Chair of Traffic Engineering and Control
TUM School of Engineering and Design
Technical University of Munich

REINFORCEMENT LEARNING BASED STRATEGY FOR HUMAN DRIVERS IN LANE FREE TRAFFIC

Mian Awais Ali (03774190)
E-mail: mianawais.ali@tum.de
Mentors: Arslan Ali Syed Dr.-Ing, Athanasia Karalakou M.Eng

Introduction

Lane-Free Traffic (LFT) enables Connected and Autonomous Vehicles (CAVs) to utilize the full road width, rather than being confined to fixed lanes.
Simulation results show that LFT can more than double road capacity, with full CAV penetration (Rostami et al., 2022).

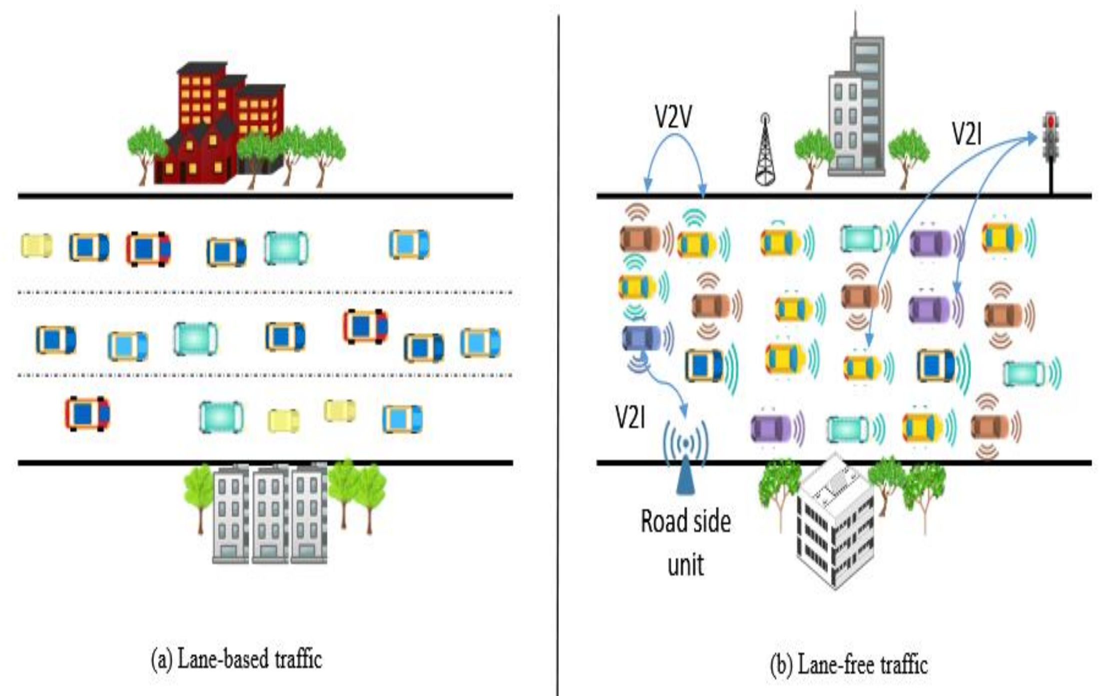


Figure 1. (a) Lane-less mixed traffic, (b) Lane-free traffic with CAVs. Source (Sekeran et al., 2022)

Lane-Free Traffic (LFT) Controller

Potential Line Controller(PL) (Rostami et al., 2022).

- Road used as a **continuous surface** (no fixed lanes).
- **Potential Lines (PL)** assign lateral positions based on desired speed.
- Artificial forces → collision avoidance + smooth coordination

Even a **small number of Human-Driven Vehicles (HDVs)** causes major performance loss in LFT when using the Potential Lines PL Controller:

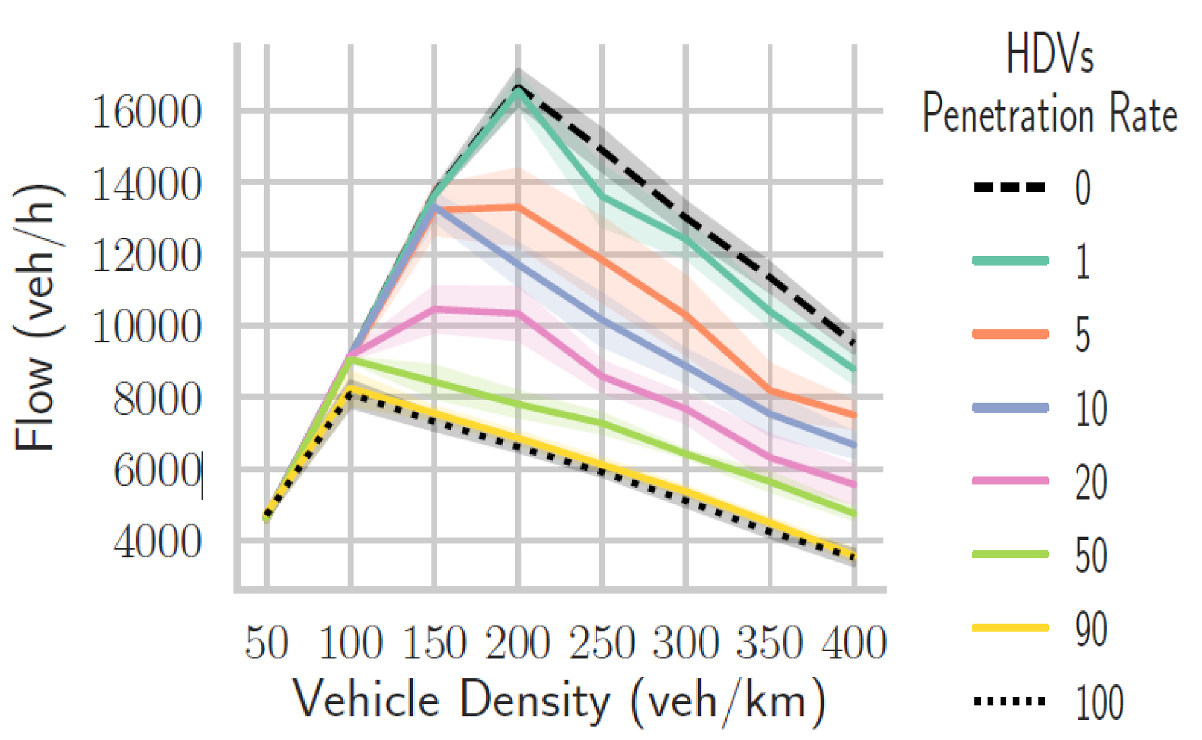


Figure 2: Traffic flow rate and mean speed of the PL controller with different HDV penetration rates. Source (Syed et al., 2025).

Adaptive Potential Lines (APL) for Mixed Traffic

- Creates an APL corridor (green area in Figure 2) around each HDV.
- **Compresses PLs** into free lateral gaps available between the HDVs inside the corridor.
- Allows **coordinated overtaking**.
- Effectiveness depends on **how the margin** x_j^{th} in (meter) is chosen.

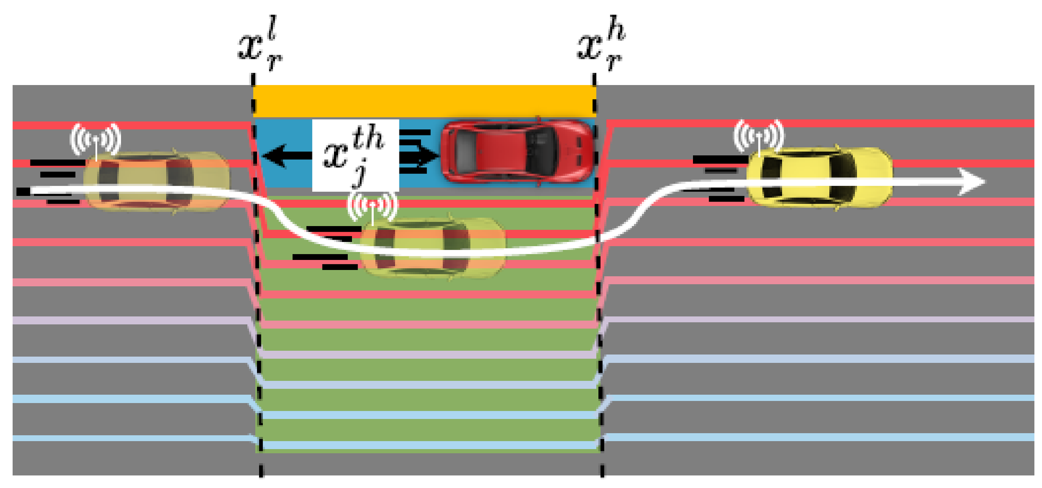


Figure 3: APL controller. It adapts the PL areas near HDVs, represented by the green colour and referred to as the APL corridor in the study. The blue colour shows the areas marked as occupied by HDVs. Source (Syed et al., 2025).

Objectives of the study

- To learn the margin x_j^{th} for each HDV by Reinforcement Learning RL.
- To formulate **the** Markov Decision Process**(MDP)** and analyse the results.

MDP Formulation

Framework = Centralized Learning and Decentralized execution **(CTDE)** in RayRlib.
Algorithm = Proximal Policy Optimization **(PPO)**

- **State S**

Consists of the Actor and the Critic observations

1. Actor Observations

Local state observation around each HDV, including:

HDV speed and gap, Nearby CAV/HDV speeds, Distance to leader and follower, Local density, Current action, Previous action. Lateral gaps availability inside the corridor.

2. Critic Observations

Global signals used by the centralized critic:

Average CAV/HDV speeds, Normalized global flow, Average Corridor Size, HDV percentage (current episode), Average number of CAVs inside the corridor, and current policy action.

- **Action A**

Continuous action: Normalized action range (-1,1) mapped to the physical range margin in meters,

$$a_j \in [-1, 1] \;\rightarrow\; x_j^{th} \in [0, m_{\max}]$$

Where m_{max}= 50m (maximum allowed margin)

- **Reward R**

Two reward functions were formulated, named as,
1. Reward Design A
2. Reward Design B

That are the weighted combination of:

- **Flow ↑**
- **APL Corridor Efficiency ↑**
- **Corridor Geometry**
- **Instability of Corridor ↓**
- **Low Lateral speed ↓**
- **Margin fluctuation ↓**

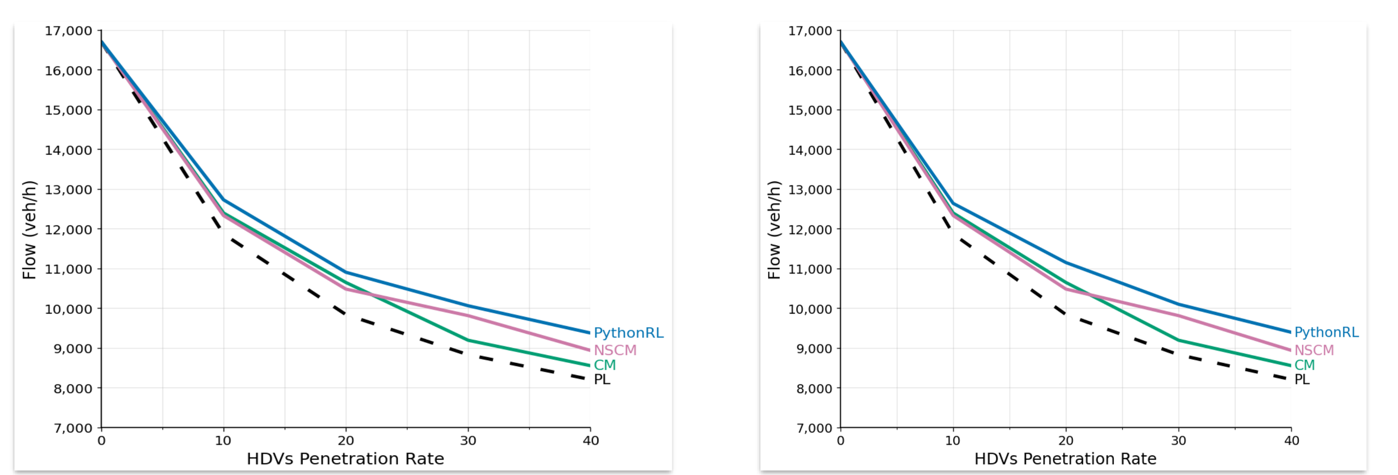
Training and Evaluation Setup.

- **1-km ring**, **10.2-m width**, traffic density **200 veh/km**
- HDV penetration rates: **10%, 20%, 30%, 40%**
- **Traffic Fluid Sim (SUMO)**

Results

Trained Policies **(PythonRL)** with two reward functions, **A & B,** were evaluated against PL controller and APL controller strategies (Constant Margin **(CM)** and Neighbour speed-based constant Margin **(NSCM)**, in terms of **Flow and Comfort.**
Flow rate at 40 % HDV for **Reward Design A and B**

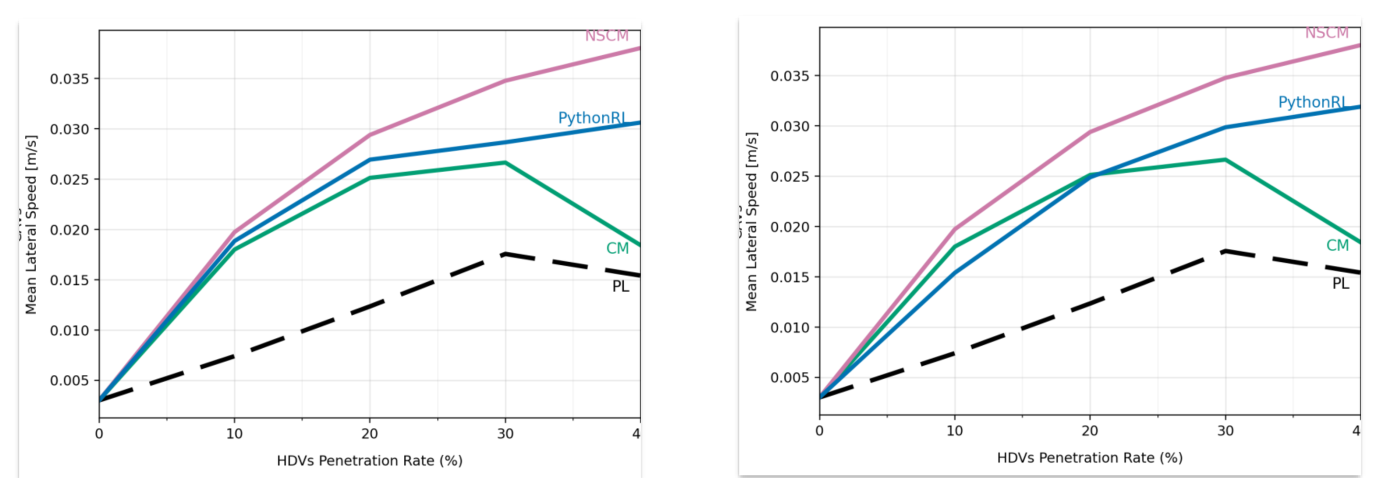
Flow Vs HDV Penetration(10 to 40%)



Reward Design A	Reward Design B
Flow(PythonRL: 9,372 veh/h	Flow(PythonRL): 9,390 veh/h

Figure 4: Flow vs HDV penetrations. At 40 % HDV, flow of PL: 8,202 veh/h, CM: 8,550 veh/h, NSCM: 8,934 veh/h

CAVs Lateral Speed (Comfort)



Reward Design A	Reward Design B
~18% lower lateral speed than NSCM 0,031m/s at 40% HDV	~18% lower lateral speed than NSCM 0,033m/s at 40% HDV

Figure 5: CAVs' Lateral speed vs HDV penetrations.

Conclusion

- RL-trained policies, when evaluated by both reward functions, improve LFT performance.
- Both reward functions outperform PL, CM, and NSCM in all HDV scenarios.
- Up to **+14.5%** higher flow at 40% HDV against PL, **+9.6 %** vs CM, **+4.9 %** vs NSCM.
- Smoother and safer lateral motion
- Generalizes well across 10–40% HDV.RL shows strong potential for mixed traffic.

References

- Ghasemi, M., Ebrahimi, D., 2024. Introduction to reinforcement learning. URL: https://arxiv.org/abs/2408.07712, arXiv:2408.07712.
- Rostami-Shahrbabaki, M., Zhang, H., Sekeran, M., Bogenberger, K., 2022. Increasing the capacity of a lane-free beltway for connected and automated vehicles using potential lines. Manuscript submitted 7 December 2022.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. URL: https://arxiv.org/abs/1707.06347, arXiv:1707.06347.
- Papageorgiou, M., Mountakis, K., Karafyllis, I., Papamichail, I., 2019. Lane-free artificial-fluid concept for vehicular traffic. CoRR abs/1905.11642. URL: http://arxiv.org/abs/1905.11642, arXiv:1905.11642.
- Sekeran, M., Rostami-Shahrbabaki, M., Syed, A.A., Margreiter, M., Bogenberger, K., 2022. Lane-free traffic: History and state of the art, in: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), IEEE, Macau, China. pp. 1037–1042. doi:10.1109/ITSC55140.2022.9922282. main conference Oct 8–12, 2022.
- Syed, A.A., Rostami-Shahrbabaki, M., Bogenberger, K., 2025. Can human drivers and connected autonomous vehicles co-exist in lane-free traffic? a microscopic simulation perspective. Transportation Research Part C: Emerging Technologies 180, 105315. URL: http://dx.doi.org/10.1016/j.trc.2025.105315, doi:10.1016/j.trc.2025.105315.
- Ray Project, 2025. Rllib algorithms — official documentation. https://docs.ray.io/en/latest/ rllib/rllib-algorithms.html.