Proceedings of Machine Learning Research 1–14, 2025 Machine Learning in Feedback Systems

Uncertainty-Aware Collision Avoidance through Safe Reinforcement Learning

Doris Xu LX253@CORNELL.EDU
Menghan Xu MX253@CORNELL.EDU
Wenjie Guan WG285@CORNELL.EDU
Cornell University

Abstract

Navigating unsignalized intersections remains a critical challenge for autonomous vehicles due to the unpredictability of human drivers and perceptual uncertainties. Traditional planning methods often struggle to balance safety guarantees with navigational efficiency in such dynamic environments. In this work, we propose an uncertainty-aware Safe Reinforcement Learning (RL) framework formulated as a Constrained Markov Decision Process (CMDP). Our core approach utilizes Lagrange Proximal Policy Optimization (LPPO) to dynamically balance task rewards against hard safety constraints. To enhance sample efficiency and robustness, we systematically evaluate three extensions: integrating Behavior Cloning (BC) to leverage expert demonstrations, incorporating a Barrier Force Function (BFF) for proactive safety filtering, and employing a Residual Model Predictive Control (MPC) architecture. We validate these methods through extensive experiments in a high-fidelity simulation and deploy the learned policies on a hardware-in-the-loop testbed using Clearpath Jackal robots. Results demonstrate that our LPPO-based policies achieve a 100% success rate across diverse scenarios, including zero-shot generalization to out-of-distribution agent behaviors. While the residual MPC framework struggled with computational inefficiency and limited control authority, the integration of behavior cloning significantly improved training stability and navigational efficiency, enabling smoother and faster trajectory completion. The code and simulation environment are available at `https://github.com/menghan-xu/safe-rl-intersection`.

1. Introduction

Autonomous vehicles (AVs) have made promising strides into the real world. Kodiak Robotics has been operating driverless trucks in West Texas Kodiak Robotics (2024), and Waymo has expanded to over 250,000 paid rides per week in the US Waymo (2025). Despite recent advances, there are still accidents due to a variety of causes including sensing and perception mistakes, new environments, and unexpected agent behaviors. A recent crash analysis concluded that AVs are more likely than human driven vehicles to be involved in collisions at dusk or down, when the sensors and cameras are more likely to cause perception errors Abdel-Aty and Ding (2024), which can include false positive or false negative detections from deep learning detectors, noisy velocity measurements, or incorrect agent intent inference, all of which can lead to unsafe ego vehicle trajectories. Planning with safety assurance to ensure real time collision avoidance under these many cases of uncertainty remains a critical challenge.

Most existing planners either assume deterministic obstacle behavior or heavily rely on worst-case approximations, which often results in undesirable freezing or solver infeasibility Trautman

and Krause (2010). More recent methods have incorporated probabilistic modeling, but often at the cost of real time feasibility Dai et al. (2018); Nair et al. (2024).
In this work, we propose an uncertainty-aware safe reinforcement learning (RL) framework that explicitly incorporates tracking uncertainty into the cost function, mitigates uncertainty in obstacle intent during training through randomly sampled agent-intent scenarios, and yields collision-avoidant behavior. Specifically, our contributions are:

- An uncertainty aware safe RL path planning framework formulated as a Lagrange Proximal Policy Optimization (LPPO) model, as well as 3 of its derivatives (LPPO with Behavioral Cloning, LPPO with Barrier Force Function, and LPPO with Model Predictive Control).
- Extensive validation in both simulation environments as well as on a hardware system, and a comparison of performance across the different derivatives of the safe RL LPPO solution.

2. Related Work

In Hoel et al. (2020), Hoel et al. used an ensemble of DQNs to produce discrete decisions in the action space for a multi-agent highway driving setting. In addition, several papers focused on developing safe policies for autonomous driving using probabilistic approaches. For example, Bouton et al. Bouton et al. (2019) and Alshiekh et al. Alshiekh et al. (2018) used shielding: a model-checking temporal-logic specs to derive admissible actions and constrain RL, yielding probabilistic safety guarantees. In an unsignalized left-turn study, shielding preserved safety while improving efficiency over rule-based baselines. Wang et al. Wang et al. (2022) proposed a hybrid imitation learning and reinforcement learning decision-making framework for determining whether to enter a roundabout using camera images, achieving improved "go/wait" performance over supervised baselines. Dalal et al. Dalal et al. (2018) introduced a safety layer for continuous action RL that analytically corrects unsafe actions using a learned linearized safety model. Jayant et al. Jayant and Bhatnagar (2022) proposes a model-based constrained PPO method that learns environment dynamics online and uses a Lagrangian relaxation to enforce safety constraints, achieving higher sample efficiency and fewer hazard violations compared to existing constrained RL baselines.

3. Problem Formulation

3.1. Scenario Description

We consider an unsignalized intersection scenario involving the interaction between an autonomous ego vehicle and a human-driven agent vehicle. As illustrated in Fig. 1, the road geometry consists of two orthogonal lanes intersecting at a central conflict zone. The ego vehicle travels longitudinally along a fixed lane, while the agent vehicle approaches the intersection from a perpendicular direction.

The complexity of this scenario arises from the variability in the human agent's behavior. The agent may execute one of three latent intents: *keeping straight*, *turning left*, or *turning right*. The ego vehicle must infer the potential conflict risks solely from observations and navigate through the intersection efficiently without explicit knowledge of the agent's future trajectory.

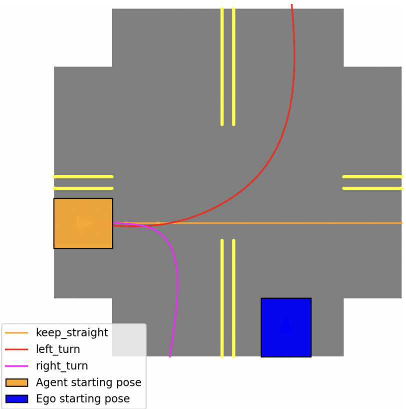


Figure 1: Intents in the intersection.

The objective is to compute a control policy that ensures collision-free passage while optimizing for traffic flow and ride comfort.

3.2. Mathematical Formulation

To address the safety-critical nature of this interaction, we formulate the autonomous navigation task as a Constrained Markov Decision Process (CMDP), defined by the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, C, \gamma)$.

3.2.1. STATE AND ACTION SPACE

The state $s_t \in \mathcal{S} \subseteq \mathbb{R}^8$ at time step t is constructed to capture the ego vehicle's longitudinal kinematics and the agent's observable state under uncertainty:

$$s_t = [y_{\text{ego}}, v_{\text{ego}}, x_{\text{agent}}, y_{\text{agent}}, v_{x,\text{agent}}, v_{y,\text{agent}}, \sigma_{x,\text{agent}}, \sigma_{y,\text{agent}}]^\top. \tag{1}$$

Here, y_{ego} and v_{ego} denote the longitudinal position and velocity of the ego vehicle. The agent's state includes its position $(x_{\text{agent}}, y_{\text{agent}})$, velocity components $(v_{x,\text{agent}}, v_{y,\text{agent}})$, and the associated perceptual uncertainty represented by the standard deviations (σ_x, σ_y).

The action space $\mathcal{A} \subseteq \mathbb{R}$ consists of the continuous longitudinal acceleration command for the ego vehicle, $a_t \in [-a_{\max}, a_{\max}]$. The system dynamics P are modeled using a discrete-time double integrator:

$$y_{t+1} = y_t + v_t \Delta t + \frac{1}{2} a_t \Delta t^2, \tag{2}$$

$$v_{t+1} = v_t + a_t \Delta t, \tag{3}$$

where Δt represents the control time step.

3.2.2. REWARD FUNCTION

To encourage efficient navigation while ensuring passenger comfort and adherence to traffic regulations, the reward function $r(s_t, a_t)$ is designed as a weighted sum of progress incentives, penalties for rule violations, and terminal sparse rewards:

$$r(s_t, a_t) = r_{\text{nav}} + r_{\text{comfort}} + r_{\text{sparse}}, \tag{4}$$

where the navigation and comfort components are defined as follows:

$$r_{\text{nav}} = w_{\text{prog}} \cdot (v_{\text{ego}} \Delta t) - w_{\text{time}} \cdot \Delta t - w_{\text{speed}} \cdot \mathcal{L}_{\text{speed}} \tag{5}$$

$$r_{\text{comfort}} = -w_{\text{comf}} \cdot a_t^2. \tag{6}$$

Here, $w_{(\cdot)}$ denotes the weighting coefficient for each term. The overspeed penalty $\mathcal{L}_{\text{speed}}$ penalizes velocities exceeding the limit v_{limit}:

$$\mathcal{L}_{\text{speed}} = \max(0, v_{\text{ego}} - v_{\text{limit}})^2. \tag{7}$$

The sparse reward term r_{sparse} accounts for terminal states, providing a large positive reward R_{goal} upon reaching the target and a penalty C_{crash} in the event of a collision:

$$r_{\text{sparse}} = \begin{cases} R_{\text{goal}}, & \text{if } y_{\text{ego}} \geq y_{\text{target}} \\ -C_{\text{crash}}, & \text{if collision occurred} \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

3.2.3. COST FUNCTION (SAFETY)

To ensure safety under perceptual uncertainty, we define a cost function based on a conservative safety boundary. Let R_{ego} and R_{agent} denote the collision radii (half-diagonals) of the ego and agent vehicles, respectively. For the Jackal robot, we set $R_{\text{ego}} \approx 0.33\,\text{m}$.

We define the squared Euclidean distance between the vehicles, d_{actual}, and a conservative safety threshold, d_{cons}, which incorporates the agent's positional uncertainty (σ_x, σ_y):

$$d_{\text{actual}} = (x_{\text{ego}} - x_{\text{agent}})^2 + (y_{\text{ego}} - y_{\text{agent}})^2, \tag{9}$$

$$d_{\text{cons}} = \left(R_{\text{ego}} + R_{\text{agent}} + \sqrt{\sigma_{x,\text{agent}}^2 + \sigma_{y,\text{agent}}^2} \right)^2. \tag{10}$$

The safety cost $c(s_t)$ imposes a continuous penalty when the actual distance breaches this conservative boundary, supplemented by a hard penalty upon physical collision:

$$c(s_t) = w_{\text{safe}} \cdot \max(0, d_{\text{cons}} - d_{\text{actual}}) + c_{\text{collision}}, \tag{11}$$

where $c_{\text{collision}} = C_{\text{limit}}$ if $d_{\text{actual}} < (R_{\text{ego}} + R_{\text{agent}})^2$, and 0 otherwise.

4. Methodology

In this section, we detail the algorithmic framework proposed to solve the safety-critical intersection navigation task. We formulate the problem as a Constrained Markov Decision Process (CMDP) and employ Lagrangian Proximal Policy Optimization (LPPO) as the core learning algorithm.

To address the challenges of sample efficiency, safety assurance, and model utilization, we extend the vanilla LPPO framework with three distinct mechanisms: (1) Behavior Cloning (BC) to leverage expert demonstrations; (2) a Barrier Force Function (BFF) for physics-informed safety filtering; and (3) a Model Predictive Control (MPC) residual framework.

4.1. Lagrange Proximal Policy Optimization (LPPO)

While standard Proximal Policy Optimization (PPO) Schulman et al. (2017) is effective for continuous control, it optimizes a single scalar reward and lacks an intrinsic mechanism to satisfy hard safety constraints. To address this, we adopt a Lagrangian relaxation approach Ray et al. (2019), which transforms the constrained optimization problem into an unconstrained dual problem.

Our framework utilizes a dual-head Critic architecture that estimates two distinct value functions: $V_R(s)$ for the task reward (navigation progress) and $V_C(s)$ for the safety cost (collision risk). During the policy update, we compute a combined Lagrangian advantage, A_t^{Lag}, defined as:

$$A_t^{\text{Lag}} = A_t^R - \lambda \cdot A_t^C \tag{12}$$

where A_t^R and A_t^C are the Generalized Advantage Estimations (GAE) for the reward and cost, respectively. The Lagrange multiplier, $\lambda \geq 0$, acts as a dynamic penalty coefficient. It is updated via dual gradient ascent based on the violation of the preset cost limit: if the agent's expected cost exceeds the safety threshold, λ increases, forcing the policy to prioritize safety over progress; conversely, if the agent operates safely, λ decreases to encourage efficiency. This mechanism ensures the agent learns to balance the trade-off between reaching the target and maintaining a safe distance from dynamic obstacles.

4.2. LPPO with Behavior Cloning (LPPO+BC)

While LPPO enables the agent to learn safety constraints through interaction, purely online learning can be sample-inefficient, especially in high-dimensional continuous action spaces. To accelerate the learning process and ensure the policy remains grounded in realistic driving kinematics, we incorporate Behavior Cloning (BC) Pomerleau (1988) as an auxiliary supervised learning objective.

Since the expert ego dataset $\mathcal{D}_E = \{(s_i^*, a_i^*)\}_{i=1}^N$ is finite and does not cover the entire state space explored by the agent, we do not attempt to match the agent's current online states to expert states. Instead, we sample mini-batches directly from $\mathcal{D}_E$ during the policy update phase. We minimize the Mean Squared Error (MSE) between the expert's recorded action a^* and the action predicted by the current policy μ_θ given the expert's state s^*.

The augmented objective function is formulated as:

$$\mathcal{L}_{\text{total}}(\theta) = \mathcal{L}_{\text{LPPO}}(\theta) + w_{BC} \cdot \mathbb{E}_{(s^*,a^*)\sim\mathcal{D}_E}\left[\|\mu_\theta(s^*) - a^*\|_2^2\right] \tag{13}$$

where w_{BC} is a scalar hyperparameter weighing the imitation term. By optimizing this combined loss, the agent learns to satisfy safety constraints via the LPPO Lagrangian term while simultaneously mimicking human-like acceleration profiles via the BC term. This acts as a regularizer, preventing the policy from converging to safe but unnatural behaviors (e.g., stopping indefinitely).

4.3. LPPO with Barrier Force Function (LPPO+BFF)

To enhance safety guarantees during training and deployment, we integrated a Barrier Force Function (BFF) mechanism into our Lagrangian PPO framework, following the approach proposed in Zhang et al. (2023). The BFF modifies the policy structure by decomposing the control action as

$$u_k = v_k + \rho \nabla_v B_k(v_k) + K \nabla_x B_k(x_k),$$

where v_k denotes the nominal control input produced by the policy, and the additional terms represent repulsive forces from logarithmic barrier functions on control and dynamics constraints. Specifically, $\rho \nabla_v B_k(v_k)$ enforces acceleration limits while $K \nabla_x B_k(x_k)$ generates collision avoidance forces that increase as the ego vehicle approaches obstacles. This formulation acts as a safety filter that corrects potentially unsafe actions in real-time without modifying the reward or cost functions, allowing the agent to explore more safely during training while maintaining provable stability and convergence.

The BFF approach is complementary to our Lagrangian constraint formulation, providing two layers of safety: the Lagrangian multiplier reactively penalizes constraint violations through the cost function, while barrier forces proactively prevent violations during action selection. We set the barrier gains to $\rho = 0.2$ and $K = 0.5$ to balance safety and performance.

4.4. Model Predictive Control (MPC) Framework

To investigate the potential of combining model-based control with learning-based policies, we explored a Model Predictive Control (MPC) framework in two capacities: as a standalone baseline and as a safety-enhancing component within a residual learning architecture.

4.4.1. NON-LINEAR MPC FORMULATION

We first formulated a Non-linear MPC (NMPC) problem to serve as a deterministic baseline. We utilize the CasADi framework Andersson et al. (2019) with the IPOPT solver to optimize a finite-horizon trajectory of length N. At each time step t, the controller solves for the optimal control sequence $u_{t:t+N-1}$ to minimize the following objective:

$$J_{\text{MPC}} = \sum_{k=0}^{N-1} \left(w_{\text{track}} (v_k - v_{\text{limit}})^2 + w_{\text{comf}} a_k^2 + w_{\text{slack}} s_k^2 \right) \tag{14}$$

subject to the double integrator dynamics and actuator limits ($a_{\text{min}} \leq a_k \leq a_{\text{max}}$).

Let $d_{\text{actual},k}$ be the predicted squared Euclidean distance between the ego and agent at step k, and $d_{\text{cons},k}$ be the squared conservative safety threshold incorporating the agent's uncertainty σ. The safety constraint is formulated as:

$$d_{\text{actual},k} \geq d_{\text{cons},k} - s_k, \quad \forall k \in [0, N-1] \tag{15}$$

where $s_k \geq 0$ are slack variables. This formulation allows the solver to find valid trajectories even when the conservative safety boundary is slightly breached due to noise, heavily penalizing such violations via the w_{slack} term in the objective.

4.4.2. RESIDUAL LPPO+MPC FRAMEWORK

Building upon the NMPC baseline, we proposed a Residual Reinforcement Learning framework Johannink et al. (2019) aimed at combining the theoretical safety guarantees of MPC with the adaptability of RL.

In this hybrid architecture, the MPC computes a nominal control action a_{MPC} based on the simplified kinematic model and the conservative constraints described above. The LPPO agent, observing the full state space, learns a residual correction term a_{res}. The final action applied to the vehicle is:

$$a_{\text{final}} = \text{clip}(a_{\text{MPC}} + w_{\text{res}} \cdot a_{\text{res}}, -a_{\max}, a_{\max}) \tag{16}$$

where w_{res} is a residual scaling factor. This structure is designed to allow the RL agent to compensate for model mismatches and solver latency, potentially improving navigation efficiency while the MPC component anchors the behavior within a physically feasible region.

5. Experiments

5.1. Experiment Setup

5.1.1. PHYSICAL TESTBED

To validate the proposed approach in a realistic setting, we utilized a hardware-in-the-loop testbed at the Autonomous Systems Lab (ASL). As shown in Fig. 2, the physical setup consists of a $3\,\text{m} \times 3\,\text{m}$ indoor area. The intersection geometry and lane boundaries are projected directly onto the floor surface using an overhead projector, providing a visual reference for the vehicle operators.

We employed two Clearpath Jackal Unmanned Ground Vehicles (UGVs) to simulate the traffic participants:

- Ego Vehicle: Controlled by the autonomous navigation policy.
- Agent Vehicle: Acts as the human-driven interactive vehicle, controlled by a human driver.

Both robots are tracked using an overhead motion capture system. An Extended Kalman Filter is used to filter these raw mocap measurements, and produce the low noise state estimates used during policy deployment. This setup allows us to bridge the gap between simulation and reality by testing the algorithms on physical dynamics.

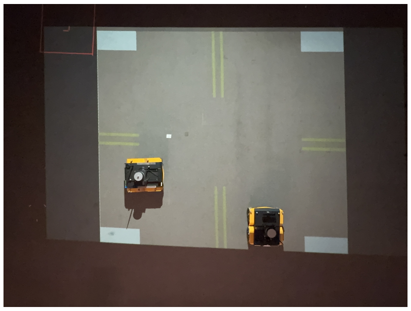


Figure 2: Artificial 3m × 3m intersection setup used for all experiments.

5.1.2. SIMULATION ENVIRONMENT

To ensure the transferability of the learned policy to the physical platform, we constructed a simulation environment that mirrors the dimensions and dynamics of the physical testbed. The simulation operates within a $3\,\text{m} \times 3\,\text{m}$ grid centered at the origin $(0, 0)$. The control loop frequency is set to $10\,\text{Hz}$ ($\Delta t = 0.1\,\text{s}$).

Vehicle Constraints: The simulated vehicle parameters are strictly calibrated to match the Clearpath Jackal specifications. The vehicle has a length of $0.508\,\text{m}$ and a width of $0.430\,\text{m}$. For collision checking, we approximate the vehicle geometry using a bounding circle with a radius of $R_{\text{robot}} \approx 0.333\,\text{m}$. To reflect physical actuator limits, the maximum velocity is constrained to $v_{\text{limit}} = 1.5\,\text{m/s}$, and the maximum acceleration is limited to $a_{\text{max}} = 3.2\,\text{m/s}^2$ and $a_{\text{min}} = -3.2\,\text{m/s}^2$.

Task Initialization: In each episode, the ego vehicle is tasked with traveling straight through the intersection. The starting position is uniformly sampled from a circular region with a radius of $0.15\,\text{m}$ centered at $(0.5, -1.25)$ to introduce initial state variance. The episode is considered successful when the ego vehicle crosses the longitudinal threshold $y_{\text{target}} = 1.25\,\text{m}$ without collision.

5.2. Data Collection

During the data collection stage, both the ego and agent were operated by human drivers. The ego vehicle's driving behavior serves as an expert demonstration of preferred collision-avoidance behavior. Scenarios in which the agent travels straight or executes a left turn are considered high-risk, as these maneuvers result in potential lane conflicts with the ego vehicle; accordingly, six repetitions of each of these scenarios were conducted. In contrast, during the right-turn scenario, the agent does not enter the ego vehicle's lane and therefore poses minimal collision risk. As a result, three repetitions of the right-turn scenario were performed. In order to generate a larger dataset for training, we selected a representative trial from each one of the three scenarios, and injected Gaussian noise into the recorded trajectories, such that in each generated trajectory, 99.7% of the noisy states are with 5% deviation from the original states. This augmentation process yielded 100 noisy trajectories for each of the three maneuvers (Straight, Left Turn, and Right Turn), totaling 300 trajectories. We performed an 80%-20% train-test split, resulting in 240 agent trajectories used for training the model.

To evaluate the policy's robustness against out-of-distribution (OOD) agent behaviors, we additionally generated 5 ”zigzag” trajectories for testing. These trajectories were created by applying a sinusoidal wave pattern to the y-coordinate of straight eastbound agent paths while keeping the x-coordinate unchanged. The sine wave oscillation is parameterized by an amplitude of $0.035\,\text{m}$ (corresponding to a lateral deviation of $\pm 3.5\,\text{cm}$) and a wavelength of $0.5\,\text{m}$. The agent's velocities and heading angles were recalculated from these perturbed positions using finite differences to maintain kinematic continuity. Consequently, the final test set consists of 65 trajectories (60 from the standard split plus 5 zigzag cases), allowing us to evaluate both the overall performance and the policy's adaptability across four distinct behavioral categories.

5.3. Evaluation Metrics

To quantitatively assess the performance of the proposed policy, we utilize the following four metrics, covering both safety and efficiency aspects:

- **Success Rate**: The percentage of episodes in which the ego vehicle successfully reaches the target longitudinal position ($y_{\text{ego}} \geq y_{\text{target}}$) without any collision or timeout. This is the primary indicator of task completion capability.

- **Collision Rate**: The percentage of episodes where a collision occurs between the ego vehicle and the agent. This metric directly reflects the safety performance of the policy under uncertainty.

- **Average Reward**: The cumulative reward averaged over all test episodes. This metric serves as a comprehensive evaluation of the policy's quality, balancing navigation progress, safety constraints, and ride comfort.
- **Time-to-Completion**: The average time duration (in seconds) required for the ego vehicle to reach the target. Note that this metric is calculated only over successful episodes to measure the navigation efficiency.

5.4. Simulation Results

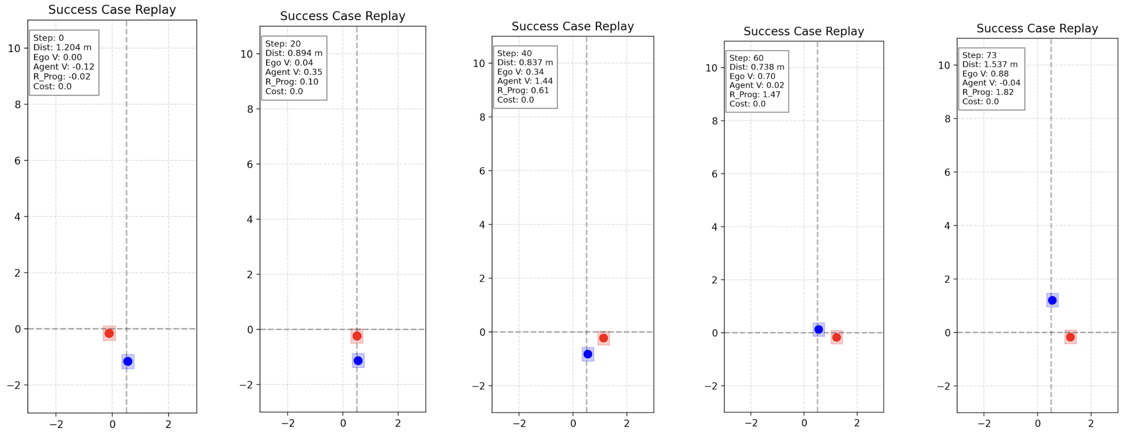
We evaluated our proposed methods against baselines across multiple metrics and scenarios. In this section, we present a detailed analysis of the quantitative performance, training characteristics, and behavioral patterns of the learned policies.

5.4.1. QUANTITATIVE PERFORMANCE AND GENERALIZATION

We trained the models on the training set with manually tuned hyperparameters and selected the policy with the highest stabilized reward. Table 1 summarizes the key performance metrics.

To our satisfaction, with a proper choice of hyperparameters, the Lagrange PPO method (with or without behavior cloning) achieves a **100% success rate and 0% collision rate** across all maneuvers. **Most notably, this includes the "zigzag" scenario, which was completely unseen during training.** This zero-shot generalization capability is particularly compelling; it indicates that the policy has not merely memorized the specific kinematics of the training set but has learned a robust, generalized collision-avoidance strategy capable of handling out-of-distribution (OOD) and erratic agent behaviors.

Figure 3 further illustrates this generalization. In the zigzagging test case, the ego vehicle successfully maintains a safe distance despite the agent's unpredictable directional shifts. We observed that this robustness is sensitive to the collision penalty: when C_{crash} is set too low, the model becomes overly aggressive and fails in these OOD scenarios, highlighting the importance of rigorous safety constraints in learning robust policies.



(*a*) (*b*) (*c*) (*d*) (*e*)

Figure 3: Visualization of the learned policy on an unseen Zigzag trajectory. The blue circle represents the ego vehicle, and the red circle represents the agent vehicle.

Table 1: Performance Comparison (All Methods)

Metric	Case	LPPO	LPPO+BC	LPPO+BFF	MPC	LPPO+MPC
Success Rate	Left	**100%**	**100%**	95.0%	80.0%	97.0%
	Right	**100%**	**100%**	**100.0%**	**100.0%**	**100.0%**
	Str.	**100.0%**	**100.0%**	**100.0%**	0%	0%
	Zig.	**100.0%**	**100.0%**	**100.0%**	0%	0%
	Avg	**100.0%**	**100.0%**	98.5%	55.4%	64.3%
Collision Rate	Left	0%	0%	0%	0%	0%
	Right	0%	0%	0%	0%	0%
	Str.	0%	0%	0%	0%	0%
	Zig.	0%	0%	0%	0%	0%
	Avg	0%	0%	0%	0%	0%
Avg. Reward	Left	**258.3**	246.1	234.7	–	–
	Right	**260.4**	246.7	258.8	–	–
	Str.	**260.3**	246.8	258.7	–	–
	Zig.	**261.2**	244.7	257.7	–	–
	Avg	**260.1**	246.1	251.3	–	–
Time (s)	Left	8.57	**7.31**	12.21	9.46	9.15
	Right	6.43	4.08	6.43	2.00	**1.90**
	Str.	7.91	**6.25**	7.90	0.00	0.00
	Zig.	7.90	**7.34**	8.28	0.00	0.00
	Avg	7.70	**6.25**	8.75	**5.31**	5.63

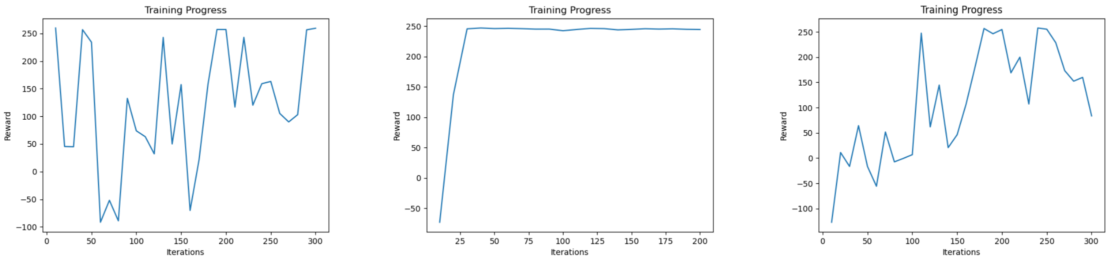
5.4.2. TRAINING EFFICIENCY AND BEHAVIORAL CHARACTERISTICS

It is interesting to note the divergence between the optimization objectives of LPPO and LPPO+BC. While Vanilla LPPO achieves the highest cumulative reward, **LPPO+BC demonstrates the lowest time-to-completion**, allowing the ego vehicle to reach the target most efficiently. This suggests that the pure RL agent maximizes the mathematical reward by adopting a conservative speed to minimize acceleration penalties, whereas the BC agent inherits the efficient driving heuristics of the human expert. Furthermore, as shown in Fig. 4, the inclusion of the BC loss significantly improves training stability and convergence speed.

Beyond metrics, we conducted a qualitative inspection of the agents' behaviors. In high-risk scenarios, we observed distinct strategies:

- **LPPO:** Showed almost no reversing behavior, preferring to adjust forward velocity.
- **LPPO+BC:** Interestingly, it learned to perform slight reversing maneuvers ($\approx$ 10 cm) to yield, despite the expert dataset containing only forward trajectories.
- **LPPO+BFF:** Exhibited the most conservative behavior, frequently reversing by 20-30 cm.

While reversing is atypical for human drivers at intersections, it is a valid strategy within our CMDP formulation to satisfy strict safety constraints without incurring collision costs.



(*a*) (*b*) (*c*)

Figure 4: Reward curves. From left to right: Vanilla LPPO, LPPO + BC, and LPPO + BFF.

5.4.3. Analysis of Model-Based Baselines

The performance of the MPC baseline and the Residual LPPO+MPC framework presents an insightful contrast to the pure learning-based approaches.

First, the MPC baseline achieved a 100% success rate in the Right Turn scenario but failed completely (0%) in Straight and Zigzag scenarios. This dichotomy is expected because the MPC planner operates with a simplified deterministic model that lacks explicit awareness of the dynamic agent's future intent. Its success in the Right Turn scenario is merely an artifact of the road geometry where no conflict exists.

Second, the integration of LPPO as a residual correction term yielded only marginal improvements. The framework failed to rectify the catastrophic failures in the Straight and Zigzag cases. This suggests a limitation in the control authority of the residual agent: when the MPC planner generates a highly unsafe trajectory, the bounded residual term is insufficient to fully override the erroneous command.

Finally, the computational cost proved prohibitive. Training the LPPO+MPC model required over 10 hours on an Apple M1 processor, orders of magnitude slower than pure LPPO, due to the invocation of the non-linear solver at every time step. We conclude that for this specific task involving stochastic human behaviors, a pure end-to-end learning approach is far more efficient and effective.

5.5. Hardware Demonstration

For hardware validation, the best-performing policy from each training session was selected and deployed directly on the physical system; no policy training was conducted on hardware. Each selected policy was implemented as a ROS 2 node. The policy outputs a commanded acceleration along the vehicle's heading direction. This acceleration is converted into a corresponding velocity command using a single-step Euler integration. The resulting velocity is then applied as the linear velocity control input to the robot.

During the hardware demonstrations, we deployed three variants of our policy: vanilla Lagrange PPO, Lagrange PPO with Behavior Cloning, and Lagrange PPO with Barrier Force Function. All models were trained with randomly sampled initial positions to encourage robustness across diverse scenarios. Across all three variants, we consistently observed overly conservative behavior when the agent was positioned closely to the ego vehicle, particularly near the center of the intersection. In these situations, the ego would often respond by reversing to avoid a potential collision, even if the agent posed minimal actual threat. Conversely, when the agent was sufficiently far from the ego, the policy prioritized goal-seeking behavior, which resulted in the ego accelerating assertively through the intersection to maximize reward. This pattern showed a gap in the sim-to-real translation of the current model.

6. Discussion

Our experiments highlight the role of data diversity in training robust safety-critical policies. While Reinforcement Learning requires extensive interaction data to converge, collecting a comprehensive set of expert trajectories on a physical testbed is challenging and often yields repetitive data. Originally, we tried the RL models on these limited expert data, and it turned out none produced a good policy, with the best policy having training success rate around 80% and collision rate about 10%, and the test performance can be even worse. Furthermore, the learning curve is highly unstable, suggesting we can hardly learn useful with such a small training sample. To conquer this issue, we employed a data augmentation strategy that injected Gaussian noise into the expert demonstrations, introduced in Section 5.2. This approach has two benefits. First, we significantly increase the sample size for training and testing. Second, adding moderate noises will provide the model from overfitting, which in turn increases the robustness of our models and improves their generalizability. The success of this strategy was evident in the policy's generalization to the unseen 'zigzag' trajectories, confirming that the agent learned a fundamental collision-avoidance heuristic rather than simply memorizing training instances.

There are some limitations in our work in terms of simulation to real system translation. For example, the RL model assumes a unicycle dynamics model for the Jackal robots, however, there could be some underlying model uncertainty. The real time localization uncertainty could also deviate from the uncertainties the model has seen in the training data. In addition, the lab floor is carpeted, which might have led to unmodeled external disturbances and slippage. In the cases the ego reversed to avoid a collision with the agent in simulations, the ego was stopped manually relatively soon after it started reversing to avoid colliding into the wall. In the future, we will investigate methods to learn the unmodeled disturbances to improve the robustness of our model.

7. Conclusion and Future Work

In this work, we proposed an uncertainty-aware navigation framework for intersections utilizing Lagrange Proximal Policy Optimization (LPPO) to balance navigational efficiency with hard safety constraints. Through extensive simulation, we demonstrated that our LPPO-based policies, particularly when augmented with Behavior Cloning, could achieve a 100% success rate and generalization to mild out-of-distribution agent behaviors. While the Residual MPC framework offered theoretical safety, it struggled with computational inefficiency and limited control authority in highly dynamic scenarios.

Future work will focus on narrowing the simulation-to-reality gap by investigating methods to learn unmodeled dynamics and environmental disturbances online, thereby improving the robustness of the policy against physical uncertainties. Additionally, we aim to refine the cost function and explore advanced safety filters to eliminate unnatural reversing behaviors, ensuring that the autonomous agent interacts more predictably with human drivers.

References

Mohamed Abdel-Aty and Shengxuan Ding. A matched case-control analysis of autonomous vs human-driven vehicle accidents. *Nature communications*, 15(1):4931, 2024.

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Joel A E Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. Casadi: A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36, 2019.

Maxime Bouton, Jesper Karlsson, Alireza Nakhaei, Kikuo Fujimura, Mykel J. Kochenderfer, and Jana Tumova. Reinforcement learning with probabilistic guarantees for autonomous driving, 2019. URL https://arxiv.org/abs/1904.07189. arXiv:1904.07189v2.

Siyu Dai, Shawn Schaffert, Ashkan Jasour, Andreas Hofmann, and Brian Williams. Chance-constrained motion planning for high-dimensional robots. *arXiv preprint arXiv:1811.03073*, 2018.

Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

Carl-Johan Hoel, Krister Wolff, and Leo Laine. Tactical decision-making in autonomous driving by reinforcement learning with uncertainty estimation, 2020. URL https://arxiv.org/abs/2004.10439.

Ashish Kumar Jayant and Shalabh Bhatnagar. Model-based safe deep reinforcement learning via a constrained proximal policy optimization algorithm, 2022. URL https://arxiv.org/abs/2210.07573.

Tobias Johannink, Shikhar Bahl, Ashvin Nair, Jianlan Luo, Avinash Kumar, Philip Dames, Manuel Watter, Jost Springenberg, Klein Schnieders, and Martin Riedmiller. Residual reinforcement learning for robot control. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6023–6029. IEEE, 2019.

Kodiak Robotics. Kodiak launches fully driverless truck operations in west texas. *News report*, 2024. Kodiak Robotics began commercial driverless trucking services on private roads in late 2024.

Siddharth H Nair, Hotae Lee, Eunhyek Joa, Yan Wang, H Eric Tseng, and Francesco Borrelli. Predictive control for autonomous driving with uncertain, multimodal predictions. *IEEE Transactions on Control Systems Technology*, 2024.

Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In *Advances in neural information processing systems*, pages 305–313, 1988.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *OpenAI*, 2019.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Peter Trautman and Andreas Krause. Unfreezing the robot: Navigation in dense, interacting crowds. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 797–803, 2010. doi: 10.1109/IROS.2010.5654369.

Weichao Wang, Lei Jiang, Shiran Lin, Hui Fang, and Qinggang Meng. Imitation learning based decision-making for autonomous vehicle control at traffic roundabouts. *Multimedia Tools and Applications*, 81(28):39873–39889, 2022.

Waymo. Waymo surpasses 250,000 weekly paid rides as robotaxi services expand. *News report*, 2025. Waymo's robotaxis now operate across multiple U.S. cities (Phoenix, SF, LA, Austin, Atlanta), logging over 250,000 paid rides per week.

Xinglong Zhang, Yaoqian Peng, Biao Luo, Wei Pan, Xin Xu, and Haibin Xie. Model-based safe reinforcement learning with time-varying state and control constraints: An application to intelligent vehicles. *arXiv preprint arXiv:2112.11217*, 2023.