tau, \quad \tau \in \{1, \ldots, H\}, \qquad\qquad \mathcal{R}_1 = B_0 \mathcal{D}, \tag{11}$$

where $\mathcal{E}_\tau$ is the bounding box for the Taylor remainder at time step τ, and $\oplus$ denotes the Minkowski sum of two sets. The orange polytopes in Figure 1 (right) show the computed forward-reachable sets, and the green polytopes show their footprint-augmented counterparts.

Using the tracking error bounds $\mathcal{R}_\tau$, we define the robust rollout-based safety filter criterion:

$$\Delta_\mathcal{R}(x_t, \pi^{\text{task}}, H) := \mathbb{1}\left\{\{x_{\tau+1|t}\} \oplus \mathcal{R}_{\tau+1} \cap \mathcal{F} = \emptyset \wedge \{u_{\tau|t}\} \oplus K_\tau \mathcal{R}_\tau \subseteq \mathcal{U}, \forall \tau \in \{0, \cdots, H\}\right\}. \tag{12}$$

Precisely, $\Delta_\mathcal{R}(x_t, \pi^{\text{task}}, H) = 1$ means that after applying the proposed control from the task policy, the tracking policy $(K_{\tau|t})_{\tau=1}^H$ can maintain safety for the H subsequent steps under all possible uncertainty realizations, i.e. for any disturbance sequence satisfying $d_{t+\tau} \in \mathcal{D}$, without exceeding the control bound $\mathcal{U}$.[4] In other words, a *robust safety fallback strategy* is available after applying the proposed control. The corresponding safety filter can be constructed as follows:

$$\phi(x_{t+\tau}; \Delta_\mathcal{R}, t) = \begin{cases} \pi^{\text{task}}(x_{t+\tau}), & \Delta_\mathcal{R}(x_{t+\tau}, \pi^{\text{task}}, H) = 1 \\ K_{\tau|t}(x_{t+\tau} - x_{\tau|t}), & \Delta_\mathcal{R}(x_{t+\tau}, \pi^{\text{task}}, H) = 0 \wedge \tau \in \{1, \cdots, H\} \\ \pi^u_\theta(x_{t+\tau}), & \text{otherwise}, \end{cases} \tag{13}$$

where t is the last time step that the safety filter criterion holds, i.e., $\Delta_\mathcal{R}(x_t, \pi^{\text{task}}, H) = 1$. We then have the following finite-horizon safety theorem, which is recursively enforceable by applying the safety filter with robust rollout-based criterion in (12).

---

4. In practice, the control condition in (12) can be enforced during the rollout of $\mathcal{R}_\tau$ by scaling down $K_{\tau|t}$ as needed.

Theorem 1 (Finite-Horizon Safety) *If the initial state x_t satisfies $\Delta_{\mathcal{R}}(x_t, \pi_\theta^u, H) = 1$, the safety filter $\phi(\cdot; \Delta_{\mathcal{R}}, t)$ in (13) keeps the feedback system safe under the disturbance set $\mathcal{D}$ for at least $H + 1$ steps, i.e., $x_{t+\tau+1} = f(x_{t+\tau}, \phi(x_{t+\tau}; \Delta_{\mathcal{R}}, t), d_{t+\tau}) \notin \mathcal{F}, \forall d_{t+\tau} \in \mathcal{D}, \forall \tau \in \{0, \cdots, H\}$.*

Proof Whenever $\Delta_{\mathcal{R}}(x_t, \pi^{\text{task}}, H) = 1$, a robust H-step fallback tracking policy is constructed and found to keep $\{x_{\tau|t}\} \oplus \mathcal{R}_\tau$ disjoint from $\mathcal{F}$ for $\tau \in \{1, \ldots, H+1\}$. By construction, after applying $\pi^{\text{task}}(x_t), \forall d_t \in \mathcal{D}, x_{t+1} \in \{x_{1|t}\} \oplus \mathcal{R}_1$. Suppose that all subsequent checks fail: $\Delta_{\mathcal{R}}(x_{t+\tau}, \pi^{\text{task}}, H) = 0, \forall \tau \geq 1$, then the filter (13) applies the fallback policy, ensuring $\forall d_{t+\tau} \in \mathcal{D}, x_{t+\tau+1} \in \{x_{\tau+1|t}\} \oplus \mathcal{R}_{\tau+1}$ up until the last computed set $\mathcal{R}_{H+1}$. Therefore, $x_{t+1}, \ldots, x_{t+H+1} \notin \mathcal{F}$. If at any time $\tau \leq t+H+1, \Delta_{\mathcal{R}}(x_\tau, \pi^{\text{task}}, H) = 1$, a new fallback tracking policy is computed and the guarantee resets for another $H + 1$ steps. ∎

Remark 2 *If we further assume a robust controlled-invariant set $\mathcal{T} \subseteq \mathcal{X}, \mathcal{T} \cap \mathcal{F} = \emptyset$ under policy $\pi^{\mathcal{T}}$, the guarantee can be extended to the infinite horizon, by adding an additional condition that the system robustly reach $\mathcal{T}$ within the rollout horizon, i.e., $\mathcal{R}_\tau \oplus \{x_{\tau|t}\} \subseteq \mathcal{T}$ for some $\tau \leq H+1$. After this, the controller can always continue to apply $\pi_{\mathcal{T}}$ to remain in $\mathcal{T}$ and thus out of $\mathcal{F}$. Alternatively, more sophisticated robust certification methods (Wabersich and Zeilinger, 2018) may be used.*

4. Experimental Evaluation

4.1. Implementation Details

Environment We demonstrate our framework in a straight-road environment. We consider the uncertain dynamics of a small robot car modified from a 5D kinematic bicycle model as

$$\dot{x} = [\dot{p}_x, \dot{p}_y, \dot{v}, \dot{\psi}, \dot{\delta}] = \left[v \cos\psi + d_x, v \sin\psi + d_y, a + d_v, \frac{v}{L}\tan\delta + d_\psi, \omega + d_\delta \right], \qquad (14)$$

where (p_x, p_y) is the car's position, v is the forward speed, ψ is the heading, δ is the steering angle, $L = 0.5$ m is the wheelbase, $a \in [-3.5, 3.5]$ m/s^2 is the acceleration control, $\omega \in [-5, 5]$ rad/s is the steering angular velocity control, $d \in \mathcal{D} := \{\tilde{d} \in \mathbb{R}^5 : \|\tilde{d}\|_\infty \leq d_{\max}\}$, and $d_{\max}$ is the disturbance bound. The discrete-time dynamics from (14) are computed by fourth-order Runge-Kutta (RK4) with time step 0.1 s and implemented in JAX (Bradbury et al., 2018). The footprint of the car is represented by a box $B = [0., 0.5] \times [-0.1, 0.1]$ m, rotated by the car's heading angle and translated by its position: $B(x) := R_\psi B \oplus \{(p_x, p_y)\}$. We consider three constraints: heading angle, road boundary, and obstacles. The safety margin function is defined as

$$g(x) = \min\{g_\psi(x), g_{\text{road}}(x), g_{\text{obs}}(x)\}, \qquad g_\psi(x) = \frac{\pi}{2} - |\psi|,$$
$$g_{\text{road}}(x) = \min_{p \in B(x)} 0.6 - |p_y|, \qquad g_{\text{obs}}(x) = \min_{i \in [5]} \min_{p \in B(x)} s_{B^i}(p),$$

where $s_{\mathcal{P}} \colon \mathbb{R}^2 \to \mathbb{R}$ is the signed distance function to a nonempty set $\mathcal{P}$ and $B^i := B \oplus \{p^i\}$ are box obstacles at different locations. A bird's-eye view of the environment can be seen in Figure 2.

ISAACS and Baselines We initialize the control policy of ISAACS and SAC-DR by training a standard SAC in the absence of a disturbance. Then, we initialize ISAACS' disturbance policy by training another SAC to attack the fixed initial control policy. The length of the rollout episode is 200 steps (20 seconds) for all RL algorithms. All policy networks have three fully-connected (FC) layers with 256 neurons, and the critic networks have three FC layers with 128 neurons. This amounts to 69,634 parameters or 284 KB of storage. In SAC-DR we sample the disturbance uniformly from the set $\mathcal{D}$, while in ISAACS we sample it from the (stochastic) disturbance policy. Finally, we use resolution-complete finite-difference methods (Bui et al., 2022) to solve the HJI equation (4) numerically on a multi-dimensional grid, which we refer to as the "oracle" in this section. The value function is represented as a grid with about 91 million scalar values, taking 0.7 GB.
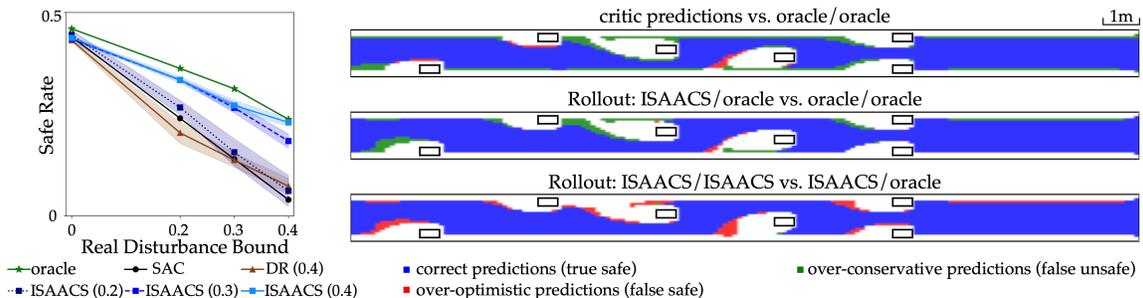
Figure 2: *Left:* Comparison of safety controllers' robustness to disturbances. As the disturbance bound increases, controllers trained without disturbance or with DR rapidly degrade. The ISAACS controller trained against the largest adversarial disturbance suffers the least safety degradation, nearing the optimal (oracle) policy. *Right:* "Confusion plots" of values and rollout outcomes for 2-D slices of the state space, with $v = 1, \psi = 0, \delta = 0.03$. *Top:* learned safety critic can wrongly predict some rollout outcomes, leading to inaccuracies in the estimated safe set boundary. *Middle:* learned ISAACS safety policy achieves near-optimal success but is occasionally suboptimal near the safe set boundary. *Bottom:* direct policy rollout using the learned disturbance can lead to over-optimistic predictions.

Safety Filters We implement our robust rollout safety filter with a modified zonotope-based FRS scheme (Bak, 2020)[5]. We also implement a direct rollout safety filter that checks gameplay trajectories: $\Delta_{\mathrm{ro}}(x_t, \pi^{\mathrm{task}}, \pi^d, H) := \mathbb{1}\{x_{\tau|t} \notin \mathcal{F}, \forall \tau \in \{1, \cdots, H + 1\}\}$, where the zero disturbance in (9) is replaced by disturbances from ISAACS $\pi^d = \pi^d_\phi$ or the oracle $\pi^d = \pi^{d*}$. The safety filter is constructed by replacing $V(x) \geq \epsilon$ with $\Delta_{\mathrm{ro}}(x, \pi^{\mathrm{task}}, \pi^d, H) = 1$ and $\pi^{u*}(x)$ with $\pi^u_\theta(x)$ in (5).

Evaluation We evaluate 400 rollouts with initial states sampled uniformly from the state space $\mathcal{X} = \{x \in \mathbb{R}^5 \mid p_x \in [0, 20], \ p_y \in [-0.6, 0.6], \ v \in [0.4, 2.0], \ \psi \in [-0.5\pi, 0.5\pi], \ \delta \in [-0.35, 0.35]\}$. We compute the *safe rate* as the fraction of trajectories that avoid the failure set for the full horizon. To more closely benchmark the predictions and performance of ISAACS against the oracle solution, we evaluate the critic and roll out the policies from each of the 91 million cells in the oracle's grid. To evaluate different safety filters, we use iterative LQR (Li and Todorov, 2004) as the task policy, with a barrier penalty to discourage violations, and use the oracle disturbance policy with bound $d_{\max} = 0.1$ to attack the controller. We select 395 initial states from which the trained ISAACS safety policy can maintain safety against oracle disturbance, but the task policy results in constraint violations. In addition to the safe rate, we also compute the *filter frequency*, which is the fraction of time steps at which the safety filter was triggered, averaged across all trajectories.

4.2. Results

Offline Adversarial RL We first evaluate the ISAACS controller's robustness. Figure 2 (left) shows the average safe rate of ISAACS and other safety policies under different disturbance bounds over three random seeds with the shaded region for one standard deviation. ISAACS' robustness improves as the bound used in the training increases, indicating that the learned disturbance policy approximates the worst-case uncertainty realization. Further, ISAACS outperforms single-agent RL

---

5. We use zonotopes since the computation of their Minkowski sum is light. We refer readers to (Althoff et al., 2021) for other representations for FRSs.

(even with DR) by a large margin. DR optimizes against the average among all possible disturbances and is less robust to worst-case realizations. Finally, when trained with the highest disturbance bound, ISAACS presents comparable robustness to the oracle safety policy.

Figure 2 (right) shows confusion matrix color plots of value and rollout predictions across 2-D slices of the state space when $v = 1, \psi = 0, \delta = 0.03$. The ISAACS critic (top plot) achieves 1.4% false-safe rate (red region), which is remarkable given that it uses over $1,000\times$ fewer parameters than the numerical HJI oracle. This contrasts with a more conservative 20.4% false-unsafe rate (green region). When pitted against the oracle worst-case disturbance, the learned ISAACS controller (middle plot) loses safety from 10.4% of true safe states, where the oracle controller succeeds. If we replace the oracle disturbance with the learned one (bottom plot), 12% of states where the ISAACS controller fails in the true worst case are mispredicted as safe by the ISAACS gameplay rollout. This fallibility motivates the use of a *robust* rollout of the learned ISAACS controller under all $d \in \mathcal{D}$ rather than relying on a direct rollout of the ISAACS controller and disturbance.

Online Robust Rollout Safety Filter
To account for linearization error, we expand the disturbance bound in each dimension by 5% for the purposes of computing the zonotopic forward-reachable set. Figure 3 shows the average safe rate and filter frequency over three random seeds with shaded regions for one standard deviation. Our proposed robust rollout-based safety filter achieves a perfect 100% safety rate for a long enough looka-
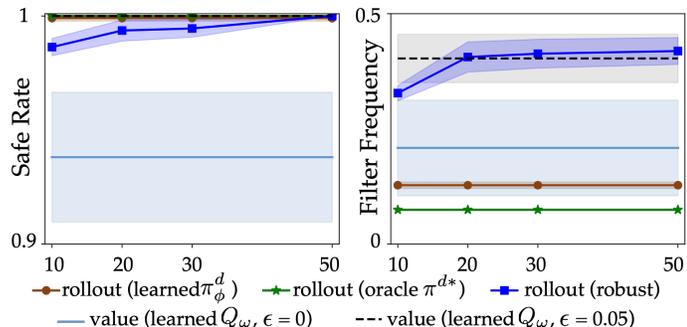

Figure 3: Safe rate and conservativeness of different safety filters. A robust rollout-based safety filter with horizon $H = 50$ achieves zero-violation safety.

head horizon (50 time steps), in contrast with the 99% safe rate obtained through a direct gameplay rollout safety filter and the 94% obtained by naively filtering with the sign of the learned safety critic. The value-based safety filter can be improved by introducing a small threshold $\epsilon = 0.05$ to mitigate approximation errors, even achieving perfect (empirical) safety. Yet, the value-based filter lacks theoretical guarantees, and its threshold needs manual tuning, difficult before real deployment. Despite its lack of guarantees, the direct gameplay rollout filter performs remarkably well, approaching the oracle in both safe rate and filter frequency. The robust rollout-based safety filter has a more conservative activation frequency, similar to the one achieved by the manually thresholded value-based safety filter. Importantly, the robust rollout filter provides strong, clear-cut guarantees, which are of practical importance for the safe deployment of autonomous systems.

5. Conclusion

We propose ISAACS, an adversarial reinforcement learning method for offline safety policy synthesis, whose learned policies can be used to build robust safety-certified control strategies at runtime. We prove safety guarantees for an ISAACS-based safety filter using robust policy rollouts under bounded uncertainty. We demonstrate experimentally that our proposed offline training has comparable performance to numerical methods in a 5-D race car simulator, and the runtime safety filter achieves a perfect zero-violation safe rate with moderate added conservativeness. These results open a promising research avenue for scalable synthesis of robust safety strategies using neural networks.

Acknowledgments

References

Bowen Alpern and Fred B. Schneider. Defining liveness. 21(4):181–185, 1985. ISSN 00200190. doi: 10.1016/0020-0190(85)90056-0. URL https://linkinghub.elsevier.com/retrieve/pii/0020019085900560.

Matthias Althoff, Goran Frehse, and Antoine Girard. Set propagation techniques for reachability analysis. *Annual Review of Control, Robotics, and Autonomous Systems*, 4(1):369–395, 2021. doi: 10.1146/annurev-control-071420-081941.

Aaron D Ames, Jessy W Grizzle, and Paulo Tabuada. Control Barrier Function Based Quadratic Programs with Application to Adaptive Cruise Control. In *53rd IEEE Conference on Decision and Control*, pages 6271–6278. IEEE, 2014.

Aaron D. Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *Proceedings of the 18th European Control Conference (ECC)*, pages 3420–3431, 2019. doi: 10.23919/ECC.2019.8796030.

Stanley Bak. Quick zono reach, 2020. URL https://github.com/stanleybak/quickzonoreach.

Somil Bansal and Claire J. Tomlin. Deepreach: A deep learning approach to high-dimensional reachability. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1817–1824, 2021. doi: 10.1109/ICRA48506.2021.9561949.

Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J. Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. In *Proceedings of the IEEE Annual Conference on Decision and Control (CDC)*, pages 2242–2253, 2017. doi: 10.1109/CDC.2017.8263977.

Osbert Bastani and Shuo Li. Safe reinforcement learning via statistical model predictive shielding. In *Proceedings of Robotics: Science and Systems*, Virtual, 7 2021. doi: 10.15607/RSS.2021.XVII.026.

Homanga Bharadhwaj, Aviral Kumar, Nicholas Rhinehart, Sergey Levine, Florian Shkurti, and Animesh Garg. Conservative safety critics for exploration. In *Proceedings of the 9th International Conference on Learning Representations, ICLR, Virtual Event, Austria, May 3-7, 2021*, 2021. URL https://openreview.net/forum?id=iaO86DUuKi.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

Minh Bui, George Giovanis, Mo Chen, and Arrvindh Shriraman. OptimizedDP: An efficient, user-friendly library for optimal control and dynamic programming, 2022. URL https://arxiv.org/abs/2204.05520.

Jason J. Choi, Donggun Lee, Koushil Sreenath, Claire J. Tomlin, and Sylvia L. Herbert. Robust control barrier-value functions for safety-critical control. In *Proceedings of the 60th IEEE Conference on Decision and Control (CDC)*, pages 6814–6821, 2021. doi: 10.1109/CDC45484.2021. 9683085.

Jaime F. Fisac, Anayo K. Akametalu, Melanie N. Zeilinger, Shahab Kaynama, Jeremy Gillula, and Claire J. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019. doi: 10.1109/TAC. 2018.2876389.

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

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870. PMLR, 7 2018. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

Kai-Chieh Hsu, Vicenç Rubies-Royo, Claire J. Tomlin, and Jaime F. Fisac. Safety and liveness guarantees through reach-avoid reinforcement learning. In *Proceedings of Robotics: Science and Systems*, Virtual, 7 2021. doi: 10.15607/RSS.2021.XVII.077.

Kai-Chieh Hsu, Allen Z. Ren, Duy P. Nguyen, Anirudha Majumdar, and Jaime F. Fisac. Sim-to-lab-to-real: Safe reinforcement learning with shielding and generalization guarantees. *Artificial Intelligence*, 314:103811, 2023. ISSN 0004-3702. doi: https://doi.org/10.1016/j.artint. 2022.103811. URL https://www.sciencedirect.com/science/article/pii/S0004370222001515.

Rufus Isaacs. Differential Games I: Introduction. 1954. URL https://www.rand.org/pubs/research_memoranda/RM1391.html.

W. Langson, I. Chryssochoos, S. V. Raković, and D. Q. Mayne. Robust model predictive control using tubes. *Automatica*, 40(1):125–133, 2004. ISSN 0005-1098. doi: 10.1016/j.automatica. 2003.08.009. URL http://www.sciencedirect.com/science/article/pii/S0005109803002838.

Weiwei Li and Emanuel Todorov. Iterative linear quadratic regulator design for nonlinear biological movement systems. In *ICINCO (1)*, pages 222–229, 2004.

Robert Mattila, Yilin Mo, and Richard M. Murray. An iterative abstraction algorithm for reactive correct-by-construction controller synthesis. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 6147–6152, 2015. doi: 10.1109/CDC.2015.7403186.

Bhairav Mehta, Manfred Diaz, Florian Golemo, Christopher J. Pal, and Liam Paull. Active domain randomization. In *Proceedings of the Conference on Robot Learning*, volume 100, pages 1162–1176, 30 Oct–01 Nov 2020. URL https://proceedings.mlr.press/v100/mehta20a.html.

Ian M. Mitchell, Alexandre M. Bayen, and Claire J. Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 50(7):947–957, 2005. ISSN 1558-2523. doi: 10.1109/TAC.2005.851439.

Quan Nguyen and Koushil Sreenath. Optimal robust time-varying safety-critical control with application to dynamic walking on moving stepping stones. In *Proceedings of the ASME 2016 Dynamic Systems and Control Conference.*, page V002T28A005. American Society of Mechanical Engineers, 2016. ISBN 978-0-7918-5070-1. doi: 10.1115/DSCC2016-9910. URL https://asmedigitalcollection.asme.org/DSCC/proceedings/DSCC2016/50701/Minneapolis,%20Minnesota,%20USA/231011.

Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, pages 2817–2826, 2017.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm, 2017.

Eric Squires, Pietro Pierpaoli, and Magnus Egerstedt. Constructive barrier certificates with applications to fixed-wing aircraft collision avoidance. In *2018 IEEE Conference on Control Technology and Applications (CCTA)*, pages 1656–1661. IEEE, 2018. ISBN 978-1-5386-7698-1. doi: 10.1109/CCTA.2018.8511342. URL https://ieeexplore.ieee.org/document/8511342/.

Brijen Thananjeyan, Ashwin Balakrishna, Suraj Nair, Michael Luo, Krishnan Srinivasan, Minho Hwang, Joseph E. Gonzalez, Julian Ibarz, Chelsea Finn, and Ken Goldberg. Recovery rl: Safe reinforcement learning with learned recovery zones. *IEEE Robotics and Automation Letters*, 6(3):4915–4922, 2021. doi: 10.1109/LRA.2021.3070252.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017. doi: 10.1109/IROS.2017.8202133.

Eugene Vinitsky, Yuqing Du, Kanaad Parvate, Kathy Jang, Pieter Abbeel, and Alexandre Bayen. Robust reinforcement learning using adversarial populations, 2020.

Kim P. Wabersich and Melanie N. Zeilinger. Linear model predictive safety certification for learning-based control. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 7130–7135. IEEE, 2018.

Xiangru Xu, Paulo Tabuada, Jessy W. Grizzle, and Aaron D. Ames. Robustness of control barrier functions for safety critical control. 48(27):54–61, 2015. ISSN 24058963. doi: 10.1016/j.ifacol.2015.11.152. URL https://linkinghub.elsevier.com/retrieve/pii/S2405896315024106.

Jun Zeng, Bike Zhang, Zhongyu Li, and Koushil Sreenath. Safety-critical control using optimal-decay control barrier function with guaranteed point-wise feasibility. In *Proceedings of the American Control Conference (ACC)*, pages 3856–3863, 2021. doi: 10.23919/ACC50511.2021.9482626.

Tijana Zrnic, Eric Mazumdar, Shankar Sastry, and Michael Jordan. Who leads and who follows in strategic classification? In *Advances in Neural Information Processing Systems*, volume 34, pages 15257–15269, 2021. URL https://proceedings.neurips.cc/paper/2021/file/812214fb8e7066bfa6e32c626c2c688b-Paper.pdf.