2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Reinforcement Learning with Probabilistically Safe Control Barrier Functions for Ramp Merging

Soumith Udatha[1], Yiwei Lyu[1], John Dolan[1]

Abstract— **Prior work has looked at applying reinforcement learning (RL) approaches to autonomous driving scenarios, but the safety of the algorithm is often compromised due to instability or the presence of ill-defined reward functions. With the use of control barrier functions embedded into the RL policy, we arrive at safe policies to optimize the performance of the autonomous driving vehicle through the advantage of a safety layer over the RL methods to ease the design of reward functions. However, control barrier functions need a good approximation of the model of the system. We use probabilistic control barrier functions [4] to account for model uncertainty. Our Safety-Assured Policy Optimization - Ramp Merging (SAPO-RM) algorithm is implemented online in the CARLA [1] Simulator and offline on the US I-80 dataset extracted from the NGSIM Database provided by NHTSA [2]. We further test the algorithm and perform ablation studies of it on the US-101 and exi-D datasets to compare the approaches. The proposed algorithm can also be applied to other driving scenarios by changing the reward and safety constraints.**

I. Introduction

It is well known that real-world dynamics are difficult to capture entirely accurately in simulators [3]. This disparity between the real dynamics and the approximations involved becomes more important when dealing with safety-critical systems. To determine safety of a system, one needs to have a good idea of the world dynamics involved.

The interaction of self-driving cars with other human drivers can be treated as a safety-critical system [4]. In the area of autonomous vehicle control and planning [5][6][7], safety is always the primary focus. Among autonomous driving scenarios requiring interaction with human drivers, ramp merging is relatively simple compared to intersections and lane changing, since the number of vehicles to negotiate with is fewer. Even so, freeway on-ramp merging is a high-risk activity for motor vehicle crashes and conflicts due to the variety of driving styles [8] and the difficulty of reactions at high speeds. Human drivers introduce uncertainty into merge scenarios, which changes the system dynamics, and Autonomous Vehicles (AV) will collide if they can't plan and be safely controlled [9]. It is therefore important that we address model uncertainty while modeling the control and planning of self-driving cars.

Strictly adhering to a rule-based system for autonomous highway behaviors to maintain safety could result in unsafe situations. For example, the autonomous vehicle might instantaneously brake very close to another vehicle, which may not be anticipated by the human drivers of the tailgating vehicles, leading to collision [10]. Further, the situations encountered on freeways are very diverse, so that it's difficult to account for all scenarios while designing a rule-based system. The efficiency and solution feasibility of rule-based systems become even worse [11] in situations with model uncertainty. Hence, we need to take diverse human behaviors into account along with model uncertainty for better safety guarantees. Reinforcement Learning (RL) methods through exploration can help in accounting for the diversity of behaviors observed.

We closely study a highway on-ramp merging scenario, where the goal is to enable the AV to merge with human-driven cars safely and efficiently. Our main contributions are in extending and providing an RL pipeline with probabilistically safe constrained barrier functional safety [4][12][14] which can be extended to any self-driving task in a simulation domain by changing the task-based reward function and task specifications. We perform experiments in CARLA ([1]) and an offline training procedure on the NGSIM dataset for a one-on-one ramp merging case. The emphasis of the paper is not on proposing a novel RL algorithm for autonomous cars, but on understanding and addressing the problem of model uncertainty with Control Barrier Functions (CBFs) ([14]) when combined with a RL pipeline to extend the research on safe RL methods for both offline and online environments.

II. Related Work

CBF-based methods ([15], [16], [17], [33],[34]) are being increasingly used in the control application domain due to their safety guarantees with the forward-invariant property when a solution is feasible. In reality, a solution may not always be feasible, as we deal with approximations of complex non-linear dynamics which are not the same as the real-world dynamics observed and the systems resort to hard-coded recovery control [18].

To address model uncertainty, several works [16][19][35] proposed to employ the CBF approach with noisy system dynamics. Integrating CBF with Model Predictive Control (MPC) is also a common planning method. [20] and [21] introduced MPC-based safety-critical control. However, these works fail to take different driving behavior styles into account. To address this problem, one possible solution is to use CBFs with reinforcement learning-based methods, where the agent trains for all the possible kinds of behaviors, given enough training data and even a black-box model. RL-CLF-CBFs proposed in [22] is used to reduce the uncertainty in plant dynamics. [23] implemented an RL algorithm

[1]Carnegie Mellon University, Pittsburgh, United States.Correspondence to Soumith Udatha: sudatha@andrew.cmu.edu

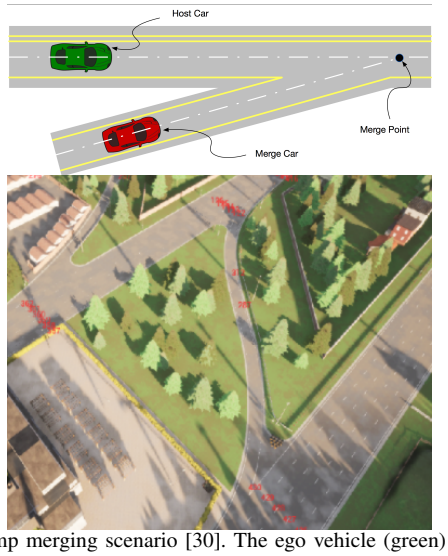

Fig. 1: Ramp merging scenario [30]. The ego vehicle (green) is the host vehicle; the merging vehicle (red) is the ego vehicle, running on the ramp. Figure 1. shows a two dimensional version of ramp-merging that we will be using. The merge is along a curve rather than a straight line.

with Constrained Policy Optimization (CPO) [24] and with linearized constraints of control barrier functions.

Generalized CBFs are introduced in [25] to consider higher-order dynamics, and Parametric CBFs [26] are one of the popular methods to model approximate higher-order dynamics. In our work, we integrate Probabilistic CBFs [4][12][32] to address model uncertainty owing to its ability to use a simpler dynamics model and linear constraints for the optimization problem compared to other methods. [23] is closely relevant prior work, which uses Generalized Discrete CBFs to generate safety certificates and combine with a Model Based Policy Optimization (MBPO) [29] framework for an intersection scenario. We have a similar RL pipeline to [23] but also include uncertainty-accounted system dynamics and Probabilistic CBFs. Further, we test our approach on online environments in CARLA and validate on an offline NGSIM dataset to show the generalization to offline environments.

III. Problem Formulation

A. Setting up the problem and Dynamics

Ramp merging has previously been studied in one dimension. We extended the approach to two dimensions to include a wider range of ramp-merging behaviors and easy generalization to other driving scenarios. The goal is to control the ego vehicle on the ramp to merge safely with the human-driven vehicles (host vehicles) already on the highway. The system dynamics of an uncertain vehicle can be described by double integrators as follows, since acceleration plays a key role in the safety considerations:

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v}_x \\ \dot{v}_y \end{bmatrix} = \begin{bmatrix} 0_{2\times2} & I_{2\times2} \\ 0_{2\times2} & 0_{2\times2} \end{bmatrix} \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} + \begin{bmatrix} 0_{2\times2} & I_{2\times2} \\ I_{2\times2} & 0_{2\times2} \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ \varepsilon_x \\ \varepsilon_y \end{bmatrix} \tag{1}$$

where $x \in \mathscr{X} \subset \mathbb{R}, y \in \mathscr{Y} \subset \mathbb{R}, v \in \mathbb{R}^2$ are the position and linear velocity of each car respectively and $u \in \mathbb{R}^2$ represents the acceleration control input. $\varepsilon \sim \mathcal{N}(\hat{\varepsilon}, \Sigma)$ is a random Gaussian variable with known mean $\hat{\varepsilon} \in \mathbb{R}^2$ and variance $\Sigma \in \mathbb{R}^{2\times2}$, representing the uncertainty in each vehicle's motion.

B. Probabilistic Control Barrier Functions

In [4], an optimization framework with probabilistically safe CBFs was proposed as an alternative to deterministic CBFs with perfect model information to address motion uncertainity, specifically for ramp merging scenarios in one dimension.

Consider the admissible space equation for linear CBF with parameter α and single dimension x ; $\dot{h}(x,u) + \alpha h(x) \geq 0$ with $h = (x_e - x_m)^2 - R^2_{safe}$. By substituting Eq. 1 we have:

$$\dot{h}^s_{em}(x,u) + \alpha h^s_{em}(x) \geq 0;$$

$$\implies 2\Delta x^T_{em}\Delta\varepsilon_{em} \geq -2\Delta x^T_{em}(\Delta v_{em} + u_e\Delta t) - \alpha h^s_{em}(x) \tag{2}$$

where $\Delta x_{em} = x_e - x_m, \Delta v_{em} = v_e - v_m, \Delta\varepsilon_{em} = \varepsilon_e - \varepsilon_m \sim \mathcal{N}(\Delta\hat{\varepsilon}_{em}, \Delta\Sigma_{em})$ for ego vehicle e and each merging vehicle m. $\Phi^{-1}(\cdot)$ is the inverse cumulative distribution function (CDF) of the standard zero-mean Gaussian distribution with unit variance.

We reorganize $\Pr\left(\dot{h}^s_{em}(x,u) + \alpha h^s_{em}(x) \geq 0\right) \geq \eta$ into the form of $\Pr(a^T c \leq b) \geq \eta \Leftrightarrow b - \bar{a}^T c \geq \Phi^{-1}(\eta)||\Sigma^{1/2}c||^2$ from [13] with $a = \Delta\varepsilon_{em}, c = -2\Delta x_{em}, b = 2\Delta x^T_{em}(\Delta v_{em} + u_e\Delta t) + \alpha h^s_{em}(x)$ and eventually get a constraint; $A_{em}u_x \leq b_{em}, \quad A_{em} \in \mathbb{R}^{1\times2}, b_{em} \in \mathbb{R}$ with

$$\begin{aligned} A_{em} &= -2\Delta x^T_{em}\Delta t \\ b_{em} &= 2\Delta x^T_{em}(\Delta v_{em} + \Delta\hat{\varepsilon}_{em}) \\ &+ \alpha h^s_{em}(x) - \Phi^{-1}(\eta)\sqrt{\Delta x^T_{em}\Delta\Sigma_{em}\Delta x_{em}} \end{aligned} \tag{3}$$

where Δt is a time unit. We derive linear control constraints for pairwise chance-constrained safety between ego vehicle and each merging vehicle m. The equations in Eq.3 can be extended to the y-dimension as well for 2-dimensional decoupled CBFs. For a more detailed derivation, we refer the reader to [4]. □

C. Reinforcement Learning with Probabilistically Safe CBFs

For better safety guarantees, we can use a reinforcement learning policy to compute high-level desirable nominal velocity and choose a CBF control as output. However, it is computationally expensive to solve two different optimization problems while interacting with a simulator. Our constrained RL framework solves it with a single constrained optimization with linear probabilistic safety certificates. Using the optimization framework defined in [24] for constrained RL problems, we extend the constraints obtained in both dimensions from subsection B for the total return objective $\mathscr{G}$ and set of safe states $\mathscr{C}$ as:

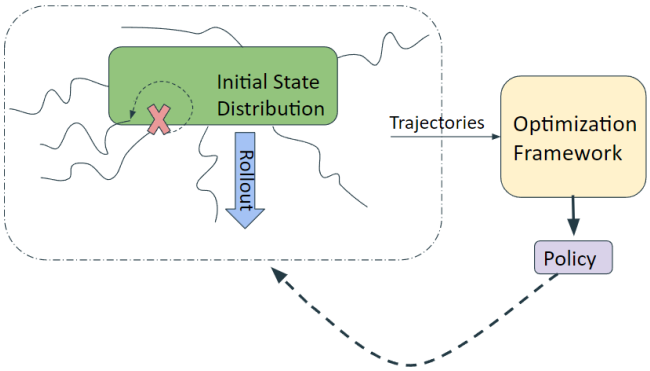

Fig. 2: **Reinforcement Learning Pipeline**: We initialize rollouts from random initial states and run an optimization on the set of collected trajectories to train a policy for each epoch. If the trajectory is invalid, we sample a different trajectory.

$$\begin{aligned} &\min_{\theta} L_a = \mathbb{E}_{a\sim\pi,s\sim C}[\mathscr{G}] \\ s.t \quad & U_{min} \leq u_e \leq U_{max} \\ & J_{c_x} = \mathbb{E}_{\approx\sim\pi}[A_{em_x} u_{e_x}] \leq b_{em_x} \\ & J_{c_y} = \mathbb{E}_{\approx\sim\pi}[A_{em_y} u_{e_y}] \leq b_{em_y} \\ & D_p(\theta,\theta_k) = \frac{1}{2}\Delta\theta^T H \Delta\theta \leq \delta \end{aligned} \tag{4}$$

Using the RL objective in Eq. 4, we use an approximate actor-critic constrained policy gradient formulation in [23] to solve this optimization. The critic objective is $L(\phi) = E_{s_t\sim\mathscr{C}}[\frac{1}{2}(\mathscr{G} - V(s_t,\phi))^2]$, where ϕ are the parameters of the critic network and $V(s_t,\phi)$ is the critic's value function.

The Reinforcement Learning objective of the actor in Eq.4 can be transformed into a dual objective as:

$$\begin{aligned} &\min_{\Delta\theta} g^{\mathbb{T}}\Delta\theta \\ &s.t.\ z + C^T\Delta\theta \leq 0 \\ &\tilde{D_p} \sim \frac{1}{2}\Delta^{\mathbb{T}} H \Delta\theta \leq \delta \end{aligned} \tag{5}$$

where H is the hessian from the trust region condition, $g = \frac{dL_a}{\partial\theta}/||\frac{dL_a}{\partial\theta}||^2$, $C = \frac{dJ}{\partial\theta}/||\frac{dJ}{\partial\theta}||^2$, and $z = J_c - b_m$. Using a Lagrange multiplier, the Lagrangian function is:

$$\mathbf{L}(\theta,\nu) = g^{\mathbb{T}}\Delta\theta + \lambda(\frac{1}{2}\Delta\theta^{\mathbb{T}} H\Delta\theta - \delta) + \nu(z + C^{\mathbb{T}}\Delta\theta)$$

λ, ν are dual variables. Using KKT Conditions [27] and simplifying the Lagrangian, we get the equations:

$$\begin{aligned} &\frac{\partial \mathbf{L}}{\partial\Delta\theta} = g + \Delta\theta + \nu C \\ &\lambda(\frac{1}{2}\Delta\theta^{\mathbb{T}} H\Delta\theta - \delta) = 0 \\ &\nu(z + C^{\mathbb{T}}\Delta\theta) = 0 \\ &\lambda, \nu \geq 0 \\ &(\frac{1}{2}\Delta\theta^{\mathbb{T}} H\Delta\theta - \delta) \leq 0 \\ &(z + C^{\mathbb{T}}\Delta\theta) \leq 0 \end{aligned} \tag{6}$$

solving which, an optimal update direction can be obtained. If there is a feasible solution, the optimal update direction is:

$$\Delta\theta^* = \frac{H^{-1}(g - C\nu^*)}{\lambda^*} \tag{7}$$

Algorithm 1 SAPO-RM

Require: $\Delta x_{em}, \Delta v_{em}, \Delta t, R_{safe}, \alpha$
Ensure: θ, ϕ
  repeat
    Sample trajectories $\mathscr{D} = \mathscr{T} \sim \pi(\theta_k)$
    Compute Constraints in the objective(Eq.4) with Eq.3
    Estimate Critic Parameters, ϕ from $\mathscr{D}$
    Calculate g,C,H in Eq.5 for the dual problem
    if feasible (Eq.9) **then**
      $\Delta\theta^* = \frac{H^{-1}(g+C\nu^*)}{\lambda^*}$
    else
      $\theta_{k+1} = \theta_k - \sqrt{\frac{2*\delta}{C^{\mathbb{T}}H^{-1}C}} H^{-1}C$
    end if
  until convergence

where ν^*, λ^* are optimal dual variables. For the retrieval update, we ignore the objective function and take the gradient descent with respect to the constraints to force the policy back to the safety region. Similar to [24], $\theta_{k+1} = \theta_k + -\sqrt{\frac{2*\delta}{C^{\mathbb{T}}H^{-1}C}} H^{-1}C$ is the retrieval policy through an iterative line search method.

The check for feasibility can be done by solving the optimization problem, proposed in [28]:

$$\begin{aligned} &\min_{\Delta\theta} \Delta\theta^T H\Delta\theta \\ &s.t.\ z + C^T\Delta\theta \leq 0 \end{aligned} \tag{8}$$

If the optimal value of the optimization problem in Eg.8 is δ_{min}, the feasible solution set is empty if $\delta_{min} \geq \delta$ and contains a solution otherwise. This problem is optimized efficiently using the Lagrangian Dual Problem:

$$\max_{\nu\geq 0} \frac{-\nu^T C^T H^{-1} C\nu}{2} + \nu^T z \tag{9}$$

From the RL pipeline in Fig.2, we use multiple rollouts sampled from various initial states, similar to [29], to capture the uncertainty in the environment and to generalize to a larger section of the state space. A pseudocode for our method can be found in Algorithm 1.

IV. Experiments

The proposed algorithm, Safety Assured Policy Optimization for Ramp Merging (SAPO-RM) is implemented on an online case in CARLA and compared with CPO, and a version we refer to as MCBF with a similar pipeline but with CBFs instead of Probabilistic CBFs. The experiments with CARLA highlight the advantage of our approach compared to the other versions and the importance of considering Probabilistic CBFs in a simulation framework. However, we also want to show the efficacy of the approach in a real-world merging scenario. We therefore use publicly available ramp-merging data from NHTSA, the NGSIM database on the US Route-101 and US Interstate-80 (I-80) to test our algorithm on real-world datasets. We report the results in terms of average relative minimum distance observed in the datasets with each method and provide visualizations to understand the merging behavior obtained by the algorithm.The reward function for the experiments (see Fig. 3) is as follows:

$$r(s,a,s') = -||s - g_s|| + 0.5 * \max d_{int},$$

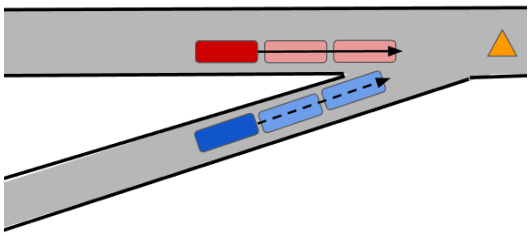

Fig. 3: **Reward Model:** The agent in red is the merging vehicle and the one in blue is an ego vehicle. The ego vehicle interpolates the trajectory of the merging vehicle for the next few time steps with the knowledge of the velocity and position. It also interpolates its trajectory and computes the minimum distance between the ego and the merging vehicle. The reward we use is the distance between ego and the goal (yellow triangle) and subtracting 0.5 times the minimum distances of the interpolated trajectories

where s is the current state, a is the action taken and s' is the next state of the system. g_s is the goal state co-ordinates of the final point, which leads to a terminal state and finishes the epoch. d_{int} is the set of interpolated distance between the ego and the merge vehicles for time steps into the future. A 4s lookout is used for interpolation with a delta of 0.1s.

A. *CARLA*

1) Setting up the Simulation Environment: We use CARLA ([1]) v. 9.12 to modify the available Town-06 environment to a ramp merging scenario. The scenario considered in Fig. 1 has a curved ramp. We use a PID Controller to track the steering control of the Ego Vehicle (On-Ramp) between waypoints generated by the default Global Planner to the specified goal position. The control for the Host (Off-Ramp) Vehicle uses both steering and longitudinal PID control. Further, the control of the host vehicle is not reactive. This assumption deals with a conservative case and satisfying this also maintains safety for the case where host control is reactive to the ego vehicle. We optimize the safe reinforcement learning objective (Eq.6) to learn a policy to control the ego-vehicle throttle position.

2) Online Experiments on CARLA: We implemented SAPO-RM in the Simulator set-up in 4.1.1. The RL pipeline introduced in Fig. 2 is followed, where the host vehicle is initialized at a specified initial position with a random offset. If the host vehicle reaches the destination before the ego vehicle, the host vehicle is re-initialized with random offset. The simulation is terminated only when the ego vehicle reaches the destination. The safety distance for the CBFs is 8m. Fig. 4 shows the results for SAPO-RM for a case where the target host velocity is 35 kmph.

*a) **Training Parameters and Environment Setup**:* The state dimension is 37, as we have to reproduce the entire state of an initial state distribution. We save the positions, velocities, angular velocities, throttle and steering positions in three dimensions for both the ego and merging vehicle. The action dimension is 1, as we are only controlling the throttle position. We use 100 trajectories with a rollout length of 20 per batch as a buffer. The actor policy has 4 MLP layers with a maximum size of 512, and the critic is two layers of MLPs of size 256. The parameter α is set to 0.75, although we change this parameters to alter the behavior. The uncertainty parameter η is 0.99 , and Δt is 0.01. We used a

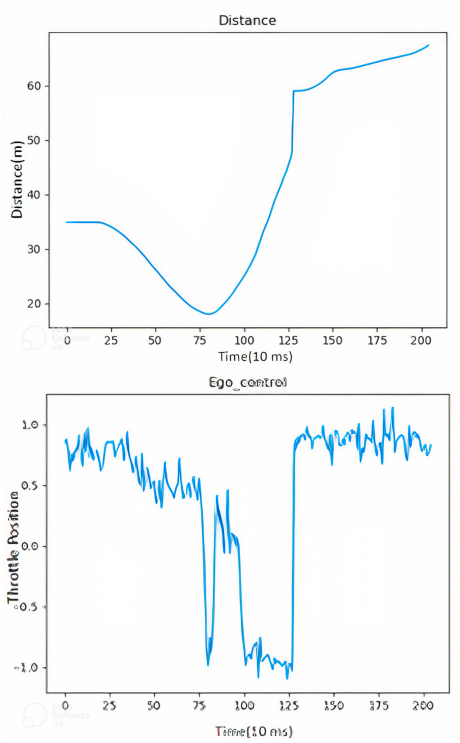

Fig. 4: **Results for Online SAPO-RM:** The host velocity is 35 kph, as seen in the center image. The image on the top depicts the distance between the ego and host vehicle. The image below is the output of the policy for the evaluated case. From the plots of the ego control we observe that when distance is close to R_{safe}, the throttle position goes to full-braking, forcing the ego vehicle to maintain the safety condition. When the distance increases again, the ego control goes towards full throttle to optimize performance.

desired velocity of 35 kmph for the ego-vehicle. The bounds on the throttle position are from [-1,1].

B. *Offline Experiments*

*1) **NGSIM Database Processing**:* An offline dataset for US I-80 highway ramp-merging was extracted from the NGSIM Database, which we use for training. We considered merging vehicles that go from on-ramp to an auxiliary ramp. To ease the complexity, the dataset was simplified into a one-on-one merging scenario, i.e, identifying the situations where the effect of the leading vehicles is negligible for the vehicle on the highway lane. The ego vehicles are first filtered based on the lane they start from and whether they merge eventually into the merging lane. Then, we consider the frame where the ego-vehicle merges into the highway lane as the merging frame and the position as the merge position. For all the ego vehicles where there are no merging vehicles within a pre-specified distance, we ignore those trajectories. For the trajectories that satisfy the condition: *distance between host and the ego vehicle is less than the specified distance in the auxiliary lane*, we further classify the merging vehicles depending on their leading vehicles. We want to observe the scenarios where the environmental effects on the merging vehicle are minimized akin to the dataset extraction in [30]. So, we consider merging vehicles without leading vehicles, which is available from the NGSIM Database. This condition still does not remove oscillations, so we address them by sorting the data of the ego vehicle with respect to the frame number

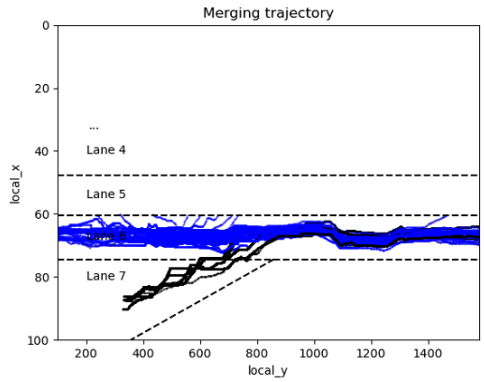

Fig. 5: **Merging Dataset:** It depicts all the trajectories extracted for I-80 dataset for one-on-one ramp merging.Trajectories in blue are that of the host vehicle and in black are of the ego vehicle.

and considering the first case where the merge happens and the ego-vehicle stays in the lane for at least three frames. A similar approach is adopted for the US Route-101 to extract a dataset that we use to test the trained algorithm.

a) US I-80 and Route 101 Dataset Extraction: Of the possible 11,850,526 data points in the NGSIM Dataset we have extracted a train, validation and test split of the merging dataset of about 100,000 data points (or ∼400 merging pairs) of the merging trajectories for a one-on-one situation. Figure 5. shows the trajectories of the merging condition we have applied to the US I-80 highway dataset. We filter the ego vehicles that start in lane 7. Of the possible combinations to go to lane 5 and lane 6, we only consider the case where vehicles merge to lane 6. For the host vehicles, we consider the vehicles that are within a specified distance of 50ft, when the ego vehicle is merging to lane 6. We also sort the data in terms of frames of ego vehicles for time-based trajectories.

*2) **Implementing Offline SAPO-RM**:* The offline dataset obtained in IV-B.1.a is then used to train the offline SAPO-RM policy. Similar to the online case, a rollout is initialized at a random state in the dataset. For each step in a rollout, only the parameters of the ego vehicle are updated according to the dynamics equation (Eq. 1). The network parameters of actor and critic, and the parameters of α, η and Δt are the same as that for the online case. We used a rollout length of 4 and 5000 trajectories for each epoch of training the offline RL policy for 100 epochs. We do not assume the information of an available trajectory future waypoints predicted by the planner. We use the knowledge of the merging point and fix the heading of the ego vehicle while its on the ramp to be the angle between initial ego position and merge point. Once the ego vehicle crosses the merge point, the heading angle is fixed to be the angle between the goal and merge point.

*3) **Experiments on the offline dataset**:* We ran our algorithm SAPO-RM, CPO and the MCBF variant of our algorithm on the offline datasets. We train the algorithms on the US I-80 dataset and test them on the I-80 test and Route-101 dataset. We also extract another test dataset where the minimum distance observed in the original dataset is less than the difference between the mean and the standard deviation of the minimum distances, representing the tail if the distances fit a normal distribution with the (-F) notation in Table 1. We keep track of the average relative minimum distance (ARMD) metric, the average of the relative difference of the minimum distance observed for a trajectory between the algorithm implemented and that in the dataset across all the trajectories.

TABLE I: ARMD Mean Values (ft) for algorithms on merging datasets

Algorithms	**I-80-test**	**I-80-test-F**	**US-101**	**US-101-F**
SAPO-RM	16.07	**26.9**	2.428	**1.879**
MCBF	16.58	26.7	**2.446**	1.728
CPO	**18.2**8	25.47	2.201	1.503

***Discussion**:* The results reported in the table indicate the improvement with SAPO-RM. The filter datasets represent the performance of the algorithms where the ego and merge vehicles come very close to each other in the original dataset and the relative metric being greater than zero for all three implementations indicate improvements on the dataset using our formulation. On the dataset that is remotely different, US-Route 101, there is $\sim 10\%$ improvement by using SAPO-RM over CPO. The CPO model performs well on the I-80 test dataset, a split of which it was trained on, but not on the I-101 dataset, which could be because of overfitting to the I-80 dataset. MCBF has almost similar performance to SAPO-RM, which shows the advantages of employing control barrier functions for constraints when compared to CPO. Further, Fig. 6 provides visualizations of the variations with the α value of CBFs on the merging behavior. The performance can be further improved when the merging point is also predicted by the network which leads to corner case initializations, which we are not considering in the current setup.

V. Conclusion and Future Work

We see from Fig. 5 and Fig. 6 that safety conditions are maintained by ego-vehicles while performing ramp-merging. The algorithm we used here can also be extended to other driving scenarios by changing the reward conditions and the dynamics simplifications used in Section 3. Inclusion of modeling uncertainty along with the adaptability of RL methods opens up further opportunities to improve safe RL for autonomous driving. Possible future extensions are handling situations with more merging vehicles and multi-agent interactions, introducing better approximations to dynamics models [17], [31], and alleviating modeling uncertainty. Further, prior work on ramp merging was majorly trained and evaluated on differently extracted datasets. For benchmarking these methods, creating standard datasets and a strong baseline are necessary to make a fair comparison.

References

[1] A. Dosovitskiy, G. Ros, F. Codevilla, A. M. L opez, and V. Koltun, "CARLA: an open urban driving simulator," CoRR, vol. abs/1711.03938, 2017.

[2] U. D. of Transportation, "Evaluation of adaptive cruise control interface requirements on the national advanced driving simulator," in NHTSA's Vehicle Safety Research, 2015

[3] N. AbuAli and H. Abou-zeid, "Driver behavior modeling: Developments and future directions," Int. J. Veh. Technol., vol. 2016, pp. 1–12, Dec. 2016

[4] Y. Lyu, W. Luo, and J. M. Dolan, "Probabilistic safety-assured adaptive merging control for autonomous vehicles," in 2021 IEEE International Conference on Robotics and Automation (ICRA), 2021, pp. 10 764–10 77

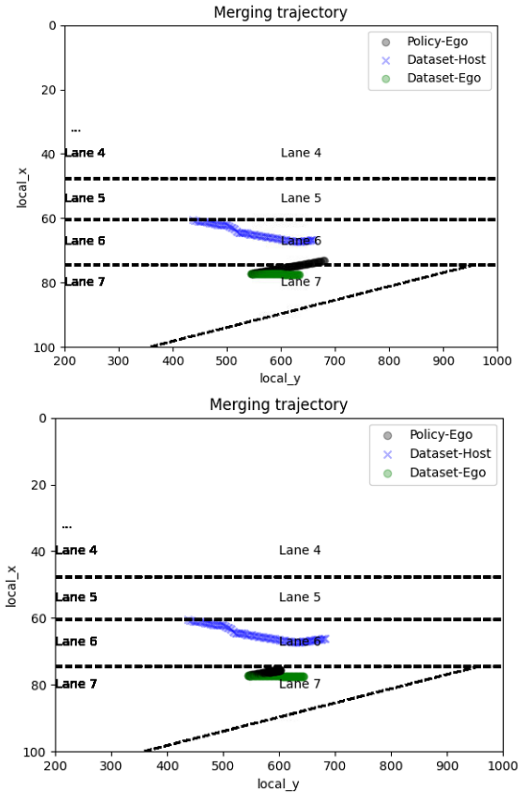

Fig. 6: The trajectory in black is generated by a policy, one in blue is the dataset host vehicle trajectory and the one in green is the dataset ego vehicle trajectory.The trajectory indicates merging with the knowledge of a merge point for the track. The figure shows the case with a higher value of alpha and a lower alpha(figure below) follows conservative behavior in SAPO-RM.

[5] Gu, T., Atwood, J., Dong, C., Dolan, J. M., and Lee, J.-W. Tunable and stable real-time trajectory planning for urban autonomous driving. In Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on, pp. 250–256. IEEE, 2015

[6] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in 53rd IEEE Conference on Decision and Control, 2014, pp. 6271–6278

[7] Luo, W., Chakraborty, N., and Sycara, K. Distributed dynamic priority assignment and motion planning for multiple mobile robots with kinodynamic constraints. In American Control Conference (ACC), 2016, pp. 148–154. IEEE, 2016

[8] Z. El abidine Kherroubi, S. Aknine, and R. Bacha, "Novel decision-making strategy for connected and autonomous vehicles in highway on-ramp merging," IEEE Transactions on Intelligent Transportation Systems, pp. 1–13, 2021

[9] Zhu, J. and Tasic, I. Safety analysis of freeway on-ramp merging with the presence of autonomous vehicles. Accid. Anal. Prev., 152(105966):105966, March 2021

[10] Z. Zheng, S. Ahn, D. Chen, and J. Laval, "Freeway traffic oscillations: Microscopic analysis of formations and propagations using wavelet transform," Procedia Soc. Behav. Sci., vol. 17, pp. 702–716, 2011

[11] H. Prakken, "On the problem of making autonomous vehicles conform to traffic law," Artif. Intell. Law, vol. 25, no. 3, pp. 341–363, Sept. 2017

[12] W. Luo, W. Sun, and A. Kapoor, "Multi-robot collision avoidance under uncertainty with probabilistic safety barrier certificates," Advances in Neural Information Processing Systems, vol. 33, 2020.

[13] Blackmore, L., Ono, M., and Williams, B. C. Chance-constrained optimal path planning with obstacles. IEEE Transactions on Robotics, 27(6):1080–1094, 2011

[14] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in 2019 18th European Control Conference (ECC), 2019, pp. 3420–3431

[15] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," IEEE Transactions on Automatic Control, vol. 62, no. 8, pp. 3861–3876, 2017

[16] G. Notomista, M. Wang, M. Schwager, and M. Egerstedt, "Enhancing game-theoretic autonomous car racing using control barrier functions," in IEEE International Conference on Robotics and Automation (ICRA), 2020, pp. 5393–5399

[17] L. Lindemann, A. Robey, L. Jiang, S. Tu, and N. Matni, "Learning robust output control barrier functions from safe expert demonstrations," ArXiv, vol. abs/2111.09971, 2021

[18] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," IEEE Transactions on Robotics, vol. 33, no. 3, pp. 661–674, 2017

[19] M. J. Khojasteh, V. Dhiman, M. Franceschetti, and N. Atanasov, "Probabilistic safety constraints for learned high relative degree system dynamics," in Learning for Dynamics and Control. PMLR, 2020, pp. 781–792

[20] J. Zeng, B. Zhang, and K. Sreenath, "Safety-critical model predictive control with discrete-time control barrier function," in 2021 American Control Conference (ACC), 2021, pp. 3882–3889.

[21] T. D. Son and Q. Nguyen, "Safety-critical control for non-affine nonlinear systems with application on autonomous vehicle," in IEEE 58th Conference on Decision and Control (CDC), 2019, pp. 7623–7628

[22] Choi, Jason, et al. "Reinforcement learning for safety-critical control under model uncertainty, using control lyapunov functions and control barrier functions." arXiv preprint arXiv:2004.07584 (2020)

[23] H. Ma, J. Chen, S. Eben, Z. Lin, Y. Guan, Y. Ren, and S. Zheng, "Model-based constrained reinforcement learning using generalized control barrier function," in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021, pp. 4552–4559.

[24] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in Proceedings of the 34th International Conference on Machine Learning, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 22–31. [Online]. Available: https://proceedings.mlr.press/v70/achiam17a.html

[25] H. Ma, X. Zhang, S. E. Li, Z. Lin, Y. Lyu, and S. Zheng, "Feasibility enhancement of constrained receding horizon control using generalized control barrier function," in 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS), 2021, pp. 551–557

[26] A. J. Taylor and A. D. Ames, "Adaptive safety with control barrier functions," in 2020 American Control Conference (ACC). IEEE, jul 2020

[27] Boyd, S. and Vandenberghe, L. Convex Optimization. Cambridge University Press, 2004. doi: 10.1017/CBO9780511804441

[28] Duan, J., Liu, Z., Li, S. E., Sun, Q., Jia, Z., and Cheng, B. Adapt ive dynamic programming for nonaffine non- linear optimal control problem with state constraints. Neurocomputing, 484:128–141, 2022. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2021.04.134. article/pii/S0925231221015848

[29] M. Janner, J. Fu, M. Zhang, and S. Levine, "When to Trust Your Model: Model-Based Policy Optimization," in Advances in Neural Information Processing Systems, 2019, vol. 32.

[30] C. Dong, J. M. Dolan and B. Litkouhi, "Intention estimation for ramp merging control in autonomous driving," 2017 IEEE Intelligent Vehicles Symposium (IV), 2017, pp. 1584-1589, doi: 10.1109/IVS.2017.7995935

[31] Q. Ge, Q. Sun, S. E. Li, S. Zheng, W. Wu and X. Chen, "Numerically Stable Dynamic Bicycle Model for Discrete-time Control," 2021 IEEE Intelligent Vehicles Symposium Workshops (IV Workshops), 2021, pp. 128-134, doi: 10.1109/IVWorkshops54471.2021.9669260.

[32] Van Koevering, Spencer and Lyu, Yiwei and Luo, Wenhao and Dolan, John, "Provable Probabilistic Safety and Feasibility-Assured Control for Autonomous Vehicles using Exponential Control Barrier Functions", 2022 IEEE Intelligent Vehicles Symposium (IV), pp 952–957, 2022,

[33] Lyu, Y., Luo, W. and Dolan, J.M., 2022, June. Adaptive safe merging control for heterogeneous autonomous vehicles using parametric control barrier functions. In 2022 IEEE Intelligent Vehicles Symposium (IV) (pp. 542-547). IEEE.

[34] Grover, J., Liu, C., Sycara, K. (2021). System Identification for Safe Controllers using Inverse Optimization. In IFAC-PapersOnLine (Vol. 54, Issue 20, pp. 346–353). Elsevier BV. https://doi.org/10.1016/j.ifacol.2021.11.198

[35] Grover, J. S., Liu, C., Sycara, K. (2020). Parameter Identification for Multirobot Systems Using Optimization Based Controllers (Extended Version). arXiv. https://doi.org/10.48550/ARXIV.2009.13817