Neural Network Reuse in Deep RL for Autonomous Vehicles among Human Drivers

Manon Legrand*, Roxana Rădulescu, Diederik M. Roijers, and Ann Nowé

Vrije Universiteit Brussel, Artificial Intelligence Lab
manon.legrand@live.be,{rradules,droijers,ann.nowe}@ai.vub.ac.be

In this thesis, we consider the problem of reinforcement learning for autonomous cars on a highway. This is inherently a multi-agent problem, in which agents have to learn to handle the dynamics of a system, other autonomous cars, and be able to account for some irrationality on the part of other drivers (typically human). We create a simple traffic simulator composed of straight lanes and simulated human drivers (following a rule-based behavioral model with some irrationality built into it), and design different neural networks as learning models for our self-driving cars. Using this simulator, we try to answer the question on how we can efficiently train agents in such a complex multi-agent setting. Our key insight is that we can reuse neural networks trained on a single-agent version of a multi-agent problem to speed up learning, leading to good performance.

1 Learning

We first create a traffic simulator. Our simulator is a discrete implementation of a highway of straight lanes – where a lane is a list of cells – and human drivers who act according to a pre-defined behavior. Autonomous cars – cars whose actions are determined by a learning model – can then be added in this environment.

We then design 5 neural network models that use different information about the environment or use similar information differently. In short, we have 3 feed-forward network models that use, respectively, the presence of the cars[1] for the current time step, the presence for the current and the previous time steps, and the presence of cars and their speeds for the current time step. Moreover, we create 2 convolutional neural networks: the first uses the presence and the speed of cars as 2-D matrices, and the second uses the presence of cars for the current and previous time steps as a 3-D matrix.

First, all models are trained with two hidden layers with a varying number of neurons in a single-agent setting. The training is done with deep Q-learning [2]: that is, the neural network serves as a function approximator for the Q-values and is trained with experience replay.

Afterwards, the same models are trained in a multi-agent setting. The first set of experiments consists in exactly the same setting as before but with more

* This work was carried out for the first author's master thesis studies, at VUB [1]. The other authors were the supervisors. Full work available at: http://ai.vub.ac.be/sites/default/files/thesis_legrand.pdf

[1] And other information about the learning agent itself that is used by all models

than one agent at the same time on the highway. We then repeat the same experiments but with a modified version of the models – models that include in the inputs whether the observed cars are autonomous. Finally, we take advantage of the fact that we have models already trained, in a single-agent environment, and use these trained models as the starting point for the multi-agent learning. This has a lot of advantages, as the trained network already contains most of the information needed to learn to behave optimally in the multi-agent setting. We show that such reuse is highly beneficial in the next section.

2 Results

In our experiments we distinguish between different outcomes: *goal* when the car reaches the end of the highway without crashing and *crash* otherwise. We show that the single-agent learning problem can be easily solved as most of the models/configurations achieved a good performance. However, further tests show that these trained models do not necessarily adapt well when the simulator's parameters change (e.g. traffic density). More importantly, the single-agent trained models show poor performance in a multi-agent setting (Figure 1 left), i.e., while the autonomous car agents can learn to deal with the human behavior, they cannot respond correctly to other agents. Therefore, it is necessary to explicitly train in a multi-agent setting. Training from scratch in the multi-agent setting takes a very long time, and does not learn to perform adequately yet in the time we had available. However, when the multi-agent learning uses a model that was already trained in a single-agent setting as a starting point, the results are significantly better (Figure 1 right).

We therefore conclude that reusing neural networks trained on a single-agent version of a multi-agent problem can lead to significant speed-ups and good performance.

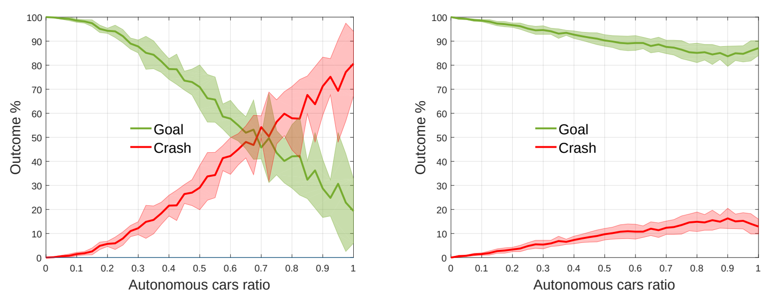

Fig. 1: Agent performance on multi-agent setting as a function of autonomous car ratio, trained on single-agent setting (left), and then re-trained on multi-agent setting (right)

References

1. Manon Legrand. Deep Reinforcement Learning for Autonomous Vehicle Control among Human Drivers. Master dissertation, Vrije Universiteit Brussel, 2017.
2. Volodymyr Mnih et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.