Bridging Hamilton-Jacobi Safety Analysis and Reinforcement Learning

Jaime F. Fisac* Neil F. Lugovoy* Vicenç Rubies-Royo Shromona Ghosh Claire J. Tomlin

***Abstract*— Safety analysis is a necessary component in the design and deployment of autonomous systems. Techniques from robust optimal control theory, such as Hamilton-Jacobi reachability analysis, allow a rigorous formalization of safety as guaranteed constraint satisfaction. Unfortunately, the computational complexity of these tools for general dynamical systems scales poorly with state dimension, making existing tools impractical beyond small problems. Modern reinforcement learning methods have shown promising ability to find approximate yet proficient solutions to optimal control problems in complex and high-dimensional systems, however their formulation is restricted to problems with an additive payoff (reward) over time, unsuitable for reasoning about safety. In recent work, we proved that the problem of maximizing the *minimum* payoff over time, central to safety analysis, can be time-discounted to induce a contraction mapping. Here, we introduce a novel, time-discounted *Safety Bellman Equation* that renders reinforcement learning techniques amenable to quantitative safety analysis, enabling them to approximate the safe set and optimal safety policy. This opens a new avenue of research connecting control-theoretic safety analysis and the reinforcement learning domain. We demonstrate our formulation on a variety of simulated robotics tasks and reinforcement learning schemes, validating our results against analytic and numerical solutions when these can be obtained, and showing scalability to previously intractable problems of up to 18 state dimensions by exploiting state-of-the-art deep reinforcement learning algorithms.**

I. Introduction

As robotic and automated systems are deployed in the world with an increasing degree of autonomy, safety becomes a central consideration. Safety is fundamentally a constraint satisfaction problem: the state of the system and its environment must never enter failure regions defined by the designer (e.g. collisions, traffic rule violations, power blackouts, etc.).

Unfortunately, providing such constraint satisfaction guarantees is computationally demanding for complex dynamical systems. In the general case, it requires solving a nonlinear optimal control problem in which the objective is not an average or cumulative performance over time but rather the worst case (expressed as a minimum) through time. Work in Hamilton-Jacobi reachability analysis [1–4] has developed rigorous theoretical formulations for which accurate numerical solutions are possible [5]; however, the computational complexity falls prey to the "curse of dimensionality". While analytic solutions exist in rare instances [6, 7], and efficient decompositions are occasionally possible [8], computing safety-ensuring controllers is intractable for many systems of interest.

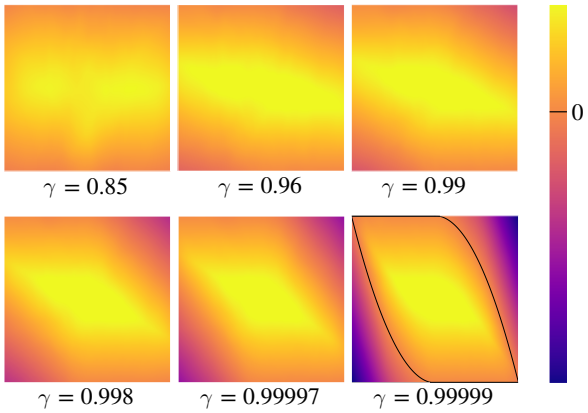

Fig. 1: Multiple snapshots of the neural network output of our Safety Q-learning algorithm for a double-integrator system. As we anneal the discount factor $\gamma \to 1$ during Q-learning, our learned discounted safety value function asymptotically approaches the undiscounted value, allowing us to recover the safe set and optimal safety policy with very high accuracy.

In recent years, reinforcement learning techniques [9] have proven their usefulness in computing data-driven approximate solutions to optimal control problems seeking the maximization of a discounted additive payoff in complex and high-dimensional systems [10–13]. Unfortunately, functions representing a sum of rewards over time are not well suited to capture the safety objective, since safety is not determined by how much a system fails on average, but by whether it fails *at all*. Partly for this reason, reinforcement learning techniques have not seen widespread use for safety analysis.

Another consequence of this disconnect between formulations is that controllers computed through reinforcement learning are typically not inherently safety-preserving, a limitation that has hindered their applicability to physical autonomous systems. In recent years, there has been a growing interest around "safe learning" schemes. Some approaches have proposed formalizing safety as stability [14] or near-constraint satisfaction [15, 16]. Others have built on the Hamilton-Jacobi reachability literature to provide constraint satisfaction guarantees by computing a safety-preserving control policy and overriding the learning controller when it attempts to violate computed safety constraints [17–19]. Unfortunately, this family of methods inherits the difficulty in scaling up computations beyond low-dimensional systems.

The present work seeks to unlock a new family of tools

The authors are with the Department of Electrical Engineering and Computer Sciences. University of California, Berkeley.
Email: {jfisac, nflugovoy, vrubies, shromona.ghosh, tomlin}@berkeley.edu

This research is supported by an NSF CAREER award, the Air Force Office of Scientific Research (AFOSR), NSF's CPS FORCES and VeHICal projects, the UC-Philippine-California Advanced Research Institute, the ONR MURI Embedded Humans, the DARPA Assured Autonomy project, and the SRC CONIX Center.

The authors thank Kene Akametalu and Josh Achiam for helpful insights.

*The first two authors contributed equally to this work.

for safety analysis by rendering a wide range of state-of-the-art methods in the reinforcement learning literature readily usable for safety analysis in high-dimensional systems. Building on the initial work in [20], which introduced a time discount into the minimum-payoff optimal control problem, we propose a similar, tighter discounted formulation of Hamilton-Jacobi safety analysis, obtaining a contraction Bellman operator that lends itself to the use of temporal difference learning techniques. We prove the key properties of this new discounted *Safety Bellman Equation* and show that our resulting *Safety Q-learning* algorithm converges to the safety state-action value function in finite Markov decision processes.

The Safety Q-learning scheme allows us to recover the globally optimal solution to the corresponding Hamilton-Jacobi analysis (to resolution completeness [21, 22]) in low-dimensional problems where dense computation is viable: we validate our results using tabular Q-learning against a double integrator system, achieving high accuracy relative to the analytic solution. We further observe comparable performance when replacing the state-space grid with a neural network function approximator. Crucially, annealing the discount factor during learning allows asymptotic recovery of the solution to the *undiscounted* safety problem (Fig. 1).

We evaluate deep Safety Q-learning on a variety of simulated robotics tasks, and observe consistently accurate results against numerical dynamic programming solutions. In high-dimensional systems beyond the reach of traditional numerical methods, predicted safety accurately matches the empirical performance of the learned safety controller.

We finally implement policy optimization through an adaptation of the basic REINFORCE algorithm [23] to our discounted safety formulation, and explore the potential of using sate-of-the-art methods by similarly adapting the soft actor-critic (SAC) scheme [24]. The promising results on an 18-dimensional problem suggest the usability of this family of reinforcement learning methods for learning policies with the ability to preserve safety in high-dimensional systems.

It is important to clarify that our formulation yields a promising new tool for safety *analysis*: it is not in itself a safe learning framework, since it requires experiencing failure states in order to learn about safety. Our approach is primarily meant to be used as a computational tool in conjunction with a model (simulation) of the system dynamics; its *truly* model-free application, learning constraint satisfaction online directly on the real system, should be limited to training conditions that are not safety-critical (for example, a vehicle test track with only virtual obstacles). Once the safety analysis has been computed (learned), the resulting control policy can be applied to the physical system in similar conditions to other safety controllers, including safe learning of performance objectives [19]. While our analysis here is presented for deterministic dynamics, robust and stochastic extensions are possible (and have been explored to some extent in [20]). We expect that such extensions will be important for implementation of our formulation on physical systems, which is of course its ultimate intended application.

II. Background

A. System dynamics

We consider a dynamical system with state $x \in \mathcal{X} \subset \mathbb{R}^n$, control input $u \in \mathcal{U} \subset \mathbb{R}^m$, and continuous-time dynamics

$$\dot{x} = f(x, u) \ . \tag{1}$$

The set $\mathcal{U}$ is assumed compact and the dynamics f are assumed bounded and Lipschitz continuous; under these conditions, system trajectories $\xi : \mathbb{R}_+ \to \mathcal{X}$ are well defined for all measurable control inputs [25]. We use the notation $\xi_x^{\mathbf{u}}(\cdot)$ for a time trajectory starting at state x under control signal $\mathbf{u}(\cdot)$. For discrete-time approximations, we denote the time step $\Delta t > 0$.

B. Hamilton-Jacobi Safety Analysis

The safety problem can be defined as that of ensuring that undesirable outcomes will not take place during the operation of a system. We introduce some important control theory results on the problem of state constraint satisfaction.

To formalize the safety problem we begin by specifying a set of *failure states* that the system should avoid entering; these failure states can generally encode physical collisions, violations of user specifications or other undesirable outcomes. The complement of the set of failure states is referred to as the *constraint set* $\mathcal{K} \subset \mathcal{X}$. The safety problem is then to determine under what conditions and control inputs the trajectories followed by the system will remain in $\mathcal{K}$ for all time. A system trajectory $\xi_x^{\mathbf{u}}(\cdot)$ is said to be *safe* if for all future times $t \geq 0$, $\xi_x^{\mathbf{u}}(t) \in \mathcal{K}$.

Letting $\mathcal{K}$ be a closed set in $\mathcal{X}$, we can always define a function $l : \mathcal{X} \to \mathbb{R}$ that is satisfies $l(x) \geq 0 \iff x \in \mathcal{K}$ (for example, the signed distance function to $\mathcal{K}$ under any metric on $\mathcal{X}$). We can hence define the optimal control problem, and its associated value function, as:

$$V(x) := \sup_{\mathbf{u}(\cdot)} \inf_{t \geq 0} l\big(\xi_x^{\mathbf{u}}\big) \ . \tag{2}$$

The value function $V : \mathcal{X} \to \mathbb{R}$ captures the minimum payoff l achieved over time by a trajectory starting at each state $x \in \mathcal{X}$ if the best possible control input is applied at every instant. This can intuitively be thought of as the *closest* the system will get to violating the constraints, as measured by the "signed distance" l. This minimum-payoff optimal control problem can be approached via dynamic programming. Considering a finite time horizon $t \in [0, T]$, it is possible to compute the optimal safety value function as the solution to a time-dependent terminal-value Hamilton-Jacobi-Bellman variational inequality of the form [4]:

$$\begin{aligned} &0 = \min\left\{ l(x) - V(x,t), \frac{\partial V}{\partial t} + \max_{u \in \mathcal{U}} \nabla_x V^\top f(x,u) \right\} \ , \\ &V(x,T) = l(x) \ , \quad \forall x \in \mathcal{X} \ . \end{aligned} \tag{3}$$

The discrete-time counterpart, which we will use as a starting point for our discounted formulation, is as follows.

$$V(x,t) = \min\left\{ l(x), \max_{u \in \mathcal{U}} V\big(x + f(x,u)\Delta t, t + \Delta t\big) \right\} \ . \tag{4}$$

In the infinite-horizon case, the value function no longer changes in finite time, and so $V(x)$ must satisfy the fixed-point Bellman equation:

$$V(x) = \min\left\{l(x), \max_{u\in\mathcal{U}} V\big(x + f(x,u)\Delta t\big)\right\} \ . \quad (5)$$

An important observation about (5) is that, unlike the common Bellman equation used in reinforcement learning, it does *not* induce a contraction mapping on V and therefore it is not generally possible to converge to the fixed point by application of value iteration or temporal difference learning.

C. Reinforcement Learning

The field of reinforcement learning comprises a wide variety of data-driven methods by which a system can compute approximations to the optimal value function and/or optimal policy to an optimal control problem. Reinforcement learning is usually formulated in the discrete-time Markov decision process framework, considering the problem of maximizing the cumulative sum of rewards of a trajectory, exponentially discounted over time.

Considering deterministic dynamics (1) and maintaining notation consistent with other sections, the dynamic programming principle associated with this control problem takes the form of the discrete-time Bellman equation [26]:

$$V(x) = \max_{u\in\mathcal{U}} r(x,u) + \gamma V\big(x + f(x,u)\Delta t\big) \ , \quad (6)$$

Crucially, this Bellman update induces a contraction mapping in the space of value functions (under the supremum norm), which implies that its successive application to any initial V will ultimately converge to the unique solution of (6). This enables key convergence results in reinforcement learning schemes, most notably temporal-difference learning methods such as Q-learning [27].

In the next section we introduce a modification of (5) that yields a contraction mapping for our problem of interest, and extend the convergence results of temporal difference learning to safety control problems.

III. The Discounted Safety Bellman Equation

Our central contribution is a modified form of the dynamic programming safety backup (5) which induces a contraction mapping in the space of value functions and is therefore amenable to reinforcement learning methods based on temporal difference learning [10, 27, 28].

Our key observation stems from an intuitive interpretation of time-discounting in the problem of cumulative rewards: at every instant, there is a small probability $1-\gamma$ of transitioning to an absorbing state from which no more rewards will be accrued. Thus in (6) the discount factor $\gamma \in [0,1)$ can be seen as the probability of the episode continuing, with $1-\gamma$ conversely representing the probability of transitioning to a terminal state.

An analogous interpretation in the problem of minimum payoff over time can be achieved by modifying (5) to account for such a transition. Here if, with probability $1-\gamma$, an episode were to end after the current time step, the minimum future $l(\cdot)$ would be equal to the current $l(x)$. This induces the discrete-time discounted dynamic programming equation

$$V(x) = (1-\gamma)l(x) + \gamma \min\left\{l(x), \max_{u\in\mathcal{U}} V\big(x + f(x,u)\Delta t\big)\right\}. \quad (7)$$

This equation yields a strictly tighter contraction mapping than the recent analysis in [20]. By discounting locally towards the current $l(x)$, rather than towards a global upper bound L on l, we significantly reduce the amount of information loss due to discounting. This is shown in the Appendix.

Letting l_i be the value of l achieved by a discrete-time state trajectory $\xi_x^{\mathbf{u}}$ at the i-th time step, the explicit form of the objective maximized in (7) is a "time-discounted" minimum:

$$J(\xi_x^{\mathbf{u}}) = (1-\gamma)l_0 + \gamma\Big[\min\big\{l_0, (1-\gamma)l_1 + \gamma(\min\{l_1, (1-\gamma)l_2 + \gamma\ldots)\}\big\}\Big] \ . \quad (8)$$

We prove two key properties of our proposed equation.

Theorem 1. *(Contraction mapping) The discounted Safety Bellman Equation* (7) *induces a contraction mapping under the supremum norm. That is, let $V, \tilde{V} : \mathcal{X} \to \mathbb{R}$, then there exists a constant $\kappa \in [0,1)$ such that $\|B[V] - B[\tilde{V}]\|_\infty \leq \kappa\|V - \tilde{V}\|_\infty$.*

Proof: It will suffice to show that for all *states $x \in \mathcal{X}$, $|B[V](x) - B[\tilde{V}](x)| < \kappa\|V - \tilde{V}\|_\infty$. We have:*

$$\begin{aligned}
&|B[V](x) - B[\tilde{V}](x)| \\
&\quad = \gamma|\min\{l(x), \max_{u\in\mathcal{U}} V\big(x + f(x,u)\Delta t\big)\} \\
&\qquad - \min\{l(x), \max_{\tilde{u}\in\mathcal{U}} \tilde{V}\big(x + f(x,\tilde{u})\Delta t\big)\}| \\
&\quad \leq \gamma|\max_{u\in\mathcal{U}} V\big(x + f(x,u)\Delta t\big) - \max_{\tilde{u}\in\mathcal{U}} \tilde{V}\big(x + f(x,\tilde{u})\Delta t\big)| \ .
\end{aligned}$$

Now, without loss of generality suppose the first maximum is the larger one, and let $u^* \in \mathcal{U}$ achieve it. We continue:

$$\begin{aligned}
&|B[V](x) - B[\tilde{V}](x)| \\
&\quad \leq \gamma|V\big(x + f(x,u^*)\Delta t\big) - \tilde{V}\big(x + f(x,u^*)\Delta t\big)| \\
&\quad \leq \gamma \max_{u\in\mathcal{U}} |V\big(x + f(x,u)\Delta t\big) - \tilde{V}\big(x + f(x,u)\Delta t\big)| \\
&\quad \leq \gamma \sup_{\tilde{x}} |V(\tilde{x}) - \tilde{V}(\tilde{x})| = \gamma\|V - \tilde{V}\|_\infty \ .
\end{aligned}$$

Thus the sought contraction constant is in fact $\gamma \in [0,1)$. □

Proposition 1. *(Value approximation) In the limit of no discounting, the fixed-point solution to the Safety Bellman Equation* (7) *converges to the undiscounted safety value function.*
Proof: Taking the limit of the optimization of (8) *as γ goes to* 1 *we recover:*

$$\lim_{\gamma\to 1} V(x) = \max_{\mathbf{u}^{0:T}} \min\big\{l_0, l_1, l_2, \ldots\big\} \ ,$$

which solves (5) *and is the discrete-time approximation to* (2). □

The above two theoretical results enable the use of reinforcement learning techniques for safety analysis. We end this section with an important consequence of Theorem 1.

Theorem 2. *(Convergence of Safety Q-learning) Let* $\mathbf{X} \subseteq \mathcal{X}$ *and* $\mathbf{U} \subseteq \mathcal{U}$ *be finite discretizations of the state and action spaces, and let* $\mathbf{f} : \mathbf{X} \times \mathbf{U} \to \mathbf{X}$ *be a discrete transition function approximating the system dynamics. The Q-learning scheme applied to the discounted safety problem and executed on the above discretization converges, with probability* 1*, to the optimal state-action safety value function*

$$Q(\mathbf{x},\mathbf{u}) := (1-\gamma)l(\mathbf{x}) + \gamma \min\left\{l(\mathbf{x}), \max_{\mathbf{u}'\in\mathbf{U}} Q\big(\mathbf{f}(\mathbf{x},\mathbf{u}),\mathbf{u}'\big)\right\} ,$$

in the limit of infinite exploration time and given partly-random episode initialization and learning policy with full support over $\mathbf{X}$ *and* $\mathbf{U}$ *respectively. Concretely, learning is carried out by the update rule:*

$$Q_{k+1}(\mathbf{x},\mathbf{u}) \leftarrow Q_k(\mathbf{x},\mathbf{u}) + \alpha_k\Big[(1-\gamma)l(\mathbf{x}) + \gamma \min\left\{l(\mathbf{x}), \max_{\mathbf{u}'\in\mathbf{U}} Q\big(\mathbf{f}(\mathbf{x},\mathbf{u}),\mathbf{u}'\big)\right\} - Q_k(\mathbf{x},\mathbf{u})\Big] ,$$

for learning rate $\alpha_k(\mathbf{x},\mathbf{u})$ *satisfying*

$$\sum_k \alpha_k(\mathbf{x},\mathbf{u}) = \infty \quad \sum_k \alpha_k^2(\mathbf{x},\mathbf{u}) < \infty ,$$

for all $\mathbf{x} \in \mathbf{X}, \mathbf{u} \in \mathbf{U}$.

Proof: Our proof follows from the general proof of Q-learning convergence for finite-state, finite-action Markov decision processes presented in [29]. Our transition dynamics $\mathbf{f}$*, initialization and policy randomization, and learning rate* α_k *satisfy Assumptions 1, 2, and 3 in [29] in the standard way. The only critical difference in the proof is the contraction mapping, which we obtain under the supremum norm by Theorem 1: with this, Assumption 5 in [29] is met, granting convergence of Q-learning by Theorem 3 in [29].* □

We stress that, beyond Q-learning, the contraction-mapping property of our discounted safety backup opens the door to straightforward application of a wide variety of reinforcement learning schemes to safety analysis. We dedicate the following section to a first demonstration in which we explore the application of canonical reinforcement learning algorithms in the two main families: value learning and policy optimization.

IV. Results

We present the results of implementing our proposed discounted Safety Bellman Equation in multiple reinforcement learning schemes: tabular Q-learning [27], deep Q-learning (DQN) [10], REINFORCE [23], and soft actor-critic (SAC) [24], and four different dynamical systems. We first validate the computed safety value function and safe set against analytically and numerically obtained ground-truth references in traditionally tractable systems. We consider two dynamical systems commonly used as benchmarks in control theory, namely a 2-D double-integrator system and a 4-D cart-pole system. We then demonstrate the scalability and usefulness of our formulation in higher-dimensional nonlinear systems, for which exact safety analysis is generally considered intractable. We use simulation environments common in reinforcement learning [30], namely a 6-D lunar lander system and an 18-D "half-cheetah" system.

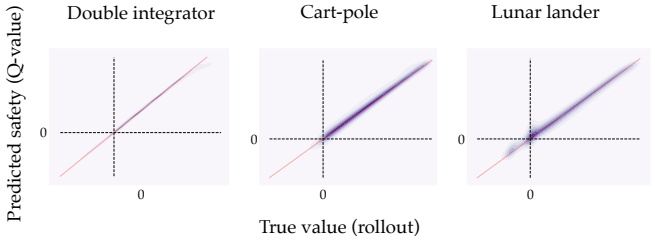

Fig. 2: Predicted vs. achieved minimum signed distance to violations for 10^6 simulated rollouts with 100 trained networks. Red line indicates identity.

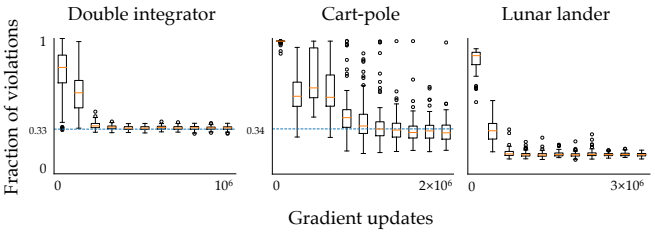

Fig. 3: Fraction of initial conditions resulting in violations as training proceeds. Each data point is a sample average from 1000 episodes; statistics are taken over 100 independent training runs. As learning progresses, the fraction of violations reliably decreases, approaching the ground-truth fraction of unsafe states (from which violation is inevitable) for the double integrator and cart-pole. Lunar lander ground truth is unknown.

A. Validation: comparison to ground truth

1) Analytic validation: double integrator: The double integrator is a classic reachability example where the control policy seeks to keep the system in the set $\{[x,v] \in \mathbb{R}^2 : x \in [x_{\min}, x_{\max}]\}$ with the dynamics characterized by:

$$\dot{x} = v \ , \quad \dot{v} = u \ , \tag{9}$$

with $|u| \leq u_{\max}$, where x can be seen as position, v as velocity, and u as an acceleration input. Analytically, the safe set is characterized by the interior of the boundary defined by the parabolic segments

$$\begin{cases} x_{\text{low}} + \frac{v^2}{2u_{\max}} & v \leq 0 \\ x_{\text{high}} - \frac{v^2}{2u_{\max}} & v \geq 0 \end{cases} \tag{10}$$

and the boundaries $x = x_{\text{low}}$, $x = x_{\text{high}}$. Although simple, this example proves a useful context for visualizing the effect of γ, since the entire value function can be represented in two dimensions.

It can be seen in Fig. 1 how as γ is annealed the time horizon of safety is effectively extended: for lower values the value function resembles $l(\cdot)$, and for higher values it approaches the undiscounted value function. Final accuracy and in-training performance are shown in Fig. 2 and Fig. 3.

Using tabular Q-learning with $l(\cdot)$ as the signed Euclidean distance to the boundary of the constraint set and annealing γ to 1 similar to [31], we observe convergence to the safe set up to the resolution of the grid. Independently training 100 deep Q-networks [10] with fully-connected layers using our discounted Safety Bellman Equation we find near-convergence to the safe set with an average 2.26×10^{-5} (minimum 0, maximum 1.27×10^{-4}) fraction of points incorrectly characterized as safe and an average 1.76×10^{-4} (minimum 3.26×10^{-5}, maximum 3.31×10^{-4}) of points falsely characterized as unsafe. Classification is visualized in Fig. 4.

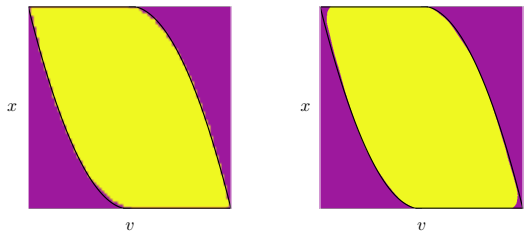

Fig. 4: Safe sets learned by tabular (left) and deep Q-learning (right) with the discounted Safety Bellman Equation compared to the analytic set (black).

2) Numerical validation: cart-pole: The cart-pole system (inverted pendulum) is a classic control problem and one ripe for safety analysis. A cart moving on a one-dimensional track is attached by a pivot to a pole. The control policy seeks to keep the pole from falling and to keep the cart from the edge of the track by applying accelerations to the cart. For this system, the ground truth safe set must be computed numerically on a grid using dynamic programming [1]. Over 100 random seeds we find that an average $5.16 \times 10^{-3}\%$ (minimum $4.90\times10^{-4}\%$, maximum $2.56\times10^{-2}\%$) of points are misclassified as safe and an average $5.80 \times 10^{-2}\%$ (minimum $4.24\times10^{-2}\%$, maximum $8.4\times10^{-2}\%$) of points are misclassified as unsafe, relative to the numerically approximated ground truth. In reality, the precision of the numerical ground truth is limited by the grid resolution; thus, if we consider any points less than one full grid cell away from a safe grid point to be safe, we find that only an average 1.47×10^{-6} (minimum 0, maximum 4.54×10^{-5}) fraction of points are misclassified as safe by our method (Figs. 2 and 3).

B. Scalability: safety for high dimensional systems

The two examples we have shown thus far help us validate our approach against well-established safety analysis tools. However, a motivating factor of this work is to enable safety analysis for systems that are too high-dimensional for traditional approaches. In this section we will explore how our method fares in two high-dimensional systems from the OpenAI Gym environment collection [30].

1) Temporal difference: lunar lander: We first consider a lunar lander system with 6 states $s = [x, y, \theta, \dot{x}, \dot{y}, \dot{\theta}]$ (vehicle pose and velocities). The signed distance safety function is defined as $l(s) = \max\{l_{\text{fly}}(s), l_{\text{land}}(s)\}$, with $l_{\text{fly}}(s) = \min\{x - x^w_{\min}, x^w_{\max} - x, y - y^w_{\min}, y^w_{\max} - y\}$, and $l_{\text{land}}(s) = \min\{x - x^p_{\min}, x^p_{\max} - x, \theta - \theta_{\min}, \theta_{\max} - \theta,\ \dot{y} - \dot{y}_{\min}\}$. Terms marked with superscript w indicate viewing window limits, and terms marked with superscript p indicate landing pad limits. The margin $l(\cdot)$ is thus constructed to allow either flying in free space or landing on the pad; this example illustrates the ability to encode arbitrary state constraints through a signed distance function.

We train 100 Safety DQNs with different random seeds and compare learned values against the observed safety by performing on-policy rollouts in simulation (Fig. 2). Since computing the safety value function through dynamic programming is intractable on 6-dimensional systems, there is no known ground truth to compare against (Fig. 3). While the learned Q-value function may be suboptimal, it does give accurate safety predictions for its induced best-effort policy. We present x-y slices of a sample trained value function in Fig. 5, where the learned safety structure can be seen.

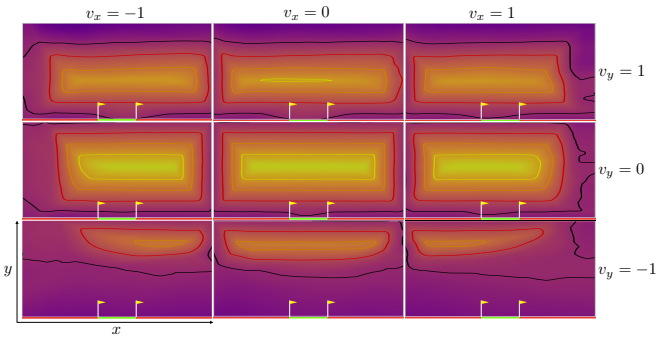

Fig. 5: Slices of the learned lunar lander value function overlaid on the image of the viewing window for $\theta = 0$ and $\dot{\theta} = 0$. Computed safe set boundary in black. At low speeds, the values near the ground are higher close to the landing pad, revealing the effect of l_{land}. For large downward velocities, ground collision is inevitable from the lower half of the screen.

2) Policy optimization: half-cheetah: Many successful modern reinforcement learning methods use neural networks to directly represent control policies and search for efficient strategies. A number of policy gradient algorithms derive their policy update from the REINFORCE rule [23]:

$$\nabla_\theta \mathbb{E}_{\xi\sim\pi_\theta}[J(\xi)] = \mathbb{E}_{\xi\sim\pi_\theta}[\nabla_\theta \log(p_{\pi_\theta}(\xi)) J(\xi)] \ , \qquad (11)$$

with $p_{\pi_\theta}(\cdot)$ denoting the probability of taking a trajectory ξ under the stochastic policy π_θ parametrized by θ, and $J(\cdot)$ denoting the outcome of ξ. Taking $J(\cdot)$ to represent the time-discounted minimum payoff $l(\cdot)$ of the trajectory as in (8), we can directly optimize a policy for discounted safety.

We consider an 18-dimensional half-cheetah system within the MuJoCo physics simulator [32], and define $l(\cdot)$ to be the minimum height of the head and the front leg, so that a failure occurs if either touches the ground (Fig. 6). Note that we must (at least in part) initialize trajectories at configurations from which the system could in principle maintain safety. Running policy gradient using REINFORCE, all policies trained for discounted safety attempt to balance, though not always successfully, and some learn to sit. In contrast, policies trained with the standard reinforcement learning formulation using $l(\cdot)$ as an additive reward tend to raise the front leg and sometimes jump, and invariably fall over. Defining an alternative reward that purely penalizes forbidden contacts similarly failed to yield safe learned behaviors.

Using the more sophisticated soft actor-critic (SAC) algorithm [24] we find that after hyper-parameter optimization, all policies trained across 20 random seeds using a discounted sum of $l(\cdot)$ launch the cheetah into the air and always fall over. Using a discounted sum of contact penalties, 65% of policies do learn to sit; however, the remaining 35% produce unsafe jumping behavior. We speculate that the sparsity of the reward signal makes learning challenging. Across the 20 random seeds, all policies trained with discounted safety visibly attempt to stand: 80% of them succeed in doing so reliably, with an additional 5% reliably sitting if standing fails. The different emergent policies are depicted in Fig. 6.

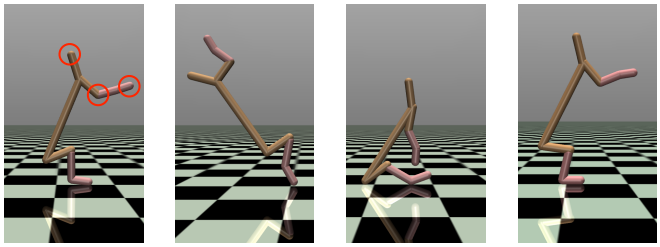

Fig. 6: Learned half-cheetah safety policies aimed to keep the head and front leg off the ground. *Left to right:* typical starting configuration; an unsafe jumping policy learned using a sum of discounted heights; a safe sitting policy learned using discounted safety or (less reliably) discounted sum of contact penalties; a safe standing policy learned using discounted safety.

V. Conclusion

A. Summary

We present a time-discounted *Safety Bellman Equation* whose unique fixed-point solution converges to the undiscounted Hamilton-Jacobi safety value function as the time discounting is asymptotically relaxed. Our new formulation can readily be used with a wide variety of state-of-the-art reinforcement learning algorithms by a simple modification of their Bellman update step. We prove the convergence of the resulting *Safety Q-learning* scheme in finite Markov decision processes, which to our knowledge is the first model-free method for computing the safety value function beyond naive Monte Carlo trajectory shooting. Adopting function approximation techniques from modern reinforcement learning, we demonstrate scalability for higher dimensional systems and find that our DQN-based approach is accurate compared to analytic solutions and existing numerical methods. Finally, by directly optimizing the control policy for discounted safety in a system with 18 continuous state dimensions, leading to *Safety REINFORCE* and *Safety SAC*, we find that it is possible to learn control policies that preserve safety with significantly more success than those obtained from the standard reinforcement learning formulation using a sum of discounted rewards.

B. Implications for Hamilton-Jacobi safety analysis

We see our contribution as unlocking a family of tools for safety analysis that can be successfully applied to systems intractable for traditional techniques. Once computed, the learned safety analysis can have practical applications in safe robot control and safe reinforcement learning, analogous to existing safety analysis tools. To this end, we expect that robust formulations, as well as research in transfer learning will facilitate the use of simulation-based safety analysis in physical systems; we also note that, subject to model fidelity, conservative approximations of the safety value can always be guaranteed through forward simulation of the computed safety policy [33]. While our focus in this work is on model-free reinforcement learning algorithms, model-based methods such as value and policy iteration can also be employed for safety analysis under the discounted safety formulation. We recently explored such methods in [20].

C. Implications for reinforcement learning

By introducing a Safety Bellman Equation that is readily compatible with the reinforcement learning framework, we hope to enable and inspire researchers and practitioners in the field of reinforcement learning to explicitly include safety in their learning algorithms. Ultimately, we hope that, by enabling learning systems to reason about constraint satisfaction, future advances in the field will bring about highly capable intelligent systems that can be deployed safely [34].

D. Limitations and future work

To reach convergence under model-free learning schemes, the system must repeatedly violate the constraints. Thus model-free algorithms using discounted safety must be used in simulation or an environment where leaving the constraint set does not result in catastrophic failure. Additionally, policies trained with our formulation will seek to maintain safety but not accomplish another task while staying safe. Thus combining discounted safety with performance-driven learning is a natural next step, e.g. by using the learned safety policy in a supervisory control framework [17, 19]. Since the Safety Q-learning scheme is off-policy, safety analysis can be continually updated in the background even while a system is controlled by a different policy. Finally, while deep neural networks are expressive function approximators, their training methods do not in general have convergence guarantees. It may prove fruitful to investigate combining our work with recent research in neural network verification [35] to provide formal guarantees about learned value functions and control policies.

Appendix

We show that our proposed discounted Safety Bellman Equation (7) yields a tighter contraction mapping than the alternative recently proposed in [20], which uses a global upper bound L on the function $l(x)$ (i.e. $l(x) \leq L, \forall x \in \mathcal{X}$):

$$V(x) = \min\left\{l(x), \max_{u \in \mathcal{U}} (1-\gamma)L + \gamma V\big(x + f(x,u)\Delta t\big)\right\} \tag{12}$$

It can be seen that (12) incurs a heavier information loss than (7) relative to the undiscounted backup (5). Both backups are exact when $l(x) \leq V\big(x + f(x,u)\Delta t\big)$, since they evaluate to $l(x)$. Wherever $l(x) > V\big(x + f(x,u)\Delta t\big)$, the right-hand side of the exact safety backup (5) is $V\big(x + f(x,u)\Delta t\big)$. For any $\gamma \in (0,1)$ we have that the right-hand side of (5) evaluates to $(1-\gamma)l(x) + \gamma V\big(x + f(x,u)\Delta t\big)$, which is strictly less than $l(x)$, and much closer to $V\big(x + f(x,u)\Delta t\big)$ for γ close to 1. Conversely, in (12), the second term in the minimum is $(1-\gamma)L + \gamma(V\big(x + f(x,u)\Delta t\big)$. By definition of L, this term is larger than the right-hand side of (7) at any states where the bound L on $l(x)$ is strict. Further, at states where $l(x)$ is sufficiently far from L, this term will be greater than $l(x)$ itself, causing the right-hand side of (12) to evaluate to $l(x)$, effectively losing the information about future safety contained in $V\big(x + f(x,u)\Delta t\big)$. This loss may persist even for γ close to 1, making the global discounting in (12) less amenable to reinforcement learning tools.

REFERENCES

[1] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin. "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games". *IEEE Transactions on Automatic Control* 50.7 (July 2005).

[2] J. Lygeros. "On reachability and minimum cost optimal control". *Automatica* 40.6 (June 2004).

[3] A. Abate, M. Prandini, J. Lygeros, and S. Sastry. "Probabilistic Reachability and Safety for Controlled Discrete Time Stochastic Hybrid Systems". *Automatica* March (2008).

[4] J. F. Fisac, M. Chen, C. J. Tomlin, and S. S. Sastry. "Reach-avoid problems with time-varying dynamics, targets and constraints". *18th International Conference on Hybrid Systems Computation and Control (HSCC)*. ACM Press, 2015.

[5] I. Mitchell and J. Templeton. "A Toolbox of Hamilton-Jacobi Solvers for Analysis of Nondeterministic Continuous and Hybrid Systems". *Hybrid Systems: Computation and Control* (2005).

[6] J. Darbon and S. Osher. "Algorithms for overcoming the curse of dimensionality for certain Hamilton-Jacobi equations arising in control theory and elsewhere". *Research in the Mathematical Sciences* 3.1 (2016).

[7] E. Garcia, D. W. Casbeer, and M. Pachter. "Design and Analysis of State-Feedback Optimal Strategies for the Differential Game of Active Defense". *IEEE Transactions on Automatic Control* (2018).

[8] M. Chen, S. Herbert, and C. J. Tomlin. "Fast Reachable Set Approximations via State Decoupling Disturbances". *Conference on Decision and Control (CDC)* (2016).

[9] R. S. Sutton, A. G. Barto, et al. *Reinforcement learning: An introduction*. MIT press, 1998.

[10] V. Mnih, K. Kavukcuoglu, D. Silver, et al. "Human-level control through deep reinforcement learning". *Nature* 518.7540 (2015).

[11] N. Heess, D. TB, S. Sriram, et al. "Emergence of Locomotion Behaviours in Rich Environments". *CoRR* abs/1707.02286 (2017).

[12] J. Schulman, S. Levine, P. Moritz, et al. "Trust Region Policy Optimization". *CoRR* abs/1502.05477 (2015).

[13] S. Levine, C. Finn, T. Darrell, and P. Abbeel. "End-to-End Training of Deep Visuomotor Policies". *CoRR* abs/1504.00702 (2015).

[14] F. Berkenkamp, R. Moriconi, A. P. Schoellig, and A. Krause. "Safe learning of regions of attraction for uncertain, nonlinear systems with Gaussian processes". *Conference on Decision and Control (CDC)* (2016).

[15] T. M. Moldovan and P. Abbeel. "Safe exploration in Markov decision processes". *International Conference on Machine Learning (ICML)* (2012).

[16] J. Achiam, D. Held, A. Tamar, and P. Abbeel. "Constrained Policy Optimization". *International Conference on Machine Learning*. 2017.

[17] J. H. Gillula and C. J. Tomlin. "Guaranteed Safe Online Learning via Reachability: tracking a ground target using a quadrotor". *International Conference on Robotics and Automation (ICRA)* (May 2012).

[18] A. K. Akametalu, J. F. Fisac, J. H. Gillula, et al. "Reachability-based safe learning with Gaussian processes". *Conference on Decision and Control (CDC)* (2014).

[19] J. F. Fisac*, A. K. Akametalu*, M. N. Zeilinger, et al. "A general safety framework for learning-based control in uncertain robotic systems". *IEEE Transactions on Automatic Control (in press)* (2018).

[20] A. K. Akametalu, S. Ghosh, J. F. Fisac, and C. J. Tomlin. "A Minimum Discounted Reward Hamilton-Jacobi Formulation for Computing Reachable Sets". *arXiv preprint* (2018).

[21] J. Barraquand and J. .-.-C. Latombe. "Nonholonomic multibody mobile robots: Controllability and motion planning in the presence of obstacles". *Algorithmica* 10.2 (Oct. 1993).

[22] Peng Cheng and S. LaValle. "Resolution complete rapidly-exploring random trees". 2003.

[23] R. J. Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning". *Machine Learning* 8.3 (May 1992).

[24] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor". *CoRR* abs/1801.01290 (2018).

[25] E. A. Coddington and N. Levinson. *Theory of ordinary differential equations*. Tata McGraw-Hill, 1955.

[26] R. Bellman. *Dynamic Programming*. 1st ed. Princeton, NJ, USA: Princeton University Press, 1957.

[27] C. J. C. H. Watkins and P. Dayan. "Q-learning". *Machine Learning* 8.3-4 (May 1992).

[28] R. S. Sutton. "Learning to Predict by the Method of Temporal Differences". *Machine Learning* 3.1 (1988).

[29] J. N. Tsitsiklis. "Asynchronous Stochastic Approximation and Q-Learning". *Machine Learning* 16.3 (1994).

[30] G. Brockman, V. Cheung, L. Pettersson, et al. "OpenAI Gym". *arXiv preprint* (2016).

[31] OpenAI. *OpenAI Five*. `https://blog.openai.com/openai-five/`. [Online; accessed 18 September 2018]. 2018.

[32] E. Todorov, T. Erez, and Y. Tassa. "Mujoco: A physics engine for model-based control". *International Conference on Intelligent Robots and Systems (IROS)* (2012).

[33] V. Rubies-Royo, D. Fridovich-Keil, S. Herbert, and C. J. Tomlin. "A Classification-based Approach for Approximate Reachability". *arXiv preprint* (2018).

[34] D. Amodei, J. Steinhardt, D. Man, and P. Christiano. "Concrete Problems in AI Safety". *arXiv preprint* (2017).

[35] G. Katz, C. W. Barrett, D. L. Dill, et al. "Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks". *CoRR* abs/1702.01135 (2017).