Sim2Real Transfer for Traffic Signal Control

Longchao Da, Hao Mei, Romir Sharma and Hua Wei†

***Abstract*— Traffic signal control is a complex and important task that affects the daily lives of millions of people. Reinforcement Learning (RL) has shown promising results in optimizing traffic signal control, but transferring learned policies from simulation to the real world remains a challenge due to the domain gap between the simulation and the complex real-life scenario. In this paper, we utilize grounded action transformation to mitigate the domain shifting problem and improve Sim2Real transfer for RL-based traffic signal control. Grounded action transformation leverages the dynamics between the simulation and real-world actions to generate effective real-world actions. We evaluate our method on a simulated traffic environment and show that it significantly improves the performance of the transferred RL policy in the real world. Our results demonstrate the potential of grounded action transformation as a promising technique for Sim2Real transfer in RL-based traffic signal control.**

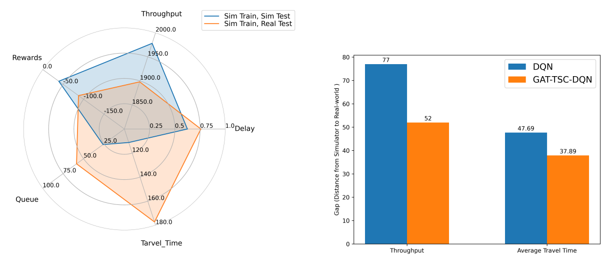
I. INTRODUCTION

With the growing availability of traffic data and advancements in deep reinforcement learning techniques, there is a developing trend toward utilizing reinforcement learning (RL) for traffic signal control (TSC). However, current research on reinforcement learning-based TSC is limited to simulators. Although simulation-based training is cost-effective, it suffers from inherent discrepancies with real-world settings due to the complex nature of real-life dynamics. As a result, a critical challenge in implementing real-life RL is finding ways to apply simulation-based training in real-world scenarios.

In order to bridge the gap, much effort has been made. Some researchers adopt domain randomization ideas, which intend to cover the actual distribution of the real-world data by randomizing the simulation multiple times. On the other hand, through feature representation and transferring, domain adaptation methods exploit the source domain data to improve the model's target domain performance, whose data is practically scarce [1]. However, the above exploration is mainly applied to the robotics domain, and few studies have been conducted on the TSC area, even though it is also suffering such a plight. We conducted a preliminary study on a specific trained policy to demonstrate the gap it may have when performing in the simulator and real-world settings, as shown in Fig 1(a). Note that the reward here is calculated as the total number of waiting vehicles. Solving the sim2real problem in TSC is an inevitable way to the policy's practical deployment.

†Corresponding author: Hua Wei with Longchao Da, Hao Mei, are at the Department of Informatics, New Jersey Institute of Technology, Newark, New Jersey, USA {hua.wei, ld49, hm467}@njit.edu

Author: Romir Sharma is with the West Windsor-Plainsboro High School South, West Windsor, New Jersey, USA sharmaromir@gmail.com



(a) Gap of sim2real (b) Method's improvement

Fig. 1. Preliminary study and our method's Improvement

In this paper, we propose to use grounded action transformation (GAT-TSC) to bridge such a gap in the training process, helping to calibrate the dynamics distribution shifting. Furthermore, we quantify the model's parameter uncertainty and leverage it to dynamically adjust the action grounding rate, contributing to the training efficiency and stability. The improvement using GAT-TSC-DQN in an example is shown in Fig 1(b). Please note that only two metrics, Throughput and Average Travel Time are presented due to the page limit.

II. METHODS

To shrink the gap in the transition dynamics between the simulator and the real world, we propose an uncertainty-controlled grounded action transformation framework inspired by Grounded Simulation Learning (GSL) [2]. With the help of the grounded action transformation step, the difference in the effects of action between the simulator and the real-world environment could be minimized. And the uncertainty is used to leverage the potential risk of extending differences by taking such a grounded action due to the underfitting of models and simultaneously keeping learning policy stable and finally converging.

a) Grounded action transformation: Grounded action transformation (GAT) [6] is a special case under the framework of the GSL algorithms to mitigate the difference between low-fidelity simulator environments and high-fidelity ones. Given a real world dataset $\mathcal{D} = \{\tau^1, \tau^2, ..., \tau^I\}$, where $\tau^i = (s_0^i, a_0^i, s_1^i, a_1^i, ..., s_L^i, a_L^i)$. The *grounding* refers to the process of finding ϕ^* that minimizes a surrogate objective:

$$\phi^* = \arg\min \sum_{\tau^i \in \mathcal{D}} \sum_{t=0}^{L} d(P(s_{t+1}^i | s_t^i, a_t^i), P_\phi(s_{t+1}^i | s_t^i, a_t^i)) \tag{1}$$

where $d(\cdot)$ is the distance between two dynamics, P is the real-world transition dynamics and the P_ϕ is the simulation transition dynamics, s_t^i and $s_{t+1}^i \in \mathcal{S}$ are current state and next time state in state space, $a_t^i \in \mathcal{A}$ is the action taken at current time step belong to action space. The grounding procedure is achieved by training an action transformation function with supervised learning:

$$g(s_t^i, a_t^i) = f_{sim}^{-1}(s_t^i, f_{real}(s_t^i, a_t^i)) \tag{2}$$

The forward model is trained with the gradient from $MSE(f_{real}(s_t^i, a_t^i), s_{t+1}^i)$ by data collected from real-world. The inverse model is trained with the gradient from $CrossEnropy(f_{sim}^{-1}(s_{t+1}^i, s_t^i), a_t^i)$ with data collected from low-fidelity simulator environment. In the policy improvement step, we also introduce a hyperparameter α as a dynamic grounding rate, to stabilize policy improvement and make the training process finally converge.

b) Dynamic grounding rate by uncertainty: During the action grounding phase, due to the imperfection of the forward and inverse models, the grounded action is prone to enlarge instead of shrinking the difference of states between the simulation and real-life environment. A more severe situation is the grounding action with low belief will make this transformation behave like random exploration, resulting in instability during training. To quantify belief masses and uncertainty, we introduce a dynamically adjusted hyperparameter α to dynamically determine taking grounded action with a qualified uncertainty value during inference. In the framework, the uncertainty of u_t^r at time t after training round r is quantified based on the Evidential Deep Learning method [7]. If $u_t^r < u^r$, where u^r is the average uncertainty of all u during the training round r, the model conducts action grounding. Based on $\alpha = u_r$, if the uncertainty at round t: $u_t < u_r$, the model conducts action grounding.

III. Experiment and Results

a) Experimental Design: We treat SUMO [3] as the real world and Cityflow [4] as the simulation. Specifically, we modify SUMO's environmental settings as the change of the real world, and we train the TSC model by DQN [5] in Cityflow. As shown in Table I, we modified the following parameters in SUMO: acceleration, deceleration, emergency deceleration, startup delay, and average container capacity for vehicles, which provides three settings for the real world (default, V1 and V2). The default setting in SUMO has the same parameters as Cityflow. Our goal is to minimize the test performance in the real world of our trained model both in average travel time and the overall throughput.

TABLE I

Experimental Settings on Real-world config

Setting	accel	decel	eDecel	sDelay	cCapacity
Default	2.6	4.5	9.0	0	0
V1	1	2.5	6	0.5	3
V2	1	2.5	6	0.75	1

b) Experimental results: Based on the listed real-world settings above, we conduct experimental analysis: First, we train an RL model in the Cityflow simulator by DQN algorithm until it steadily converges after 200 Epochs, the metrics average travel time ATT (in seconds) and throughput TP (in number of vehicles) as reported under the column Simulator, Second, we directly transfer the trained RL-models in two settings in SUMO following the parameters in Table I. From the column Real, we can see the gap between the simulator and the real-world environment exists. We then apply our method, GAT-TSC, which trains the GAT model on the top of the DQN, to test in SUMO and report the results in the GAT-Real column as shown in Table II, our method can shrink the gap of the testing performance between the real-world environment (SUMO) and the simulation (Cityflow).

TABLE II

The results of models trained in Cityflow but tested in the Cityflow (Simulator), directly transferred to SUMO (Real), and transferred with our method to SUMO (GAT-Real).

Setting	Simulator		Real		GAT-Real	
	ATT	TP	ATT (gap)	TP (gap)	ATT (gap)	TP (gap)
V1	111.23	1978	158.93 (47.69)	1901 (77)	**149.13 (37.88)**	**1926 (52)**
V2	111.23	1978	177.27 (66.03)	1898 (80)	**162.80 (51.56)**	**1926 (52)**

IV. Conclusion

We first investigate the actual gap existing in the sim2real Traffic Signal Control problems and propose a series of methods to reduce the gap including grounding action transformation, uncertainty quantification, and dynamically adjustable grounding rate. Our results reveal the great potential in this area for future exploration.

References

[1] Wang M, Deng W. Deep visual domain adaptation: A survey[J]. Neurocomputing, 2018, 312: 135-153.

[2] Farchy A, Barrett S, MacAlpine P, et al. Humanoid robots learning to walk faster: From the real world to simulation and back[C]//Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems. 2013: 39-46.

[3] Lopez P A, Behrisch M, Bieker-Walz L, et al. Microscopic traffic simulation using sumo[C]//2018 21st international conference on intelligent transportation systems (ITSC). IEEE, 2018: 2575-2582.

[4] Zhang H, Feng S, Liu C, et al. Cityflow: A multi-agent reinforcement learning environment for large scale city traffic scenario[C]//The world wide web conference. 2019: 3620-3624.

[5] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. nature, 2015, 518(7540): 529-533.

[6] Hanna J, Stone P. Grounded action transformation for robot learning in simulation[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2017, 31(1).

[7] Sensoy M, Kaplan L, Kandemir M. Evidential deep learning to quantify classification uncertainty[J]. Advances in neural information processing systems, 2018, 31.