

Decision Making in Autonomous Driving

Project Part I

Neha Desaraju

December 12, 2023

1 Paper Understanding

The paper [5] given demonstrates the state-of-the-art in robust control of autonomous driving systems. In the paper, the authors describe methods to implement robust optimization in a principled manner for non-linear systems and continuous states. Using a model-based approach, the authors are able to develop mechanisms for proper planning given a non-linear system (such as driving behavior) and continuous states (in this case, given by the position and velocity of each of the agents in the scene). To start with, the authors implement a simpler case to illustrate the concepts of their new methods; the authors discuss a method of discrete action space planning via sampling (which makes the set finite), which takes into account uncertainty while expanding the leaves in a Monte Carlo Search Tree. This will create a more definite lower bound while reasoning about the real probabilities of success of each path, rather than taking only the mean. This problem is a setup to the next problem that the authors formulate, which is when the state space is continuous. This calculates the closest interval bounds for the possible next states (trajectory ends). Once this is implemented, we trivially have a definitely lower bound on performance. The paper runs experiments on the environment we will use, described in [4].

2 Proposed Experimental Methods

2.1 Robustness

To improve robustness in the model, I propose two methods. The first is using an ensemble of independently trained models [14] to learn slightly different dynamics functions. As a model-based method, this method will utilize the cross entropy method (assuming a continuous action space) and a Monte Carlo Tree Search method as the planning method (assuming a discrete action space), which would be similar to the sample-based planning outlined in part II of the paper. These planning methods will ensure simplicity, and I can use the uncer-

	Learning method	Action Space	Planning/policy method
Model ensemble	Model-based	Finite	MCTS
	Model-based	Continuous	CEM
Gaussian Process Model	Model-based	Continuous	CEM
	Model-free	Continuous	Policy gradient

Table 1: Table of proposed experiments

tainty developed in the ensemble model to balance exploration of various paths as well as develop a stronger lower bound for performance.

Another set of experiments I would like to explore involve using a stochastic Gaussian Process model, as explored in [7], both in a model-based setting (to approximate the dynamics of the environment) as well as in a model-free setting (to learn the distribution of rewards for a particular series of actions). In the previous case, I will once again use cross entropy method, and in the second case I will learn a policy function through policy gradient.

A table of these experiments is given in Table 1. I plan to start with the implementation of the GP model (on top of a simple dynamics network) training by sampling episodes from the environment simulation. This model will learn a using the continuous action space and utilizing CEM to plan trajectories.

As the environment [4] we are using does not natively output noise or variance in the agents’ movements, I will manually add perturbations in order to demonstrate my methods of evaluating and considering model uncertainty. It is shown by [11] that adding perturbations in the observation data can help the model to generalize to unseen situations better. Therefore, I will use a simple Gaussian distribution to slightly change the observed dynamics (given by x , vx , y , and vy). Some examples to do so are demonstrated in this [10] Github link. In addition, other forms of simple noise are explored in this paper [3], and it may be possible to implement a signal-to-noise ratio function [13] to better set the correct standard deviation given some amount of noise we would like to include. Note that we are only adding noise to the *observations*, not the true behavior of the agents in the scene. This distinction is important because in the case for robustness, we want to simulate sensor uncertainty, not necessarily the randomness of true driver behavior (see next section).

2.2 Safety

Since driver behavior can be somewhat random, not optimal, and not perfectly modeled by our dynamics, we want to limit the lower bound of guaranteed outcomes (rewards). That is, we want to avoid collisions at all costs, which includes encouraging behavior such as maintaining a large follow distance behind other cars and turning only when ”safe”; that is, allowing room for other driver error.

We may want to simulate this driver error by adding some random driver

behavior in the environment itself, as described in [9], if we find that the environment is not suitable to ensure our model is constrained enough.

Many approaches to incorporating safety standards in controlling a car involves checking the safety constraints while planning (in the case of model-based learning), or only allowing safe (or likely safe) actions (in the case of model-free learning). [8] develops a safety specification as a function $l(x(t))$, where $x(t)$ is a state, and $l(x(t))$ returns a measure of safety risk or severity of risk. This work calculates cumulative risk while planning trajectories, and takes into account the resulting scalar to better inform the selection of trajectories (using traditional planning techniques).

In my first experiment, I plan to only select trajectories that meet certain hard safety constraints, then slowly relax this constraint via a function — for example, I will select trajectories that fall under a certain level of risk (within uncertainty) while also allowing for exploration. To keep it simple to start off with, this risk will only be determined by some inverse square of absolute distance to the nearest agent.

Several other methods are proposed by [12], [2], [1], and [6].

References

- [1] Felix Berkenkamp, Matteo Turchetta, Angela P. Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees, 2017.
- [2] Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces, 2018.
- [3] Raad Khraishi and Ramin Okhrati. Simple noisy environment augmentation for reinforcement learning, 2023.
- [4] Edouard Leurent. An environment for autonomous driving decision-making. <https://github.com/eleurent/highway-env>, 2018.
- [5] Edouard Leurent, Yann Blanco, Denis Efimov, and Odalric-Ambrym Mailard. Approximate robust control of uncertain dynamical systems, 2019.
- [6] Sam Mottahedi and Gregory S. Pavlak. Constrained differentiable cross-entropy method for safe model-based reinforcement learning. In *Proceedings of the 9th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, BuildSys ’22, page 40–48, New York, NY, USA, 2022. Association for Computing Machinery.
- [7] Zachary Nado, Neil Band, Mark Collier, Josip Djolonga, Michael W. Dusenberry, Sebastian Farquhar, Qixuan Feng, Angelos Filos, Marton Havasi, Rodolphe Jenatton, Ghassen Jerfel, Jeremiah Liu, Zelda Mariet, Jeremy Nixon, Shreyas Padhy, Jie Ren, Tim G. J. Rudner, Faris Sbahi, Yeming

- Wen, Florian Wenzel, Kevin Murphy, D. Sculley, Balaji Lakshminarayanan, Jasper Snoek, Yarin Gal, and Dustin Tran. Unvertainty baselines: Benchmarks for uncertainty & robustness in deep learning, 2022.
- [8] T. Nyberg, C. Pek, L. Dal Col, C. Norén, and J. Tumova. Risk-aware motion planning for autonomous vehicles with safety specifications. In *2021 32nd IEEE Intelligent Vehicles Symposium (IV)*. IEEE, May 2021.
 - [9] Ashish Rana and Avleen Malhi. Building safer autonomous agents by leveraging risky driving behavior knowledge. In *2021 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI)*. IEEE, October 2021.
 - [10] Aaditya Ravindran. Methods of introducing noise to the cartpole environment in openai gym. <https://github.com/AadityaRavindran/gym-cartpolemod>, 2017.
 - [11] Alexander Robey, Hamed Hassani, and George J. Pappas. Model-based robust deep learning: Generalizing to natural, out-of-distribution data, 2020.
 - [12] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving, 2016.
 - [13] Wikipedia. Signal-to-noise ratio — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Signal-to-noise%20ratio&oldid=1184390748>, 2023. [Online; accessed 28-November-2023].
 - [14] Jingda Wu, Zhiyu Huang, and Chen Lv. Uncertainty-aware model-based reinforcement learning with application to autonomous driving, 2021.