Reinforced Curriculum Learning for Autonomous Driving

Luca Anzalone [1] Silvio Barra [2] Michele Nappi [3]

[1]University of Bologna (DIFA) [2]University of Naples "Federico II" (DIETI) [3]University of Salerno (DI)

Abstract

We combined **Reinforcement Learning** (RL) [5] with **Curriculum Learning** [1] to learn an *end-to-end* driving policy for the **CARLA** autonomous driving simulator [3]:

- **Five stages** of curriculum learning guide training.
- **Proximal Policy Optimization** (PPO) [4] improves the agent driving policy $\pi_\theta(a \mid s)$, at each stage.
- The **value function** $V(s)$ is decomposed into *bases* b and *exponents* e, such that: $V(s) = b \cdot 10^e$.
- The **advantage function** is normalized to preserve its sign.
- For the first time, achieved **results are consistent on all towns**.

Introduction

Autonomous Vehicles (AVs) can be built in two ways [6]:

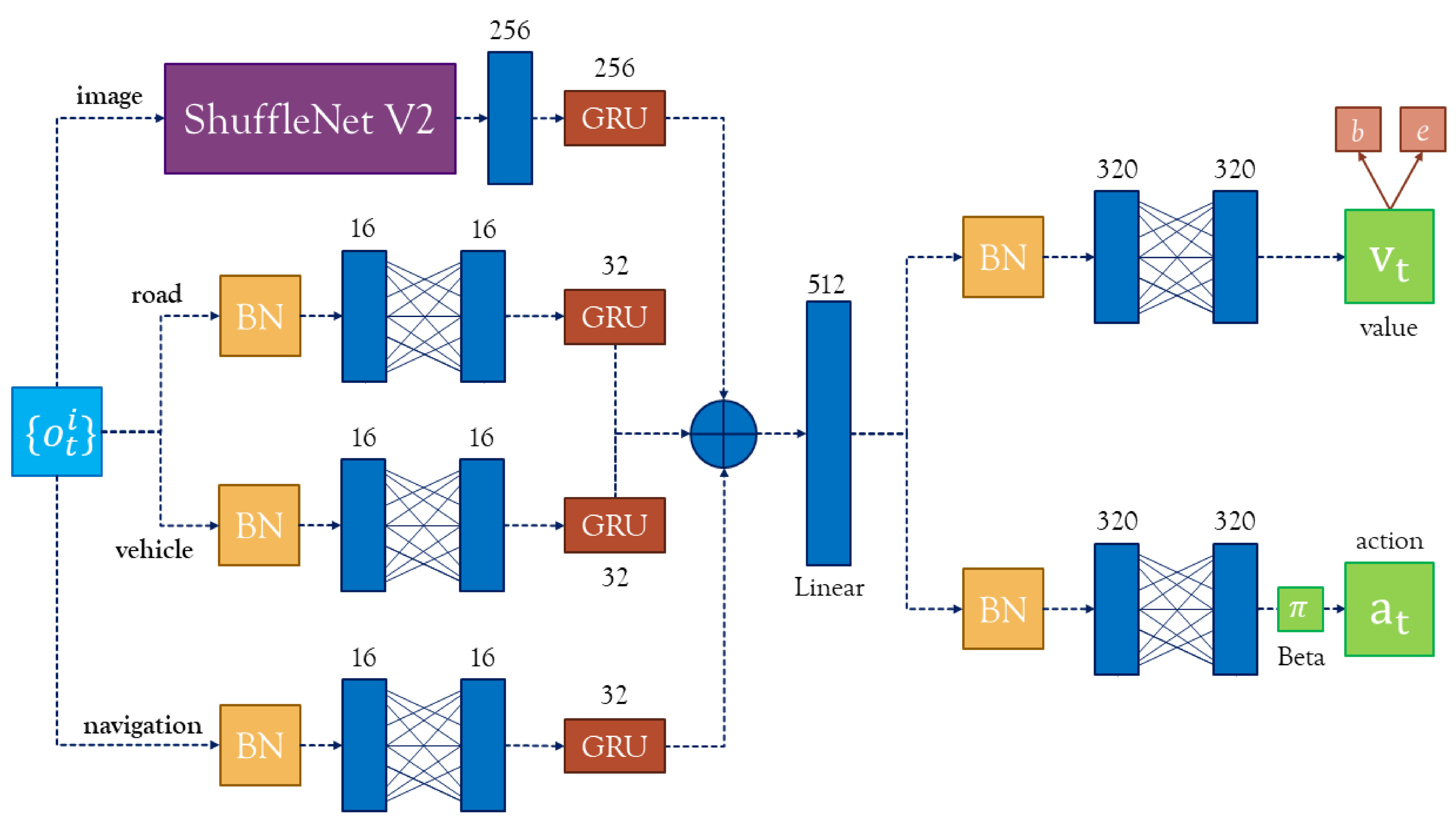
- **Modular pipeline:** *percetion*, *planning*, and *motion control* components are build and optimized in *isolation*, towards human-designed criteria. **Error propagation** is a major issue. Intermediate representations are not optimal.
- **End-to-end approach:** underlying tasks are *implicitly learned*, without domain knowledge, and *jointly optimized* with respect to a *global objective*.

In practice, end-to-end AVs are implemented leveraging:

- **Imitation Learning [2]:** supervision by large amounts of labeled expert data, training is easy and stable, out-of-distribution data is a major issue.
- **Reinforcement Learning:** requires to interact with a driving environment, often unstable, can discover better-than-expert driving policies.

Agent Architecture

We designed and end-to-end RL-based autonomous system:



The agent neural network processes a stack of four sub-observations o_t^i at each timestep t, whose outputs are aggregated by multiple Gated Recurrent Units.

Base-Exponent Value Decomposition

$V_\phi(s_t)$ is learned by minimizing the squared loss towards the returns R_t. Since returns can be very **large numbers**, the regression can become **very unstable**.

How to avoid the "large" part of a number?

$$V = b \cdot 10^e$$

Where the *base* $b \in [-1, 1]$, and the *exponent* $e \in [0, k]$. The **hyperparameter** k is set such that the maximum/minimum value or return does not exceed $\pm 10^k$.

What we regress are the bases b_v and exponents e_v of the values:

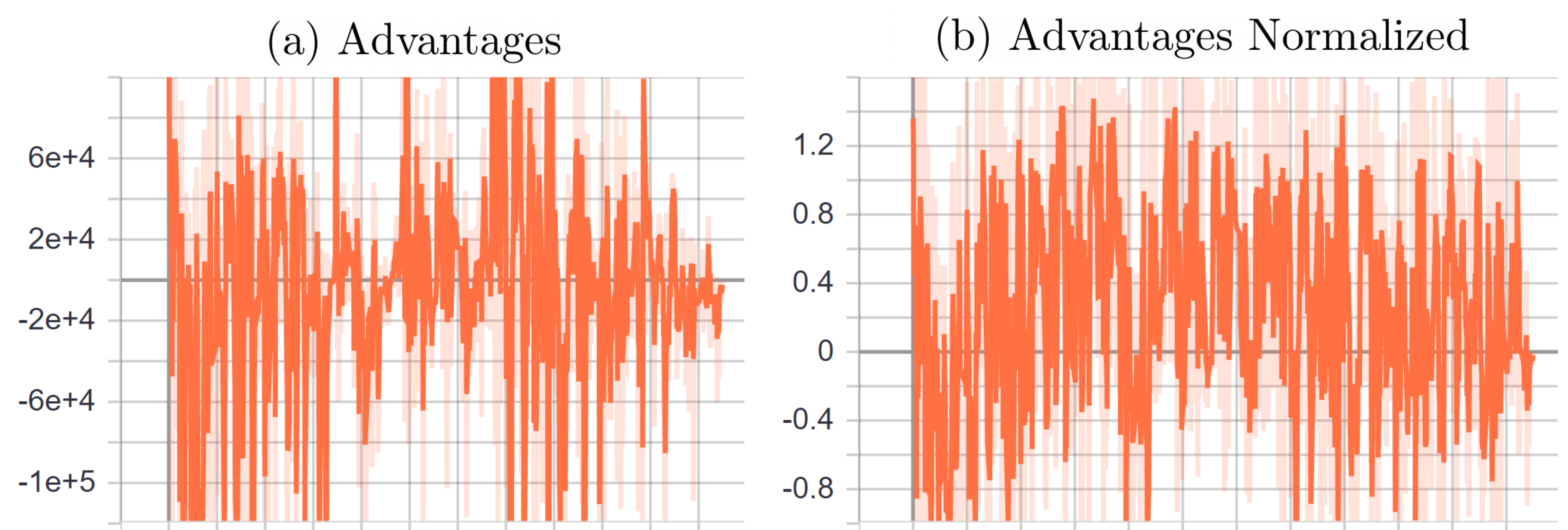
$$\mathcal{L}_V(\phi) \propto \|b_v - b_R\|_2^2 + \|e_v - e_R\|_2^2$$

Benefits:

- The gradient norm is *always* small,
- Unbiased normalization of returns R_t,
- Easy to implement and optimize.

Sign-preserving Advantage Normalization

The magnitude of the estimated advantages A_t affect the norm of the policy gradient in a multiplicative way. Higher norm means larger update steps, which can make the training process less stable.



By independently normalizing the negative and positive elements of the vector A_t, we reduce them to **unit scale** while keeping their **sign unchanged**: meaning that the intuitive notion of *better-* or *worse-than-average* actions a, described by the advantage function $A(s, a)$, is still preserved.

Benefits:

- Unitary scale of the estimated advantages,
- Small norm of the policy gradients,
- Better interpretation thanks to sign preservation.

Reinforced Curriculum Learning

A curriculum composed of **five stages** of reinforcement learning guides the agent:

- **Stage 1:** the agent starting point is sampled from a small set of 10 locations. There are no pedestrians and vehicles. Speed limits must be respected.
- **Stage 2:** the set of starting locations is enlarged to 50. Also, at most 50 pedestrians are placed around the map.
- **Stage 3:** starting locations are not limited in number anymore. Change of weather and light conditions are introduced. Moreover, 50 vehicles are also positioned across the entire town.
- **Stage 4:** data augmentations on captured camera images are enabled.
- **Stage 5:** along with all the previously defined rules, the number of pedestrians and vehicles is respectively increased to 200 and 100.

Each stage can be regarded as a distinct learning environment. Furthermore, notice that all the training stages occur in the same urban scenario: `Town03`.

Results

Stage-based reinforcement learning has proved to be *robust* and *consistent* on visually and topologically different towns from the CARLA simulator:

Metric	Town01	Town02	Town03	Town04	Town05	Town06	Town07	Total**
Collision rate (%)	86	**78**	88	**51**	49	33	**77**	64
	79	84	**70**	63	**40**	**30**	78	**63**
Speed (km/h)	7.78	**8.46**	8.13	9.05	8.55	**9.63**	7.65	8.5
	8.58	8.22	**8.30**	9.05	**9.36**	9.33	**7.68**	**8.65**
Total reward	1866	**2530**	**2157**	**2161**	1764	1951	**1813**	**2025**
	2135	2036	1996	1780	**2190**	**2030**	1479	1944
Waypoint distance (m)	**1.54**	**1.44**	**1.75**	**3.75**	**3.74**	5.16	**2.18**	**2.98**
	1.77	1.97	2.98	3.90	3.80	**4.69**	2.24	3.08

Table 1. First row of each metric refers to the curriculum-based agent. (**) Total results are aggregated over weather sets and traffic scenarios; for clarity some columns are missing.

References

[1] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[2] F. Codevilla, M. Müller, A. López, V. Koltun, and A. Dosovitskiy. End-to-end driving via conditional imitation learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4693–4700, 2018.

[3] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. *arXiv preprint arXiv:1711.03938*, 2017.

[4] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017.

[5] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[6] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8:58443–58469, 2020.