Safe Reinforcement Learning through Buffer and Barrier Functions for Autonomous Driving

Yue Wan ywan3@jh.edu. ,
Zipei Zhao zzhao115@jh.edu

Outline

- Background
  - Autonomous driving
  - Safe reinforcement learning
- Problem Formulation
- Methodology
  - Trust region policy optimization (TRPO)
  - Control barrier functions (CBFs)
  - Gaussian processes (GPs)
  - Buffer Mechanism
  - Integrated Framework
- Experiment
  - Setup
  - Results
- Discussion
- Conclusion

Background

- Autonomous driving is a complex task that requires rapid decisions in dynamic environments.
- Reinforcement learning offers flexibility in dealing with this driving case.
- Real-world scenarios contain unsafe elements that are ignored when RL maximizes the long-term reward.
- Safe RL with control barrier functions (CBFs) improves safety and exploration efficiency in RL.

RL Formulation

The problem is modeled as an Infinite Horizon Markov Decision Process (MDP), defined by the tuple $(S, A, P, r, \rho_0, \gamma)$,

- S is the state space, which includes the positions, velocities, and relative distances of the ego vehicle and surrounding vehicles
- A is the action space, representing the continuous acceleration and steering controls of the ego vehicle
- $P(s' \mid s, a)$ defines the transition dynamics, which describe how the system transitions from state s to state s′ after taking action a
- $r(s, a)$ is the reward function that evaluates the immediate performance of the ego vehicle based on safety, efficiency, and comfort
- ρ_0 is the initial state distribution, representing the starting positions and velocities of the vehicles
- $\gamma \in (0,1)$ is the discount factor, balancing the importance of immediate versus future rewards

Trust Region Policy Optimization (TRPO)

Goal: Optimize policy $\pi_\theta(a \mid s)$ while ensuring stability and monotonic improvement.
Approach: Use a trust region to constrain updates and prevent drastic policy changes.
Usage in Framework: Generates the RL-based action u^{RL}, ensuring safe and efficient learning.

Optimization Objective:

- Maximize cumulative reward: $$\max_\theta \mathbb{E}_{s\sim\rho^\pi, a\sim\pi_\theta}\left[\frac{\pi_\theta(a \mid s)}{\pi_{\theta_{\text{old}}}(a \mid s)} A^{\pi_{\theta_{\text{old}}}}(s, a)\right],$$
- **Key Term**: $A^{\pi_{\theta_{\text{old}}}}(s, a)$: Advantage function (improvement measure for actions).

Stability via Trust Region:

Constrain the Kullback-Leibler (KL) divergence: $D_{\text{KL}}(\pi_{\theta_{\text{old}}} \| \pi_\theta) \leq \delta$

- Prevents large policy updates.
- δ: Threshold controlling update size.

Control Barrier Functions

- **Purpose**: Ensure system safety by keeping the state within a predefined **safe set** C.
- Definition of Safe Set: $C = \{s \in S : h(s) \geq 0\}$.
    - $h(s) \geq 0$: Safe State.
    - $h(s) \leq 0$: Unsafe State.

- **Goal**: Enforce forward invariance, ensuring the system remains in the safe set over time.

Safety Constraint:

- Control input a must satisfy: $\sup_{a \in A} \left[\frac{\partial h(s)}{\partial s}(f(s) + g(s)a) + \alpha h(s) \right] \geq 0$
    - Components:
        - $f(s)$: System Dynamics
        - $g(s)$: Control Input Effect
        - $\alpha > 0$: Tunable Safety parameter

Safe Action via Quadratic Programming:

- If the RL-proposed action u^{RL} is unsafe, CBF computes a safe action:

$$u^{\text{safe}} = \arg\min_a \|a - u^{\text{RL}}\|^2, \quad \text{s.t.} \quad \frac{\partial h(s)}{\partial s}(f(s) + g(s)a) + \alpha h(s) \geq 0.$$

- Ensures minimal deviation from u^{RL} while maintaining safety.

Role in framework:

- Real-Time Safety Filter:
    - Monitors and adjusts actions proposed by the RL policy.
    - Ensures every executed action satisfies safety constraints.

- Enables Safe Exploration:
    - Supports reinforcement learning without compromising critical safety.

Gaussian Processes (GP)

Purpose:

- approximate unknown functions and quantify uncertainty
- System Dynamics are defined as: $s_{t+1} = f(s_t) + g(s_t)a_t + d(s_t)$
    - $f(s_t)$ and $g(s_t)$ are the known nominal dynamics
    - $d(s_t)$ is the unknown component that needs to be modeled
- For any state s, the GP provides: $d(s) \sim \mathcal{GP}(\mu_d(s), k(s, s'))$.
    - $\mu_d(s)$: mean function
    - $k(s, s')$: covariance function
- GP Predictions
    - $\mu_d(s) = k^\top (K + \sigma^2_{\text{noise}} I)^{-1} y,\ \sigma_d^2(s) = k(s, s) - k^\top (K + \sigma^2_{\text{noise}} I)^{-1} k$
    - Aims: Refine the safety constraints enforced by the Control Barrier Function (CBF)
    - The mean prediction $\mu_d(s)$ and the uncertainty $\sigma_d(s)$ are incorporated into the CBF's constraint:

$$h(f(s) + g(s)a + \mu_d(s) - k\sigma_d(s)) + \alpha h(s) \geq 0$$

- Benefits
    - Handle partially modeled dynamics and account for uncertainties
    - Ensuring robust safety guarantees in dynamic

Dual Buffer Mechanism

Buffer Types

- **Safe Buffer (Buf_S)**: Stores transitions where RL policy actions were safe (no CBF intervention needed)
- **Collision Buffer (Buf_C)**: Stores transitions where CBF corrected unsafe RL policy actions

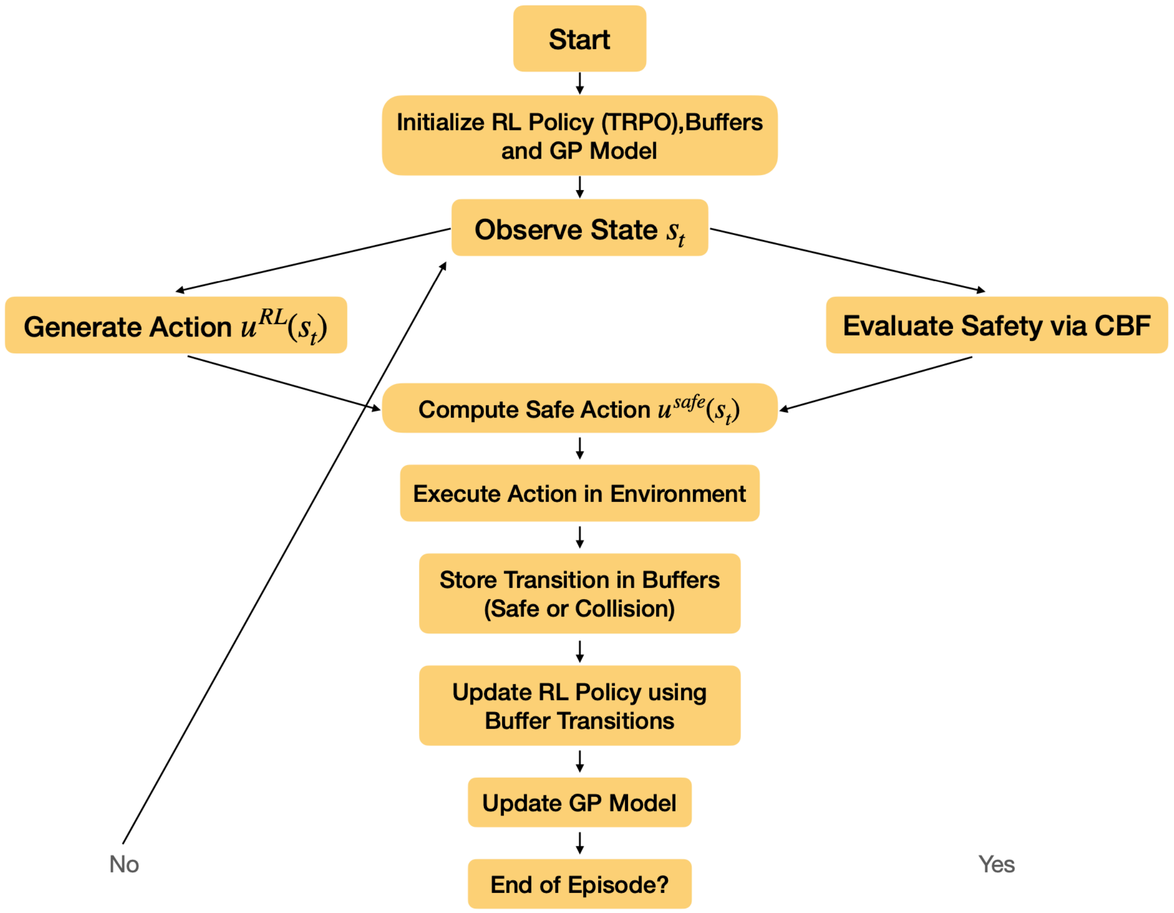
Transition Storage Logic

- If $u^{\text{RL}} = u^{\text{Safe}} \rightarrow$ Store in Safe Buffer
- If $u^{\text{RL}} \neq u^{\text{safe}} \rightarrow$ Store in Collision Buffer

Policy Update Process

- Samples drawn from both buffers: $\mathscr{B} = \mathscr{B}_S \cup \mathscr{B}_C$
- Collision buffer transitions weighted more heavily to discourage unsafe actions
- Modified loss function includes penalties for collision buffer transitions

Flowchart of purposed framework



TRPO-CBF with Buffer Mechanism Algorithm

Algorithm 1 RL-CBF Algorithm with Buffer Mechanism

```
Initialize: RL policy π_0^{RL}, GP model, safe buffer (Buf_S), collision buffer (Buf_C), and state s_0 ∼ ρ_0.
for each episode do
   for each timestep t do
      Generate action u_0^{RL}(s_t) from π_0^{RL}.
      Solve for u_0^{CBF}(s_t) (Equation 8).
      Deploy u_0(s_t) = u_0^{RL}(s_t) + u_0^{CBF}(s_t).
      Observe (s_t, u_0, r_t, s_{t+1}).
      if u_0^{RL}(s_t) = u_0(s_t) then
         Store in Buf_S.
      else
         Store in Buf_C.
      end if
      Update GP model using Equation 11 and 12.
   end for
   Sample transitions from Buf_S and Buf_C for minibatch B.
   Update π_k^{RL} using modified loss (Equation 15).
   Train approximation u_{φ_k}^{bar} for prior CBF controllers.
   for each timestep t do
      Generate action u_k^{RL}(s_t) + u_{φ_k}^{bar}(s_t).
      Solve for u_k^{CBF}(s_t) (Equation 8).
      Deploy u_k(s_t) = u_k^{RL}(s_t) + u_{φ_k}^{bar}(s_t) + u_k^{CBF}(s_t).
      Observe and store transitions in Buf_S or Buf_C.
   end for
   Update GP model and increment k.
end for
Return: π_k^{RL}, u_{φ_k}^{bar}, u_k^{CBF}.
```

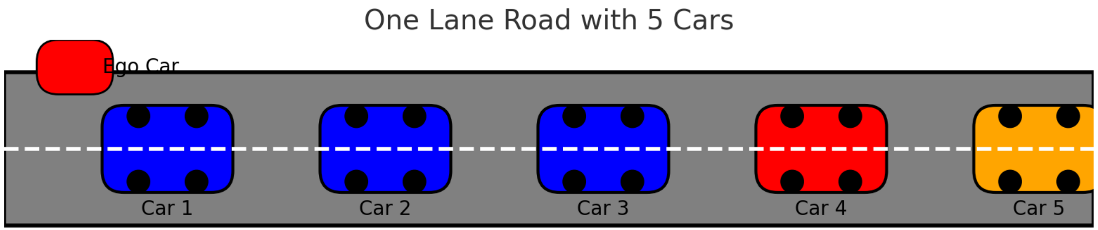
$$u^{\text{safe}} = \arg\min_a \|a - u^{\text{RL}}\|^2, \quad \text{s.t.} \quad \frac{\partial h(s)}{\partial s}(f(s) + g(s)a) + \alpha h(s) \geq 0. \tag{8}$$

$$\mu_d(s) = k^\top (K + \sigma^2_{\text{noise}} I)^{-1} y, \tag{11}$$

$$\sigma^2_d(s) = k(s,s) - k^\top (K + \sigma^2_{\text{noise}} I)^{-1} k, \tag{12}$$

$$L(\theta) = \mathbb{E}_{(s_t,a_t,r_t)\in\mathcal{B}} \left[\nabla_\theta \log \pi_\theta(a_t|s_t) A^\pi(s_t,a_t)\right] - \lambda \mathbb{E}_{(s_t,a_t,r_t)\in\mathcal{B}_C} \left[\|a_t - u_t^{\text{safe}}\|^2\right], \tag{15}$$

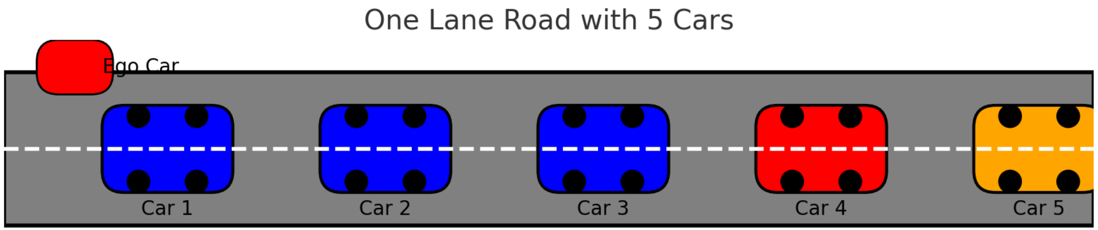
Experimental Setup



Simulated Car Following

- Consider a chain of five cars following each other on a straight road.
- Control the acceleration and deceleration of the 4th car in the chain.
- Train a policy to maximize fuel efficiency during traffic congestion while avoiding collisions.
- Car dynamics: $$\begin{bmatrix} \dot{s}^{(i)} \\ \dot{v}^{(i)} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & -k_d \end{bmatrix} \begin{bmatrix} s^{(i)} \\ v^{(i)} \end{bmatrix}}_{f(s_t)} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix} a}_{g(s_t)a},$$
- The 4th car has access to every other cars' position, velocity and acceleration
- For the fourth car, $k_d = 0,$ meaning the crude model assumes no natural damping in the velocity.

Experimental Setup



Reward Function

$$r = -\sum_{t=1}^{T}[v_t^{(4)}\max((a_t^{(4)}),0) + \sum_{i=3}^{4} G_i(\frac{500}{s_t^{(i)} - s_t^{(i+1)}})]$$

Where $$G_m(x) = \begin{cases} |x| \text{ if } s^{(m)} - s^{(m+1)} \leq 3 \\ 0 \text{ otherwise} \end{cases}$$

- The above function optimizes fuel efficiency and encourages cars to keep a 3-meter distance from others.

With the buffer, the reward is updated that

$$y_j = \begin{cases} r_{j+1} \text{ if sample is from } Buf_c \\ r_{j+1} + \gamma r \text{ if sample is from } Buf_s \end{cases}$$

Experimental Results

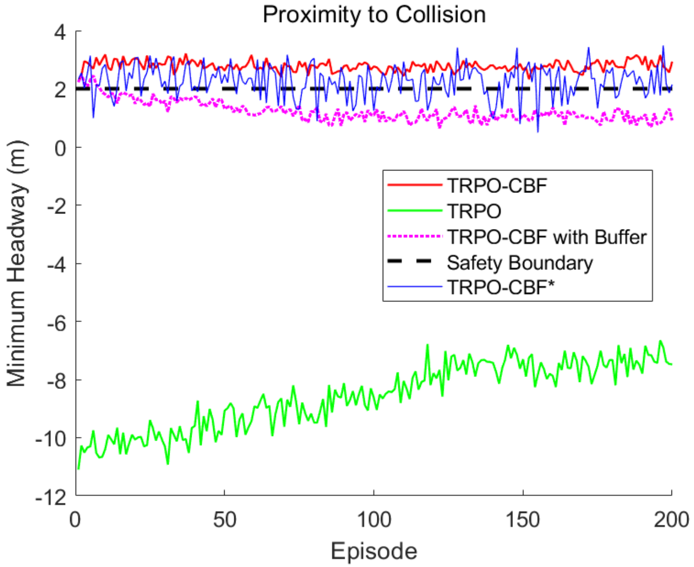
Safety Performance Comparison

Experiment Setup

- Compared three algorithms: TRPO, TRPO-CBF, TRPO-CBF with buffers
- 200 episodes × 4 runs
- Metric: Proximity to collision (minimum headway)

Key Findings

- Basic TRPO: Consistently violated safety
- TRPO-CBF: Successfully maintained safety
- TRPO-CBF with buffers: Failed to stay within safety set
- TRPO-CBF* (reproduced): Larger fluctuations, occasionally unsafe



Experimental Results

Reward performance Analysis

Performance Comparison

TRPO-CBF:

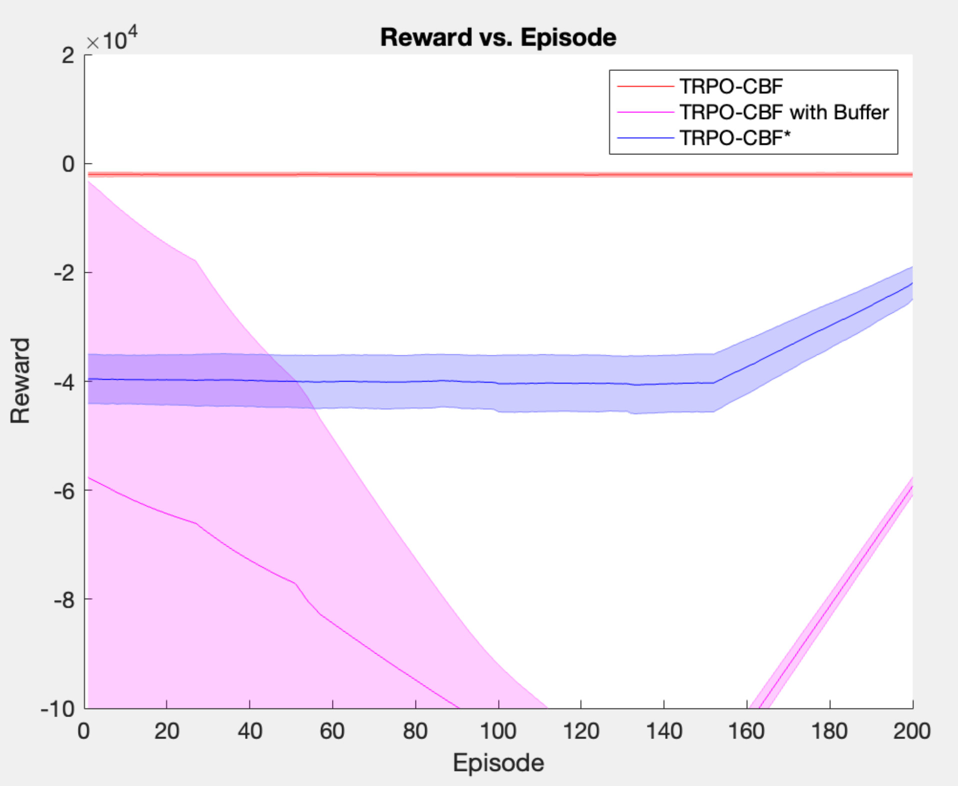
- Highest reward values
- Better stability
- Successful convergence

TRPO-CBF with buffers:

- High initial instability
- No convergence within 200 episodes

TRPO-CBF*(reproduced):

- Lower average rewards
- Larger fluctuations
- No stable convergence



Discussion & Future Work

Key Findings & Limitations

- **Safety Achievement:** CBF successfully prevents collisions in MDP
- **Best Performance:** TRPO-CBF shows optimal results in both safety and rewards
- **Buffer Limitation:** Additional buffers don't improve performance
  - Potential issue with loss function parameters
  - Suboptimal balance between safe/unsafe transitions
- **Computing Impact:** Hardware differences show minimal effect on results

Challenges & Future Direction

Current Limitations:

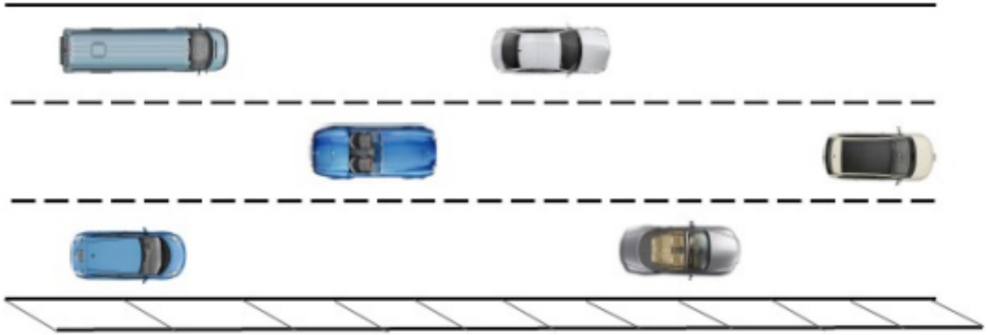
- Training duration insufficient (>200 episodes needed)
- Limited to acceleration/deceleration only
- Single-lane scenario only

Planned Improvements:

- Optimize buffer mechanism parameters
- Extend training duration for convergence
- Implement multi-lane traffic scenarios
- Test performance with additional vehicle actions

Discussion & Future Work

More situations…



Thank you!