Highway Merging Control Using Multi-Agent Reinforcement Learning

Ali Irshayyid and Jun Chen*, *Senior Member, IEEE*
Department of Electrical and Computer Engineering
Oakland University
Rochester, MI 48309, USA
Email: {aliirshayyid,junchen}@oakland.edu

2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI) | 979-8-3503-7297-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICMI60790.2024.10585649

***Abstract*—This paper presents a multi-agent reinforcement learning approach for autonomous vehicle highway merging control. A decentralized partially observable Markov decision process is formulated, where each autonomous vehicle acts independently based on local observations. The scenario considered in this paper assumes randomly spawning vehicles and fluctuating traffic flows and a self-attention network is used to handle varying numbers of agents (vehicles). The proposed method is validated in SUMO traffic simulator, which provides a realistic highway simulation environment. Results demonstrate the approach can enable safe, efficient coordination for merging maneuvers, successfully handling dynamic number of agents. Future work will continue to enhance multi-agent reinforcement learning for autonomous vehicle coordination in complex traffic environments by reducing the training time.**

I. Introduction

AUtonomous vehicles (AVs) promise major benefits in terms of safety, efficiency, and accessibility [1]–[3]. However, developing reliable control policies for AVs remains an immense challenge [4]. A key difficulty involves handling merging scenarios, where AVs must interact safely and efficiently both among themselves and with human drivers with diverse driving styles [5]. To address this, recent research has explored deep reinforcement learning (DRL) for learning AV merging policies that map observations to actions. DRL leverages deep neural networks as powerful function approximators to handle complex, high-dimensional state spaces [6]. Moreover, multi-agent reinforcement learning (MARL) [7], where agents learn by not only interacting with the environment but also by taking into account the actions and strategies of other agents, has also been studied for AV merging control.

However, most existing work uses predefined number of vehicles that remain constant during the training episode [8]. Therefore the trained policy cannot be deployed in real-world where the number of vehicles can be time varying. To address this issue, this study proposes a dynamic environment where the number of AVs, within a single episode, varies over time. Specifically, the MARL is employed to control AVs attempting to merge onto a highway already occupied by human-driven vehicles (HDVs). See Fig. 1, where the goal is to determine the optimal merging point to achieve the highest traffic flow. In particular, the highway environment is modeled as a two-lane road, where the right lane terminates after 300 meters, necessitating the merging of vehicles from the right lane to the left lane. This environment is simulated in the open-source microscopic traffic simulator SUMO (Simulation of Urban Mobility) [9], where newly arriving vehicles enter the highway randomly from either lane. The focus on a dynamic environment and varying number of learning agents in this work not only tests the scalability of MARL algorithms to changing traffic conditions but also aligns more closely with the fluctuating real-world traffic flows and tests generalization.

II. Environment

The SUMO simulator offers a Traffic Control Interface (TraCI) [10] to enable the interaction between the environment and the RL agents. Vehicle longitudinal dynamics are modeled using the Krauss car-following model [11]. Furthermore, the default SUMO lane change decision models are disabled and replaced with RL actions, while the default lane change control model is utilized to execute lane changes initiated by RL.

The state space representation in this work includes a local observation for each agent, comprised of the ego agent's speed, distance to the merge point, and state of merge (1 if merged and 0 if not yet merged) as well as the relative longitudinal position and speed of surrounding vehicles within 8 meters range. This local observation is normalized and then passed to the actor network. The full state representation used for the critic is created by concatenating the local observations of all active agents. This allows the actor to act based on its local view, while the critic evaluates state-action values using the global observations to enable learning cooperative merging behaviors. The action space for each agent is defined as a discrete binary set, where an action value of 0 indicates the agent should maintain its current lane, while an action of 1 signals a request to perform a merge into the adjacent lane.

The reward function consists of two main components - a safety reward R_c and a speed reward R_s, with a weighting term w applied to combine them into a total reward R_t. The safety reward R_c penalizes the agent for getting too close to the merging point and is defined as: $R_c = -\left(\frac{x-d}{d}\right)^2$ if $x \leq d$

This work is supported in part by SECS Faculty Startup Fund from Oakland University and National Science Foundation through Award #2237317.

*Jun Chen is the corresponding author.

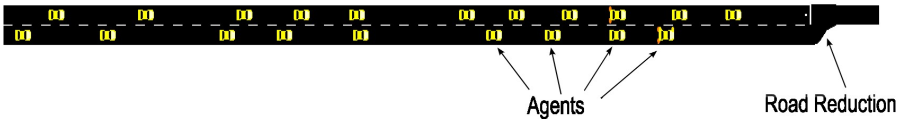

Fig. 1. The AV merging scenario in SUMO simulation environment.

and otherwise $R_c = 0$, where x is the distance from the agent to the merge point, and d is a defined threshold distance.

The speed reward R_s is based on the average speed of the ego vehicle and surrounding vehicles as $R_s = \frac{1}{N}\sum_{i=0}^{N-1} s_i$, where s_i is the speed of vehicle i and N is the total number of vehicles. Only vehicles within close longitudinal proximity are included to avoid credit assignment problem. The total reward R_t combines the coordination and speed rewards as $R_t = w_1 R_s + w_2 R_c$. This provides a composite reward signal that balances optimizing speed while achieving safe, coordinated merging maneuvers.

III. Multi-agent Reinforcement Learning

The problem is modeled as a decentralized partially observable Markov decision process (Dec-POMDP) [12] defined by the tuple: $(\mathbf{S}, \mathbf{O}, \mathbf{A}, R, P, n, \gamma)$. Here, $\mathbf{S}$ is the state space. $\mathbf{O}$ is the joint local observation space of all agents. $\mathbf{A}$ is the joint action space for all agents. $P(S' \mid S, A)$ is the transition probability to S' given the current S and $A = (a_1, \ldots, a_n)$. γ is the discount factor. R is the immediate shared reward received when taken the actions A at state S. This work uses a Centralized Training Decentralized Execution (CTDE) MARL framework, adopted from [13], for multi-agent automated traffic control with a dynamic number of agents. At each time step, SUMO provides local traffic state observations to each actor agent. The actors independently choose actions using their policies $\pi_\theta(a|s)$ parameterized by deep neural networks (DNNs). A centralized critic network utilizes self-attention to evaluate state-action values based on the combined observations of all currently active agents, allowing the critic to handle varying number of agents at each time-step. Additionally, a centralized counterfactual baseline network with self-attention is used to estimate each agents contribution. For each active agent, the baseline marginalizes out the agent's action and conditions on other agents' observations and actions.

IV. Preliminary Results and Discussion

The learning process achieved increasingly higher mean rewards over the training process, as shown in Fig. 2. The increasing reward progression over training indicates that the RL agents successfully learned policies to navigate the merging scenario through balancing the composite reward comprising of safety, coordination, and speed. While these preliminary results are promising, further investigation is required to fully assess the approach's robustness. The current training time of approximately 90 hours is extensive and warrants reduction through methods such as hyperparameter tuning and optimized implementation. The technique currently determines agent contributions using a third neural network. Although functional, this solution adds complexity and resource overhead. Reward shaping, which uses expert rewards to quantify per-agent performance, is worth investigating. Exploring augmented state representations and alternate reward formulations may also yield further performance improvements.

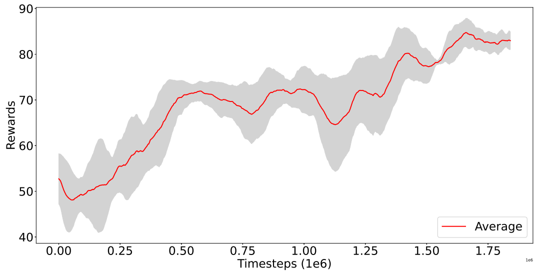

Fig. 2. Training results.

References

[1] J. Deichmann, *Autonomous Driving's Future: Convenient and Connected.* McKinsey, 2023.

[2] T. Litman, "Autonomous vehicle implementation predictions: Implications for transport planning," 2020.

[3] Z. Zhou, C. Rother, and J. Chen, "Event-triggered model predictive control for autonomous vehicle path tracking: Validation using CARLA simulator," *IEEE Trans. Intel. Veh.*, vol. 8, no. 6, pp. 3547–3555, June 2023.

[4] J. Chen and Z. Yi, "Comparison of event-triggered model predictive control for autonomous vehicle path tracking," in *IEEE Conf. Control Technology and Applications*, Austin, TX, USA, 2021, pp. 808–813.

[5] T. Gindele, S. Brechtel, and R. Dillmann, "Learning driver behavior models from traffic observations for decision making and planning," *IEEE Intelligent Transp. Syst. Mag.*, vol. 7, no. 1, pp. 69–79, 2015.

[6] A. Irshayyid and J. Chen, "Comparative study of cooperative platoon merging control based on reinforcement learning," *Sensors*, vol. 23, no. 2, pp. 1–23, 2023.

[7] L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 38, no. 2, pp. 156–172, 2008.

[8] J. Dinneweth, A. Boubezoul, R. Mandiau, and S. Espié, "Multi-agent reinforcement learning for autonomous vehicles: A survey," *Autonomous Intelligent Systems*, vol. 2, no. 1, p. 27, 2022.

[9] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility: an overview," in *Proceedings of SIMUL 2011, The Third Int. Conf. on Advances in System Simulation*, 2011.

[10] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux, "TraCI: an interface for coupling road traffic and network simulators," in *Proc. Comm. Net. Simulation Symp.*, 2008, pp. 155–163.

[11] S. Krauß, P. Wagner, and C. Gawron, "Metastable states in a microscopic model of traffic flow," *Physical Review E*, vol. 55, no. 5, p. 5597, 1997.

[12] F. A. Oliehoek, C. Amato *et al.*, *A concise introduction to decentralized POMDPs.* Springer, 2016, vol. 1.

[13] A. Cohen, E. Teng, V.-P. Berges, R.-P. Dong, H. Henry, M. Mattar, A. Zook, and S. Ganguly, "On the use and misuse of absorbing states in multi-agent reinforcement learning," *arXiv preprint arXiv:2111.05992*, 2021.