Safe Reinforcement Learning through Barrier Functions

Sriram Kodey, Department of Mechanical Engineering

Introduction

Reinforcement Learning (RL) is quite successful at learning near optimal policies for continuous control tasks in simulation but faces challenges in real-world applications due to safety concerns. Systems will probably fail before learning an optimal policy.

Traditional model-based approaches like Lyapunov methods and model predictive control guarantee safety but are not exploratory algorithms.

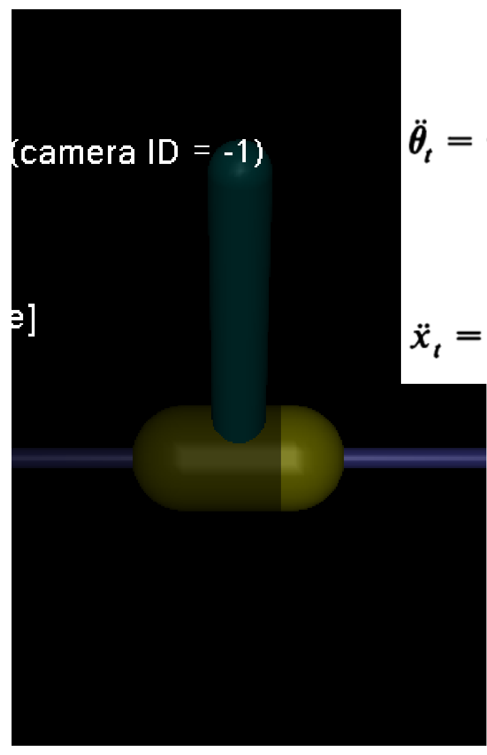
Current safe RL techniques, such as reward-shaping and policy optimization, only approximate safety after initial learning, often violating safety early on.

This project explores using Control Barrier Functions (CBFs) as hard constraints to ensure safety throughout the learning process and investigates guiding policy exploration using CBFs.

Problem Description

RL's model-free nature allows it to generalize across various dynamic systems. This paper presents a framework that integrates any RL architecture with any Lyapunov-type Barrier Function, ensuring broad applicability and enhanced safety.

Inverted Pendulum cart pole system, with nonlinear dynamics and non-minimum phase characteristics.



$$\ddot{\theta}_t = \frac{g\sin\theta_t + \cos\theta_t\left[\frac{-F_t - ml\dot{\theta}_t^2\sin\theta_t + \mu_c\,\mathrm{sgn}(\dot{x}_t)}{m_c+m}\right] - \frac{\mu_p\dot{\theta}_t}{ml}}{l\left[\frac{4}{3} - \frac{m\cos^2\theta_t}{m_c+m}\right]}$$

$$\ddot{x}_t = \frac{F_t + ml\left[\dot{\theta}_t^2\sin\theta_t - \ddot{\theta}_t\cos\theta_t\right] - \mu_c\,\mathrm{sgn}(\dot{x}_t)}{m_c+m}$$

g = −9.8 m/s^2, acceleration due to gravity,
m_c = 1.0 kg, mass of cart
m = 0.1 kg, mass of pole,
l = 0.5 m, half-pole length,
μ_c = 0.0005, coefficient of friction of cart on track,
μ_p = 0.000002, coefficient of friction of pole on cart,
F_t = ±10.0 newtons, force applied to cart's center of mass at time t.

Fig 1: Inverted Pendulum in MuJoCo

The Barrier Function can be tailored to specific applications. In this project, the goal is to constrain the cart's position, ensuring it stays within the rail boundaries while learning a policy to keep the pole upright.

Background

System setup: a Markov Decision Process with control affine dynamics.

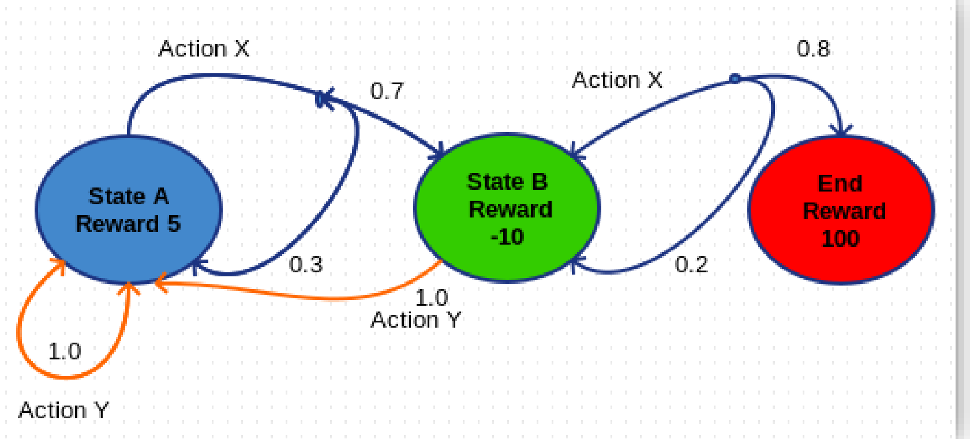


Fig 2: Markov Decision Process

$$s_{t+1} = f(s_t) + g(s_t)a_t + d(s_t)$$

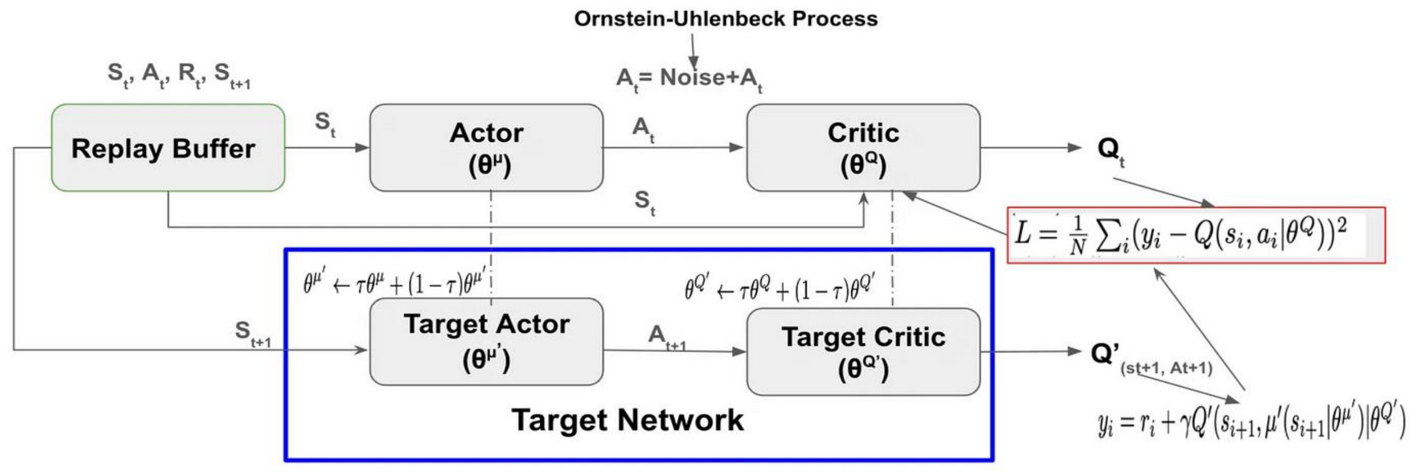
$$J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t)\right]$$

RL algorithm: learns a stochastic control policy $\pi(a|s)$ that maximizes the expected reward $J(\pi)$ of the trajectory $\tau\sim\pi, \tau = \{s_t, a_t, \ldots, s_{t+n}, a_{t+n}\}$.

Control Barrier Functions: Given a system $\dot{x} = f(x,u)$, having a safe set C defined by a function h, if $\exists u \;\; s.t. \;\; \dot{h}(x,u) \geq -\alpha(h(x)) \Leftrightarrow C \; is \; invariant.$

Proposed Approach

DDPG: is an actor-critic algorithm that combines the benefits of DQNs with capability of policy gradient methods.


Ornstein-Uhlenbeck Process
S$_t$, A$_t$, R$_t$, S$_{t+1}$
A$_t$= Noise+A$_t$
Replay Buffer
S$_t$
Actor
(θ^μ)
A$_t$
Critic
(θ^Q)
Q$_t$
S$_t$
$L = \frac{1}{N}\sum_i(y_i - Q(s_i, a_i|\theta^Q))^2$
$\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$
$\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$
S$_{t+1}$
Target Actor
(θ$^{μ'}$)
A$_{t+1}$
Target Critic
(θ$^{Q'}$)
Q'$_{(st+1, At+1)}$
$y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
Target Network

Actor ~ Policy, Critic ~ Q Function

Used to train the Target Actor and Critic. Target networks are deployed.

Barrier Function: $b(z) = r^2 - x^2$

HOCBF: Condition for control invariance

$$\mathcal{L}_f^2 b(z) + \mathcal{L}_g\mathcal{L}_f b(z)u + \mathcal{L}_f(\alpha_1 \circ b)(z) + \alpha_2\big(\mathcal{L}_f b(z) + \alpha_1(b(z))\big) \geq 0$$

DDPG-CBF: The model-free RL controller proposes a control based on the measurement (state estimate), which is fed to the CBF Controller. The CBF controller adds a signal to input that ensure control invariance of the system at all times.

$$u_{cbf} = -\frac{\big(\mathcal{L}_f^2 b(z) + \mathcal{L}_f(\alpha_1 \circ b)(z) + \alpha_2\big(\mathcal{L}_f b(z) + \alpha_1(b(z))\big) + \mathcal{L}_g\mathcal{L}_f b(z)u_\theta\big)}{\mathcal{L}_g\mathcal{L}_f b(z)} + \frac{\eta}{\mathcal{L}_g\mathcal{L}_f b(z)}$$

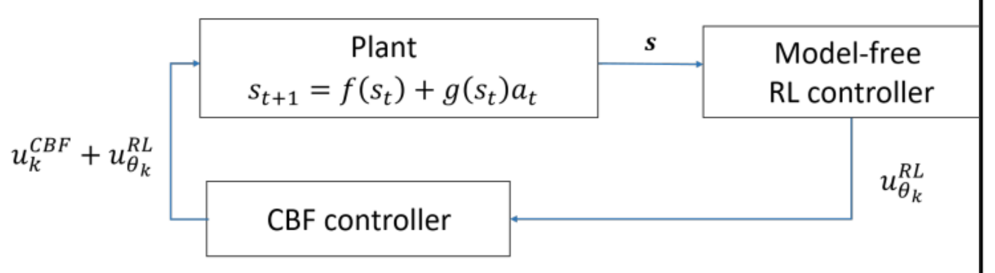


Fig 3: RL-CBF Controller architecture

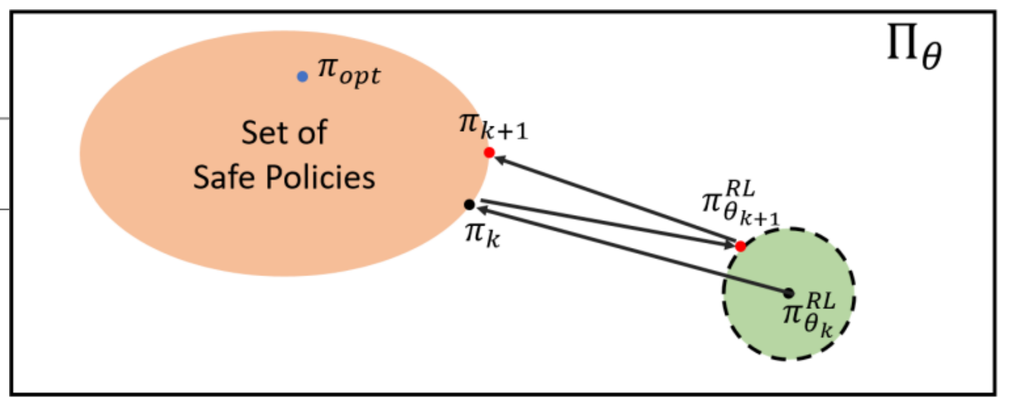


Fig 4: Policy optimization with Barrier compensator

Experimental results

The RL-CBF controller was deployed and evaluated on the Inverted Pendulum environment in gym with the MuJoCo physics simulator.

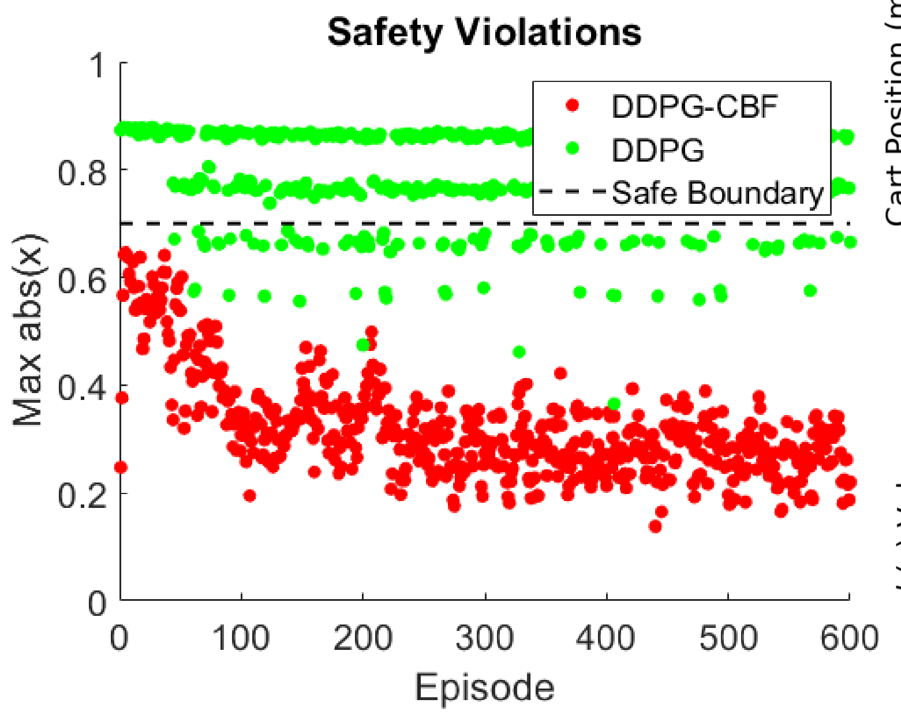


Fig 5a: Safety violations during training.

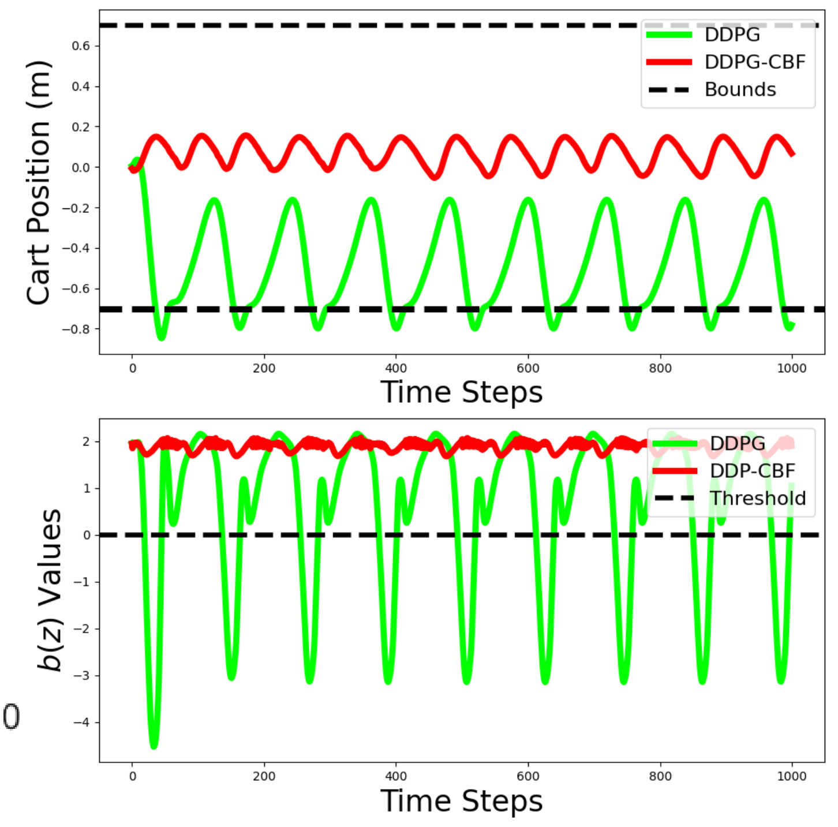


Fig 5b: Performance after training

Discussion:

- The pure DDPG controller has clearly violated the safety boundary throughout the learning process.
- The DDPG-CBF controller establishes hard safety constraints through the use of CBFs, thus no violations are observed during the entire training process.
- DDPG-CBF requires more training time, since it has to check barrier constraint violations and compute the barrier compensator input for each batch during training.
- CBF guided policy exploration can make training faster.
- Barrier function doesn't violate threshold with CBFs.

Conclusions

The results demonstrate the **successful application of CBFs in RL** architecture for safety guarantees during the training process.

Future work includes using CBFs to guide RL exploration for safe and efficient training.

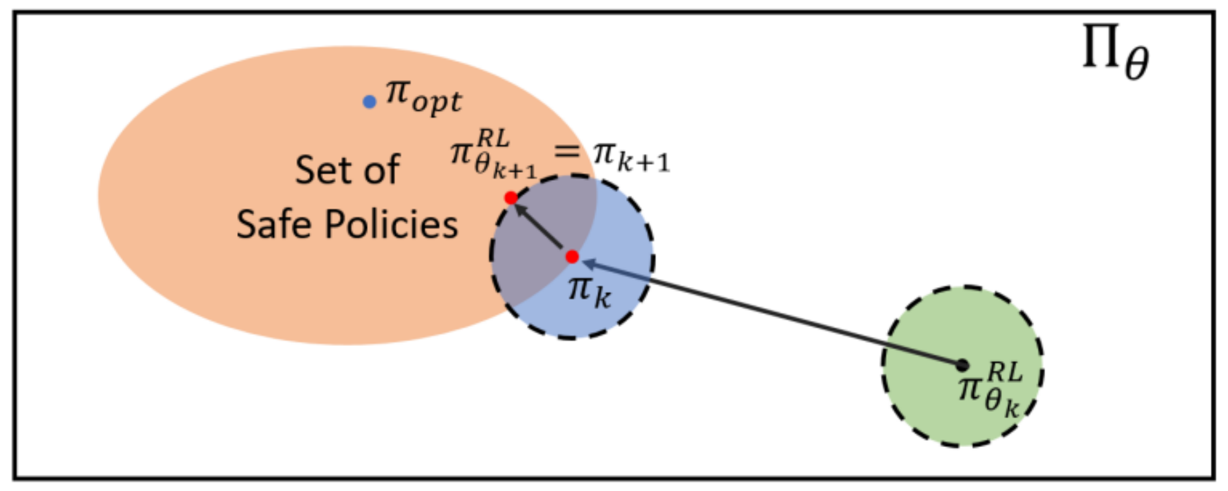


Fig 6: Barrier guided Policy optimization

References:
[1] R. Cheng, G. Orosz, R. Murray, and J. Burdick, "End-to-End Safe Reinforcement Learning through Barrier Functions for Safety-Critical Continuous Control Tasks," 2019 International Conference on Robotics and Automation (ICRA), 2019, pp. 4298-4305.
[2] W. Xiao and C. Belta. High order control barrier functions. IEEE Transactions on Automatic Control, 67(7):3655–3662, 2021.