Deep AutoRally: Agile Autonomous Driving via End-to-End Imitation Learning

Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keunteak Lee,
Xinyan Yan, Evangelos A. Theodorou and Byron Boots

Institute for Robotics & Intelligent Machines, Georgia Institute of Technology, Atlanta, GA, USA
{ypan37,cacheng,kamilsaigol,keuntaek.lee,voidpointer,evangelos.theodorou}@gatech.edu, bboots@cc.gatech.edu

I. INTRODUCTION

High-speed autonomous driving on rough terrain is a challenging robotics problem [11, 5] (Figure 1). In this task, a robot is required to perform precise steering and throttle maneuvers in a physically-complex, uncertain environment by making high-frequency decisions. Traditional engineering approaches to autonomous driving, which decouple the agent into independent perception [5], planning and control [11] modules, has enjoyed great success when the robot’s interaction with the environment can be precisely characterized. However, as robots move into unstructured natural environments and operate at higher speeds, it is becoming more difficult to model these interactions a priori.

One possible solution to this problem is to add costly sensors and focus on complicated system engineering—which consumes large amounts of time and money—for robust but conservative solutions. For example, a similar task has been considered by Williams et al. [11] using model-based Reinforcement Learning (RL). While the authors demonstrate impressive results, their internal control scheme assumes full observability and relies on an accurate state estimator that has access to exteroceptive sensors (e.g. GPS) and a dynamics model of the car. This requires extensive calibration and the robot to operate in a controlled environment, which limits the applicability of their approach.

In this paper, we focus on an alternative framework for designing intelligent robots: policies that govern a robot’s behavior can be learned from the robot’s interaction with its environment rather than hand-crafted by an engineer. We aim to learn an agile driving policy that uses only on-board measurements (e.g. images, wheel speeds) to control continuous-valued actions. With these sensory limitations, it becomes unclear how to accurately describe the dynamics as required in the traditional model-based approach. Building on the success of deep RL [4, 6], we adopt deep neural networks to parametrize the control policy, essentially jointly optimizing the perception and the control systems. While the usage of deep neural network as a policy representation is not uncommon, in contrast to most previous works that showcase RL in simulated environments [6], our agent is a high-speed physical system that incurs real-world cost: a single poor decision can physically impair the robot. Therefore, direct application of model-free RL techniques is not only sample inefficient, but also potentially costly and dangerous in our

These real-world factors motivate us to adopt imitation learning [8] to optimize the control policy instead. Self-driving cars [1, 9] have recently started to employ an end-to-end imitation learning approach: based on deep neural network policies, these systems require only expert demonstrations during the training phase and on-board measurements during the testing phase. For example, Nvidia’s PilotNet [1], a convolutional neural network that outputs steering angle given an image, is trained to copy the human driver’s reaction and demonstrates impressive performance in real-world road tests.

Here we show the idea of imitation learning can be extended to high-speed off-road driving tasks. Our problem and setup, however, differs from these on-road driving tasks considered previously. Prominent visual features, such as road lines, are absent, and the surfaces that the robot navigates is constantly evolving and highly stochastic. In addition, high-speed driving on rough terrains requires both steering and throttle commands to be applied at a high frequency, whereas previous works [7, 11] [9] only concern steering commands.

To tackle with these difficulties, we study the properties of batch and online imitation learning algorithms in theory and experiments. Empirically, we find that imitation learning in general is more data-efficient than learning a new dynamics model for model-based RL, such as model predictive control (MPC) [11]. Furthermore, training the control policy with online learning and DAgger [10], along with an MPC expert, improves the robot’s performance in tasks with clear objectives; batch learning is preferred for complex tasks where the expert is a human and a cost function is difficult to parametrically define (e.g. obstacle avoidance using raw images). Leveraging imitation learning, our AutoRally car with deep neural network policy can learn to perform high-speed navigation at a state-of-the-art average speed of ~6 m/s, and obstacle avoidance at 4-5 m/s.
II. OUR APPROACH

We formulate the learning of control policy as a discrete-time continuous-valued RL problem. In our setting, the state space is unknown to the agent; observations consist of onboard measurements, including a monocular RGB image from the front-view camera, wheel speed, and inertial measurement unit (IMU) readings; actions include continuous-valued steering and throttle commands. Let $A$ and $O$ be the the action space and observation space. The goal is to find a stationary deterministic policy $\pi : O \mapsto A$ such that $\pi$ achieves low accumulated cost over a horizon of $T$.

A. Imitation Learning

Directly solving a RL problem is challenging for high-speed off-road autonomous driving. On one hand, since our task involves a physical robot, model-free RL techniques are intolerably sample inefficient and have the risk of permanently damaging the car when applying a partially optimized policy in exploration. On the other hand, although model-based RL requires fewer samples, it can lead to suboptimal, potentially unstable, results when the model fails to fully capture the complex dynamics of dirt track driving.

Considering these limitations, we propose to solve for the control policy by imitation learning. We assume the access to an oracle or expert $\pi^*$ to generate demonstrations during the training phase, which relies on resources that are unavailable in the testing phase, e.g., additional sensors, model knowledge, and computations. Such an expert can be a computationally intensive optimal controller that relies on exoteric sensors not available at test time (e.g. GPS for state estimation), or an human teleoperating driver.

The goal of imitation learning is to perform as well as the expert with an error that has at most linear dependency on the task time horizon $T$. In order to tackle the limitations of batch learning, e.g., the compounding error that grows quadratically with task horizon $T$, we train the neural network policy $\pi$ iteratively using a meta-learning algorithm, DAGger [10], in which at each iteration a supervised learning subproblem is solved. While online learning seems appealing theoretically, batch learning has been empirically shown to outperform online learning in certain tasks [3], especially when combined with expressive function approximators like deep neural networks.

B. End-to-End Neural Network Policy Learning

We parameterize the policy $\pi$ by a deep neural network, called the Deep AutoRally Network (DARN). DARN consists of three sub-networks: a convolutional neural network (CNN) that takes RGB images as inputs, and two feedforward networks with fully-connected layers, that take wheel speeds and IMU readings as inputs. To learn the policy, we consider $A$, equipped with $\| \cdot \|_1$, and solve for the policy using ADAM [2], which is a stochastic gradient descent algorithm with adaptive learning rate. Note that the neural network policy does not use the state, but rather the synchronized raw observation as input.

III. EXPERIMENTS

We considered two tasks: 1) high-speed navigation along the track, and 2) high-speed obstacle avoidance using monocular images. In the high-speed driving task, we used both an MPC expert and a human driver. In the obstacle avoidance task, we only used the human driver because a cost function that takes images as input is hard to specify. We implemented our method on a 1/5 scale autonomous AutoRally car (Figure 1) and this platform was used to carry out both simulated (Gazebo-based) and real-world experiments. Simulation results on high-speed navigation tasks show that our approach is more data efficient than model-based RL in which a dynamics model of the vehicle needs to be learned from data. The real track tests show that DARN is able to perform fast off-road navigation autonomously at an average speed of 6 m/s, and obstacle avoidance (Figure 3) at 4-5 m/s.

REFERENCES