Cooperative Cruising: Reinforcement Learning-Based Time-Headway Control for Increased Traffic Efficiency

Yaron Veksler[1*], **Sharon Hornstein**[1†], **Han Wang**[2], **Maria Laura Delle Monache**[2], **Daniel Urieli**[1‡],

[1] General Motors R&D Labs
[2] University of California, Berkeley
[*]yaron.veksler@gm.com, [†]sharon.hornstein@gm.com, [‡]daniel.urieli@gm.com

Abstract

The proliferation of connected automated vehicles represents an unprecedented opportunity for improving driving efficiency and alleviating traffic congestion. However, existing research fails to address realistic multi-lane highway scenarios without assuming connectivity, perception, and control capabilities that are typically unavailable in current vehicles. This paper proposes a novel AI system that is the first to improve highway traffic efficiency compared with human-like traffic in realistic, simulated multi-lane scenarios, while relying on existing connectivity, perception, and control capabilities. At the core of our approach is a reinforcement learning based controller that dynamically communicates time-headways to automated vehicles near bottlenecks based on real-time traffic conditions. These desired time-headways are then used by adaptive cruise control (ACC) systems to adjust their following distance. By (i) integrating existing traffic estimation technology and low-bandwidth vehicle-to-infrastructure connectivity, (ii) leveraging safety-certified ACC systems, and (iii) targeting localized bottleneck challenges that can be addressed independently in different locations, we propose a potentially practical, safe, and scalable system that can positively impact numerous road users.

1 Introduction

Highway congestion has widespread social impacts, including disproportionately affecting low-income communities with longer commutes, increased pollution, high stress levels, and reduced economic productivity (Lomax, Schrank, and Eisele 2021; Fattah, Morshed, and Kafy 2022). The proliferation of connected automated vehicles (CAVs) equipped with technologies such as adaptive cruise control (ACC) represents an unprecedented opportunity to utilize these technologies to improve highway traffic flow and reduce congestion (Stern et al. 2018; Wu, Bayen, and Mehta 2018; Delle Monache et al. 2019).

Prior research on highway congestion reduction explored distributed and centralized vehicle speed control approaches. Distributed approaches typically implement an in-vehicle speed controller that uses information of the vehicle's surroundings to decide when to increase headway and allow vehicles to merge into its lane. These approaches are scalable and effective when the location and time of lane-changes can be accurately predicted, such as in merge and certain bottleneck scenarios (Cui et al. 2021; Zhang et al. 2023a; Vinitsky et al. 2023). However, in multi-lane scenarios, lane changes can occur unpredictably at any time and any point on the road, driven by drivers' intentions and behaviors, rendering these distributed approaches ineffective. Centralized approaches leverage aggregate traffic data to provide high-level guidance for influencing spatio-temporal traffic density (Bayen et al. 2020). While such approaches circumvent the need to precisely predict lane changes, they face challenges in generating the complex speed-control commands needed for influencing traffic density to reduce congestion.

To address these challenges, we propose a centralized AI system that influences density more directly by generating time-headway requests that are used by ACC systems to adjust their vehicles' headways. At the core of our systems is a reinforcement learning based controller that continually outputs desired vehicle headways for each road segment leading to the bottleneck, based on real-time traffic conditions. To narrow the gap to real-world deployment, our system is designed to integrate with existing traffic estimation technology, low-bandwidth vehicle-to-infrastructure connectivity, and safety-certified ACC systems. Through hundreds of large-scale simulated experiments, we show that our system significantly improves traffic flow compared to human-driven traffic, particularly in scenarios where previous methods fall short. As a secondary contribution, we address a previously identified flaw in measuring average speed in simulated road networks where vehicles enter and leave the simulation dynamically (Cui et al. 2021), and introduce a method to better approximate realistic average speed values. By leveraging existing technologies, ensuring safety through ACC systems, and addressing localized bottleneck challenges, we propose a potentially practical, safe, and scalable system that could enhance the travel experience of numerous road users. Our code is publicly available on GitHub[1].

2 Related Work

Traffic congestion poses a significant challenge in highway planning, prompting the development of traffic flow models

[1] https://coopcruise.github.io/

to understand and mitigate its adverse impacts (Hall 1996; Ni and Leonard II 2006; Ferrara et al. 2018; Mohammadian et al. 2021). Studies have shown that traffic breakdowns can occur spontaneously and stochastically, even without the presence of bottlenecks (Sugiyama et al. 2008), a phenomenon that classical traffic flow theories fail to adequately explain (Kerner 2016). Traffic microsimulators, such as SUMO (Krajzewicz et al. 2012) which we use in this paper, were suggested for reproducing observable traffic phenomena, using *car-following models* of human driving. One such model is the Intelligent Driver Model (IDM), which has been instrumental in reproducing complex phenomena like traffic jams, stop-and-go waves, and bottleneck congestion (Treiber, Hennecke, and Helbing 2000). IDM can reproduce realistic vehicle interactions, particularly when modeling diverse traffic scenarios, and is superior to other models such as the Optimal Velocity Model (Bandō et al. 1995) that tends to smooth out traffic by encouraging vehicles to maintain optimal velocity.

The idea to utilize automated vehicles as mobile actuators for alleviating traffic problems is based on the assumption that these vehicles can be systematically coordinated and controlled to optimize traffic flow (Stern et al. 2018; Wu, Bayen, and Mehta 2018; Delle Monache et al. 2019; Wang et al. 2023a,b). Inspired by this idea, the CIRCLES project (CIRCLES 2020) has focused on developing traffic control algorithms to optimize traffic flow on highways, and conducted a large-scale, open-road field experiment with 100 CAVs (Wang et al. 2024a; Lee et al. 2024).

Since developing model-based controllers for traffic congestion is challenging due to the problem's scale and complexity, research on data-driven controllers trained in simulation using reinforcement learning (RL) (Sutton and Barto 2018) has emerged. This line of research demonstrated that centralized RL controllers enable wave dissipation and significant average speed increase in single-lane roads with merges or bottlenecks with as low as 10% CAV penetration (Kreidieh, Wu, and Bayen 2018; Vinitsky et al. 2018; Wang et al. 2024b). However, these approaches may not scale well, as learning a single policy to individually control numerous vehicles becomes infeasible due to the high-dimensional action space.

To enable scalability, distributed approaches have employed in-vehicle speed controllers that utilize information about a vehicle's surroundings to determine when to increase headway and allow other vehicles to merge into its lane. These approaches work well when the location and time of lane-changes can be accurately predicted, e.g. in merge and certain bottleneck scenarios (Cui et al. 2021; Zhang et al. 2023a; Vinitsky et al. 2023). However, these approaches become ineffective in multi-lane scenarios, where lane changes occur unpredictably at any time and place.

As an alternative, centralized approaches have used aggregate traffic data to provide high-level guidance without modeling local behaviors such as lane-changes. The Variable Speed Limits (VSL) method (Hegyi, De Schutter, and Hellendoorn 2005; Lu and Shladover 2014) aims to prevent traffic breakdowns by regulating inflow into congestion-prone areas. Extensive studies of VSL (Alasiri, Zhang, and Ioannou 2023; Zhang et al. 2023b; Hua and Fan 2023) have explored its application through optimization problems, such as minimizing total travel time or maximizing traffic throughput (Bayen et al. 2022). However, these approaches have struggled to address realistic multi-lane scenarios due to the difficulty of achieving the required road dynamics when only speed limits are controlled (as discussed in Section 4.2). To overcome these limitations, we propose a centralized AI system that avoids the need to predict lane changes or implement complex speed control. Instead, our method employs time-headway control to directly influence spatio-temporal traffic density to enhance traffic efficiency. Prior work (Pang and Huang 2022) proposed using time headways for platoon leaders, whereas our method assigns them to road segments.

3 Domain Description

In this section, we define the problem addressed by this paper and the simulation setup used in the experiments.

3.1 Problem Description

Traffic congestion frequently occurs when the demand for road use exceeds the available capacity. Addressing this imbalance by reducing demand typically involves long-term, systemic changes, such as enhancing public transportation infrastructure. Therefore, in this paper we assume that demand is given, and focus on optimizing road capacity. Road capacity can be optimized by influencing driving behavior to mitigate the impact of capacity-reducing phenomena, such as lane changes at bottlenecks.

Our problem is defined as follows. Given a road network with multiple lanes, a merging road, and mixed autonomy traffic consisting of both human-driven vehicles and CAVs, maximize the network's traffic efficiency by controlling CAVs, where traffic efficiency is measured in terms of average speed. We assume that CAVs are altruistic, sharing the common goal of reducing traffic congestion, which can be facilitated by incentivizing such behavior. A solution to our problem is a control policy that maps the traffic state to actions that influence CAVs to enhance traffic efficiency. For reasons described in Section 4, we propose and focus on control policies that influence CAVs equipped with technologies such as ACC, by sending them time-headway commands based on real-time traffic information.

We note that in simulated *open road networks* where vehicles enter and exit dynamically, increased average speed may not reflect real-world traffic flow improvements due to discrepancies between finite simulated roads and real-world conditions (Cui et al. 2021). We therefore propose a novel average speed metric that aligns more closely with real-world scenarios, enabling more reliable evaluations of average speed improvements in open road simulations (Section 4.1).

3.2 Simulation Platform & Scenario Parameters

To test our system in large-scale simulations, we use the SUMO traffic simulator (Krajzewicz et al. 2012), which dynamically models all vehicles and their interactions. We interface SUMO with a custom environment following the

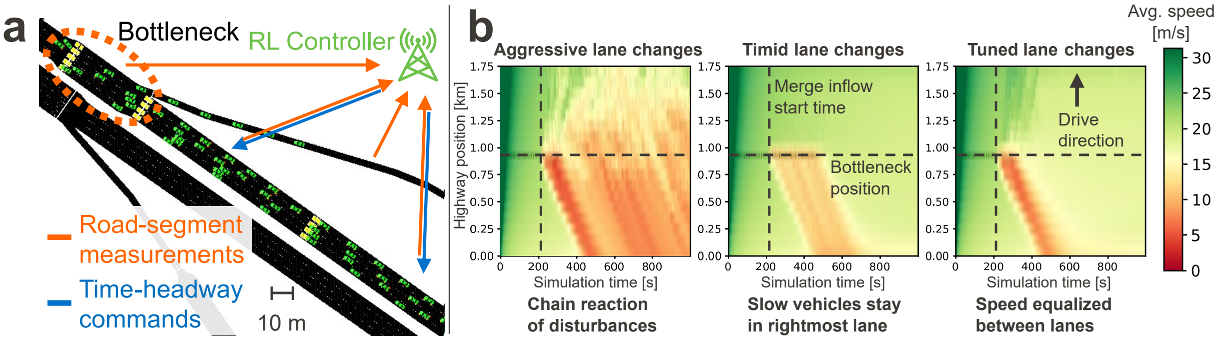

Figure 1: **Centralized Time-headway control for multi-lane highway congestion reduction:** (a) The analyzed scenario. An RL-based controller sends time-headway commands to CAVs near bottlenecks, based on measured traffic speed and density. (b) Lane-changing behavior simulation results. Aggressive lane-changing behavior significantly impacts traffic dynamics.

Gymnasium API (Towers et al. 2024), enabling the training of controllers using reinforcement learning algorithms from the RLlib library (Duan et al. 2016).

The road network used in our simulations is a 2 km segment of I-24 in Tennessee, USA, extracted from OpenStreetMap (2017), chosen as a representative example of a general highway merge geometry. The simulation includes hundreds of vehicles engaging in complex lane-changing and car-following interactions, with vehicles merging onto the highway and executing required lane changes. Prior studies (Samaei et al. 2023) modeled a larger section of I-24 using real-world traffic data. Figure 1a presents a partial snapshot of our scenario in SUMO, featuring a centralized control policy that senses traffic conditions and issues time-headway commands. These commands are utilized by ACC systems to dynamically adjust vehicle headways.

Human-driven vehicles are modeled using IDM (Treiber, Hennecke, and Helbing 2000) which has a constant time-headway parameter and a safety enforcement module. CAVs are modelled using IDM with a time-headway parameter that can be dynamically adjusted. Before real-world deployment, automated vehicle models should be calibrated with real-world data prior to using them with time-headway controllers, however due to the lack of such data and the aim of ACC systems to follow human-driving behaviors, we used the aforementioned adjustable IDM models. An additional challenge is modeling lane-change behavior, which significantly affects traffic flow (Figure 1b). Aggressive lane-change behavior, characterized by vehicles merging into smaller gaps, causes major disturbances and decreases both speed and throughput. Timid lane-change behavior cause disturbances by low utilization of additional lanes. We adjusted lane-change aggressiveness through visual inspection to better align with real-world driving behavior. Lane-change behavior should ideally also be calibrated with highway data if available. Values of all SUMO parameters can be found in the Appendix.

4 Methodology

In this section, we present our methodology for designing and testing the proposed time-headway controllers. First, we introduce an average-speed metric that addresses the limitations of average-speed measurements in simulated open road networks which were pointed out by prior research. Next, we analyze vehicle interactions contributing to traffic inefficiencies, identify the shortcomings of existing methods, and justify our centralized time-headway control approach. We continue by outlining our practical design choices. Finally, we describe a baseline fixed-value time-headway control policy, and our proposed RL-based control policy.

4.1 Congestion Metrics

To demonstrate improvements in traffic flow efficiency, it is essential to define how congestion is measured. Average speed, average throughput, and time delay with respect to travel time at maximum speed are commonly used metrics, but have drawbacks in simulations. Average throughput is sensitive to simulation length, since a temporary throughput decrease can be offset by a subsequent increase, for long enough simulations including periods with demand lower than road capacity. While changes in average speed or time delay can effectively measure congestion, caution is needed when using these metrics in finite road length simulations: controllers which prevent vehicles from entering the simulation might artificially inflate speeds without being penalized for the reduction of speeds that would have happened in real-world road sections preceding the simulated road network (Cui et al. 2021).

We solve the aforementioned problem by tracking vehicles whose entry to the simulation is delayed and penalizing for their delay time. The metric we use is the average speed change compared to a simulated human-driven traffic, taking into account the entry delay time of each of the vehicles:

$$\Delta V \equiv \frac{1}{N}\sum_{i}^{N}\frac{\bar{v}^{(i)}_{\text{control}}-\bar{v}^{(i)}_{\text{baseline}}}{\bar{v}^{(i)}_{\text{baseline}}}, \qquad (1)$$

where N is the total number of vehicles, $\bar{v}^{(i)}_{\text{baseline}}$ is vehicle i's

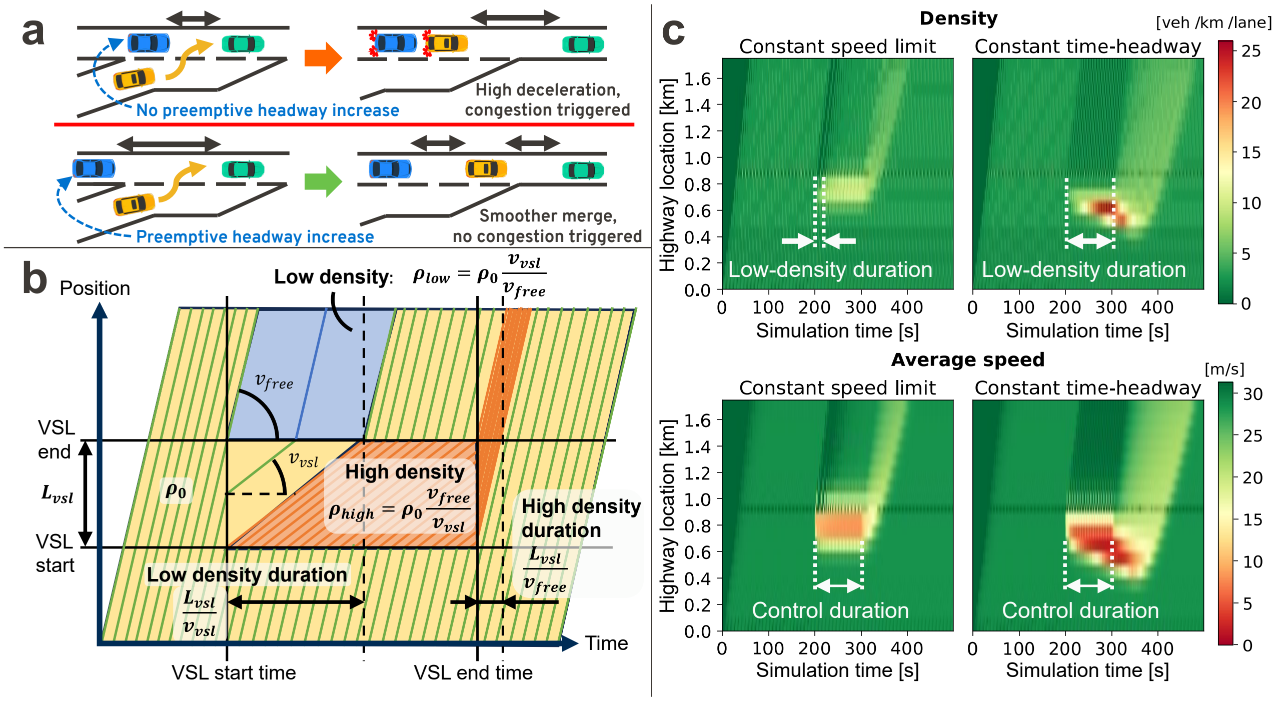

Figure 2: **Time-headway control motivation:** (a) Aggressive lane-change may cause excessive speed and throughput decrease (top). Preemptively increasing headway can reduce the negative effect (bottom). (b) Analysis of constant speed-limit command dynamics. Low downstream density is maintained for a limited duration (blue). (c) Simulation results for constant speed and time-headway signals. Constant time-headway signals maintain lower downstream density for arbitrary duration.

average speed in simulated human-driven traffic, and $\bar{v}^{(i)}_{\text{control}}$ is the same vehicle's speed in a simulation with headway control. Note that this metric takes all vehicles into account, ensuring that traffic-blocking strategy is not a viable solution.

Average speed computations assume that delayed vehicles have a speed of 0 while they wait to enter the simulation:

$$\bar{v}^{(i)} = \frac{L^{(i)}}{\min\left(T_f^{(i)}, T_{sim}\right) - T_s^{(i)}}, \tag{2}$$

where $L^{(i)}$ is the distance driven by vehicle i, $T_s^{(i)}$ is its *planned* simulation entry time, $T_f^{(i)}$ is its *measured* exit time, and T_{sim} is the final simulation time. $T_s^{(i)}$ of each vehicle is known apriori and stored in a per-vehicle route file used to initialize the SUMO simulation.

4.2 Centralized Time-Headway Control

To motivate our centralized time-headway control approach, we identify and address factors that may have prevented prior methods from handling realistic multi-lane scenarios. Prior research has shown that lane changes during merges can cause slowdowns and trigger congestion. However, if a vehicle in the target lane preemptively increases its headway as it approaches the merge, it can prevent congestion from forming, as illustrated in Figure 2a. Distributed approaches typically implement an in-vehicle speed controller that uses information of the vehicle's surroundings to decide when to preemptively increase headway. These approaches are scalable and effective when the location and time of lane-changes can be accurately predicted, such as in merge and certain bottleneck scenarios. However, in multi-lane scenarios, lane changes can occur unpredictably at any time and place on the road, driven by drivers' intentions and behaviors, rendering these distributed approaches ineffective.

In such scenarios, it may be desirable to try approaches for higher-level control over road density, motivated by real-world data showing that lower traffic density reduces the negative impact of lane changes on traffic flow (Yang, Wang, and Quddus 2019; Gao and Levinson 2023). Since density is an aggregate measure, it is naturally controlled using a centralized controller. Centralized approaches that attempt to control each CAV individually face scalability issues (Cui et al. 2021). Instead, centralized approaches such as Variable Speed Limits used aggregate traffic data to provide high-level guidance by employing speed-limit adjustments in controlled road segments. However, these approaches struggle with controlling traffic density. The qualitative vehicle trajectory time-space diagram in Figure 2b shows that applying a constant speed limit in a road segment can reduce downstream density only temporarily (blue area). In contrast, constant time-headway signals maintain lower downstream density for an arbitrary duration. This is empirically demonstrated in Figure 2c, where a constant time-headway control signal results in a complex, time-dependent upstream speed profile that would be difficult to achieve with speed-limit control.

We develop a centralized time-headway control capable of reducing traffic density around bottlenecks, allowing vehicles to self-organize in sparser traffic, as often observed in real-world uncontrolled traffic. Our system offers a scalable solution, as its use of local traffic information and control enables independent deployment at numerous highway junctions.

4.3 Practical Design Choices

An AI system for CAVs equipped with technologies such as ACC must address several key objectives to be practical:

- **Safety:** The system should enhance traffic flow while maintaining safe operations.
- **Simplicity and Generality:** To ensure broad applicability and ease of implementation across diverse environments, the system must be simple and generalizable.
- **Deployability and User Acceptance:** For successful deployment, the system's decisions must be transparent to instill confidence in drivers and stakeholders.

To address safety, our system is designed to influence existing safety-certified ACC systems by sending them desired time-headway commands that can only *increase* the time-headway above its default value. To promote generality, we model a representative scenario of a typical highway merge. We simplify implementation by designing state and action spaces independent of vehicle count, and assume the availability of low-bandwidth vehicle-to-infrastructure communication, where vehicles periodically receive a few floating-point values representing desired time-headways. The infrastructure is assumed to measure traffic metrics across different road segments, such as average speed, density, and throughput. This type of sensing is already available, and was recently used in a large-scale open-road experiment (Lee et al. 2024), supporting the deployability of our proposed system. The system's time-headway decisions can be made transparent to users and support user-acceptance.

4.4 Fixed-Valued Time-Headway Control

Before proposing an RL-based time-headway control policy, it is natural to ask whether a simpler fixed-value time-headway policy can outperform human-driven traffic and, if so, whether RL can provide significant additional improvements. We therefore design a controller that sends an optimized fixed time-headway signal to road segments located before a bottleneck. The controller activates when vehicles are detected on the merging road within 200 meters before the merge and deactivates when no vehicles are present in this segment. When deactivated, vehicles return to their default time-headway. We find the optimal time-headway value using a parameter sweep.

4.5 RL-based Time-Headway Control

The dynamic nature of traffic suggests that a controller capable of adjusting time-varying headway values would outperform a fixed time-headway approach (Yanakiev and Kanellakopoulos 1995). RL's ability to learn from environment interactions makes it a good fit for complex, dynamic, and stochastic setups such as traffic control. We design an RL-based controller that continuously monitors the traffic state, determines time-headway commands for each controlled road segment, and communicates these commands to automated vehicles traveling within those segments. Although RL policies typically lack safety guarantees, our RL controller ensures safety by issuing headway commands above the minimum safe threshold to a safety-certified commercial ACC system, which consistently maintains a safe following distance. Notably, no crashes occurred in our experiments.

MDP To apply RL to our traffic control problem, we model the problem as a discrete-time, finite-horizon Markov Decision Process (MDP) (Puterman 2014), defined by a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, \rho_0, T)$, where $\mathcal{S}$ is the set of possible environment states, $\mathcal{A}$ is the set of all possible actions, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0,1]$ is a state transition probability distribution, $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ is a reward function mapping a given state and the action taken from it to a numeric reward, $\rho_0 : \mathcal{S} \rightarrow [0,1]$ is a distribution over initial states, and T is the problem's time horizon. In an MDP, the goal of the RL algorithm is to learn a decision-making policy $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0,1]$ that stochastically maps states to actions and maximizes the expected cumulative sum of rewards over all trajectories $E_\tau \sum_{t=0}^{T} r(s_t, a_t)$. Here τ is a trajectory $[s_0, a_0, s_1, a_1, ..., s_T, a_T]$, where the initial state s_0 is sampled from the initial state distribution: $S_0 \sim \rho_0$, actions are sampled from the policy: $a_t \sim \pi\left(s_t\right)$, and the next state in the trajectory is sampled from the transition probability $s_{t+1} \sim P\left(s_t, a_t\right)$, defined by the traffic simulator.

Modeling Traffic Control as an MDP We model our traffic control problem as an MDP by defining its states, actions, reward function, and horizon. The initial state distribution and the transition function are determined by the simulator.

States To enable effective traffic density control, our state representation encapsulates relevant information that (i) is necessary for predicting traffic dynamics over time and (ii) can be feasibly obtained with current technology. Specifically, the state includes average speeds and densities across 21 road segments before and after the traffic bottleneck, with each segment spanning approximately 100 meters. In general, both speed and density are essential in the state representation to prevent ambiguity.

Actions The actions correspond to the required time-headway values for automated vehicles in each segment. Since density tends to accumulate in segments preceding the traffic bottleneck, adjusting density in these segments offers the greatest potential for improving traffic efficiency through density control. Thus, our action is a vector of real numbers representing desired time-headways in each controlled segment before the bottleneck. Our experiments tested setups with 2–5 controlled segments, with 2 segments enabling faster RL convergence without performance loss. Consequently, the reported empirical results use this setup. While it is possible to explore more granular action spaces such as separate headways for each lane within a segment, they are less practical for real-world implementation as they require vehicle lane-position estimation.

Reward Function The reward function plays a crucial role in the RL training process, as it guides the agent toward maximizing the desired performance metric. As described in Section 4.1, our performance metric is the relative increase in average speed compared to a baseline of simulated human-driven traffic. However, the average speed of a vehicle can only be computed once a vehicle had completed its route, so using it would result in *delayed reward* which poses challenges for current RL algorithms.

To provide a more immediate reward, we use a time-delay reward function that is measured relative to free-flow traffic conditions. This reward is closely correlated to the performance metric when the traveled distance is fixed (see Appendix D), as is the case for the vast majority of vehicles in the simulation. The reward at time t is:

$$r_t = \frac{1}{C} \sum_{i=1}^{N_t} \left(\frac{v_t^{(i)} - v_{free}\left(x_t^{(i)}\right)}{v_{free}\left(x_t^{(i)}\right)} \right) \Delta t, \qquad (3)$$

where N_t is the number of vehicles planned to enter the simulation by time t, $x_t^{(i)}$ and $v_t^{(i)}$ are the location and velocity of vehicle i at time t, $v_{free}\left(x\right)$ is the speed limit of the road at location x, and C is a normalization factor that scales episode returns to the interval $[0, 1]$, to avoid numerical issues when using neural networks. For delayed vehicles which did not yet enter the simulation, the speed is assumed to be 0. When this reward is accumulated for all simulation time-steps, it provides an approximation for the average time delay for all vehicle trajectories, relative to free flow. The advantage of this reward is its immediate feedback on the impact of the current traffic state over the overall average speed.

Horizon The horizon is scenario dependent. We discuss the horizon length determination in Section 5.

Training Setup To solve the traffic control problem modeled as an MDP, we utilize the Proximal Policy Optimization (PPO) algorithm (Schulman et al. 2017). PPO is chosen for being well-suited for complex control tasks in continuous action spaces and for its training stability, but other state-of-the-art continuous RL algorithms could work similarly well. We use RLlib's PPO implementation (Duan et al. 2016) with most of its default hyperparameters, including a dual-head neural network representing both the policy and value functions with two hidden layers of 256 units each and $\tanh$ activation functions, and a linear output layer representing a diagonal Gaussian with mean and standard deviation for each controlled segment. We use a batch size of 2000, surrogate-, value-function-, and KL-losses, discount factor $\gamma = 0.99$ reflecting an effective horizon of about 100 steps, actions that are bounded to 1.5-6 second headway to reflect realistic values, and rewards that are normalized such that the value function's magnitude lies in a range that can be processed by a neural network without numerical issues. The training process is carried out over 25000 episodes, with each episode representing a finite-horizon simulation of traffic flow in our environment. Overall, the PPO algorithm, combined with our carefully designed training setup, enables the development of a policy that dynamically adjusts time-headways to optimize traffic flow while maintaining safety.

5 Empirical Analysis

In this section, we describe the experimental setup for evaluating the proposed time-headway control strategies and present results from hundreds of large-scale simulations.

Experiments were conducted in the SUMO simulation environment on a realistic 2 km, four-lane highway with a merging road (described in Section 3.2). The highway was divided into 20 segments of approximately 100 meters each, plus an additional segment for the merging road. This segment length balances effective sensing and control resolution while ensuring stable control policies.

The default time-headway is 1.5 seconds, and the highway speed limit is 31.29 m/s (70 mph). Traffic inflow matches the highway's maximum capacity of 1800 vehicles/hr/lane. Vehicles from the merging road enter at maximum inflow after a 200-second warm-up, allowing the simulation to reach a steady state. Merging traffic continues for 30 seconds in single-lane scenarios and 50 seconds in multi-lane scenarios, creating realistic traffic disturbances that impact flow.

Simulations ran for 500 seconds to allow congestion to develop and dissipate. An action interval of 2.5 seconds was chosen to reflect a realistic time frame for influencing traffic dynamics, considering vehicle response times and acceleration characteristics.

For each scenario we tested the performance of the following traffic configurations:

1. **(Baseline) 100% Human-Driven Traffic**
2. **(Baseline) Mixed Traffic with Fixed-Value Time-Headway Control:** Traffic consists of both human-driven vehicles and 20-100% CAVs, which follow headway commands from the fixed-value time-headway controller (described in Section 4.4). The controller was tuned to use the best-performing fixed time-headway.
3. **(Ours) Mixed Traffic with RL-based Controller:** Traffic consists of both human-driven vehicles and 20-100% CAVs, which follow headway commands from the RL-based time-headway controller (described in Section 4.5).

The two baselines were selected for the following reasons. First, human-driven traffic represents the current status quo, serving as a benchmark for improvement. Previous congestion reduction methods have generally not tackled realistic multi-lane highway scenarios involving lane changes and merges, likely due to the limitations discussed in Section 4.2. In our evaluations, these methods underperformed compared to human-driven traffic and were therefore omitted. Second, the manually tuned baseline is included to demonstrate the necessity of a more sophisticated RL-tuned controller.

Notably, an ablation analysis would be less informative for our system, as it relies on essential components whose removal would compromise functionality. Specifically, speed and density are necessary in the state representation to avoid ambiguity, and the reward function is directly derived from the performance metric to ensure alignment with our objectives. Additionally, sensitivity analysis indicated that the system is robust to minor changes in the number and length of road segments, as well as in action ranges.

Figure 3 presents results from hundreds of experiments under varying percentages of controlled vehicles. We used maximal vehicle inflow, which is the most sensitive to traffic disturbances. Each scenario configuration was run 30 times with different random seeds, enabling the computation of 95% confidence intervals. We focused on two scenario types: a simplified merge into a single-lane road and a complex merge into a multi-lane highway.

Single-Lane Scenarios Highway vehicles travel on a single-lane road with another single-lane road merging into it. This simplified setup tests time-headway control feasibility. The tuned fixed-value control improves merge efficiency and traffic flow over the human baseline across different CAV fractions (Figure 3a). RL-based control further enhances performance for 40% CAV fractions and above, achieving up to a 13% average speed increase at 100% CAV fraction.

Multi-Lane Scenarios Highway vehicles travel on a four-lane road, with a single-lane road merging into it, creating a congestion-prone bottleneck. Time-headway control signals are sent to all lanes in controlled segments. While lane-specific control could be more effective, it requires extra sensing, limiting practicality. Fixed-value time-headway control improves merge efficiency and traffic flow only at low or 100% CAV fractions (Figure 3b). In contrast, our RL-based controller outperforms both the human baseline and fixed-value control across all CAV fractions, achieving up to a 7% increase in average speed at 100% CAV fraction.

Overall, our RL-based controller outperforms the fixed-headway controller in scenarios with complex dynamics: specifically, in single-lane settings with CAV fractions of 40% or more, and in multi-lane settings with CAV fractions of 20% or more, where fixed-value headway controllers struggle to represent effective control strategies. These experiments demonstrate the effectiveness of our approach, which is the first to handle realistic multilane scenarios. Notably, the RL controller is most effective at high CAV fractions. With the widespread adoption of technologies such as ACC, it could become a viable tool for mitigating highway congestion.

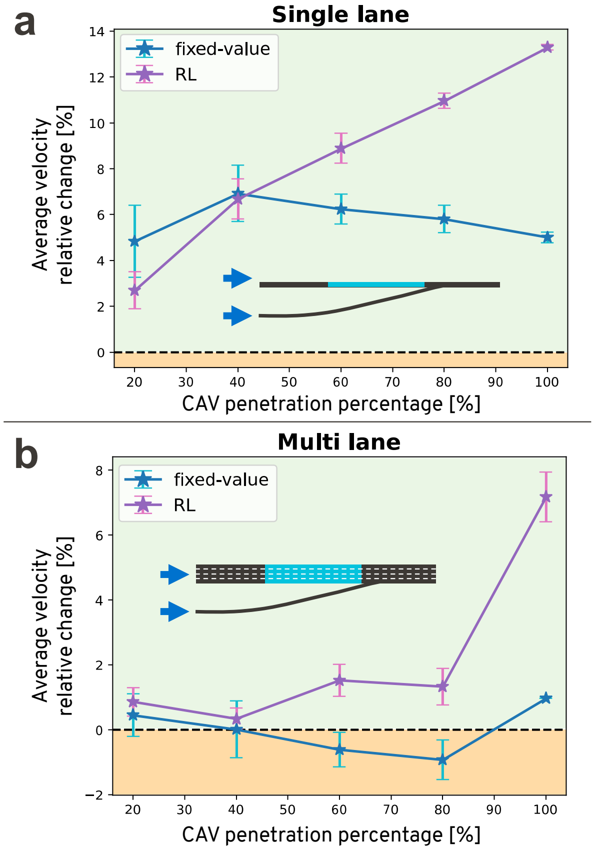

Figure 3: **Time-headway control performance**: Simulation performance for (a) single-lane, and (b) multi-lane scenarios. Performance of our safe RL-based controller and fixed-valued time-headway control baseline is measured relative to simulated human-driven traffic (dashed black line). Error bars show 95% confidence intervals for mean performance values, each derived from 30 simulations.

6 Conclusions

This paper proposes a dynamic time-headway control approach for increasing the average travel speed of vehicles while maintaining safety in high-volume highway traffic. By integrating with existing traffic estimation technology and low-bandwidth vehicle-to-infrastructure connectivity, and leveraging safety-certified adaptive cruise control systems, our method offers a practical path towards real-world implementation. Our safe reinforcement learning-based time-headway controller outperforms both baselines and alternative approaches across a variety of automated vehicle penetration rates, in both single- and multi-lane realistic simulated scenarios featuring hundreds of vehicles. Notably, even at low penetration rates, adjusting time-headways led to measurable improvements in average traffic speeds.

While these results are encouraging, several avenues for future work remain. First, deploying an RL controller trained in simulation into the real world requires it to be robust to diverse traffic flows and driving styles, and trained on simulations calibrated with real-world data. Additionally, real-world testing is crucial to validate the simulation results and overcome practical implementation challenges. Finally, integrating this approach with other traffic management strategies, such as ramp metering or dynamic lane assignment, could potentially yield even greater efficiency gains. As automated vehicle technology continues to advance, time-headway control emerges as a promising tool for transportation engineers and policymakers seeking to alleviate congestion and improve mobility in our road networks.

References

Alasiri, F.; Zhang, Y.; and Ioannou, P. A. 2023. Per-lane variable speed limit and lane change control for congestion management. *IEEE Transactions on Intelligent Transportation Systems*.

Bandō, M.; Hasebe, K.; Nakayama, A.; Shibata, A.; and Sugiyama, Y. 1995. Dynamical model of traffic congestion and numerical simulation. *Physical review. E*, 51 2: 1035–1042.

Bayen, A.; Delle Monache, M. L.; Garavello, M.; Goatin, P.; and Piccoli, B. 2022. *Control problems for conservation laws with traffic applications: modeling, analysis, and numerical methods*. Springer Nature.

Bayen, A. M.; Lee, J. W.; Piccoli, B.; Seibold, B.; Sprinkle, J. M.; and Work, D. B. 2020. CIRCLES: Congestion Impacts Reduction via CAV-in-the-loop Lagrangian Energy Smoothing. Presented at the 2020 Vehicle Technologies Office Annual Merit Review, Washington, DC (virtual).

CIRCLES. 2020. circles-consortium.github.io.

Cui, J.; Macke, W.; Yedidsion, H.; Goyal, A.; Urieli, D.; and Stone, P. 2021. Scalable Multiagent Driving Policies For Reducing Traffic Congestion. In *Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Delle Monache, M. L.; Liard, T.; Rat, A.; Stern, R.; Bhadani, R.; Seibold, B.; Sprinkle, J.; Work, D. B.; and Piccoli, B. 2019. *Feedback Control Algorithms for the Dissipation of Traffic Waves with Autonomous Vehicles*, 275–299. Springer International Publishing. ISBN 978-3-030-25446-9.

Duan, Y.; Chen, X.; Houthooft, R.; Schulman, J.; and Abbeel, P. 2016. Benchmarking Deep Reinforcement Learning for Continuous Control. In Balcan, M. F.; and Weinberger, K. Q., eds., *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*. New York, USA: PMLR.

Fattah, M. A.; Morshed, S. R.; and Kafy, A.-A. 2022. Insights into the socio-economic impacts of traffic congestion in the port and industrial areas of Chittagong city, Bangladesh. *Transportation Engineering*, 9: 100122.

Ferrara, A.; Sacone, S.; Siri, S.; et al. 2018. *Freeway traffic modelling and control*, volume 585. Springer.

Gao, Y.; and Levinson, D. 2023. Lane changing and congestion are mutually reinforcing. *Communications in Transportation Research*, 3.

Hall, F. L. 1996. Traffic stream characteristics. *US Federal Highway Administration*, 36.

Hegyi, A.; De Schutter, B.; and Hellendoorn, H. 2005. Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C*, 13(3): 185–209.

Hua, C.; and Fan, W. D. 2023. Dynamic speed harmonization for mixed traffic flow on the freeway using deep reinforcement learning. *IET Intelligent Transport Systems*, 17(12): 2519–2530.

Kerner, B. S. 2016. Failure of classical traffic flow theories: Stochastic highway capacity and automatic driving. *Physica A: Statistical Mechanics and its Applications*, 450: 700–747.

Krajzewicz, D.; Erdmann, J.; Behrisch, M.; and Bieker, L. 2012. Recent development and applications of SUMO-Simulation of Urban MObility. *International journal on advances in systems and measurements*, 5(3&4).

Kreidieh, A. R.; Wu, C.; and Bayen, A. M. 2018. Dissipating stop-and-go waves in closed and open networks via deep reinforcement learning. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 1475–1480.

Lee, J. W.; Wang, H.; Jang, K.; Hayat, A.; Bunting, M.; Alanqary, A.; Barbour, W.; Fu, Z.; Gong, X.; Gunter, G.; Hornstein, S.; Kreidieh, A. R.; Lichtlé, N.; Nice, M. W.; Richardson, W. A.; Shah, A.; Vinitsky, E.; Wu, F.; Xiang, S.; Almatrudi, S.; Althukair, F.; Bhadani, R.; Carpio, J.; Chekroun, R.; Cheng, E.; Chiri, M. T.; Chou, F.-C.; Delorenzo, R.; Gibson, M.; Gloudemans, D.; Gollakota, A.; Ji, J.; Keimer, A.; Khoudari, N.; Mahmood, M.; Mahmood, M.; Matin, H. N. Z.; Mcquade, S.; Ramadan, R.; Urieli, D.; Wang, X.; Wang, Y.; Xu, R.; Yao, M.; You, Y.; Zachár, G.; Zhao, Y.; Ameli, M.; Baig, M. N.; Bhaskaran, S.; Butts, K.; Gowda, M.; Janssen, C.; Lee, J.; Pedersen, L.; Wagner, R.; Zhang, Z.; Zhou, C.; Work, D. B.; Seibold, B.; Sprinkle, J.; Piccoli, B.; Monache, M. L. D.; and Bayen, A. M. 2024. Traffic Control via Connected and Automated Vehicles: An Open-Road Field Experiment with 100 CAVs. arXiv:2402.17043.

Lomax, T.; Schrank, D.; and Eisele, B. 2021. 2021 Urban Mobility Report. https://mobility.tamu.edu/umr/.

Lu, X.-Y.; and Shladover, S. E. 2014. Review of Variable Speed Limits and Advisories: Theory, Algorithms, and Practice. *Transportation Research Record*, 2423(1): 15–23.

Mohammadian, S.; Zheng, Z.; Haque, M. M.; and Bhaskar, A. 2021. Performance of continuum models for realworld traffic flows: Comprehensive benchmarking. *Transportation Research Part B: Methodological*, 147: 132–167.

Ni, D.; and Leonard II, J. D. 2006. Direct methods of determining traffic stream characteristics by definition. Technical report, University of Massachusetts Amherst.

OpenStreetMap contributors. 2017. https://www.openstreetmap.org.

Pang, M.; and Huang, J. 2022. Cooperative Control of Highway On-Ramp with Connected and Automated Vehicles as Platoons Based on Improved Variable Time Headway. *Journal of Transportation Engineering, Part A: Systems*, 148(7): 04022034.

Puterman, M. L. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Samaei, M.; Ameli, M.; Davis, J. F.; Mcquade, S. T.; Lee, J.; Piccoli, B.; and Bayen, A. M. 2023. Bi-Level Optimization Model for DTA Flow and Speed Calibration. In *The 9th International Symposium on Dynamic Traffic Assignment (DTA2023)*. Chicago, United States.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347.

Stern, R. E.; Cui, S.; Delle Monache, M. L.; Bhadani, R.; Bunting, M.; Churchill, M.; Hamilton, N.; Haulcy, R.; Pohlmann, H.; Wu, F.; Piccoli, B.; Seibold, B.; Sprinkle, J.; and Work, D. B. 2018. Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments. *Transportation Research Part C*, 89: 205–221.

Sugiyama, Y.; Fukui, M.; Kikuchi, M.; Hasebe, K.; Nakayama, A.; Nishinari, K.; ichi Tadaki, S.; and Yukawa, S. 2008. Traffic jams without bottlenecks—experimental evidence for the physical mechanism of the formation of a jam. *New Journal of Physics*, 10(3): 033001.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book. ISBN 0262039249.

Towers, M.; Kwiatkowski, A.; Terry, J.; Balis, J. U.; Cola, G. D.; Deleu, T.; Goulão, M.; Kallinteris, A.; Krimmel, M.; KG, A.; Perez-Vicente, R.; Pierré, A.; Schulhoff, S.; Tai, J. J.; Tan, H.; and Younis, O. G. 2024. Gymnasium: A Standard Interface for Reinforcement Learning Environments. arXiv:2407.17032.

Treiber, M.; Hennecke, A.; and Helbing, D. 2000. Congested Traffic States in Empirical Observations and Microscopic Simulations. *Physical Review E*, 62: 1805–1824.

Vinitsky, E.; Lichtle, N.; Parvate, K.; and Bayen, A. 2023. Optimizing mixed autonomy traffic flow with decentralized autonomous vehicles and multi-agent RL. *ACM Transactions on Cyber-Physical Systems*.

Vinitsky, E.; Parvate, K.; Kreidieh, A.; Wu, C.; and Bayen, A. 2018. Lagrangian Control through Deep-RL: Applications to Bottleneck Decongestion. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*.

Wang, H.; Fu, Z.; Lee, J.; Matin, H. N. Z.; Alanqary, A.; Urieli, D.; Hornstein, S.; Kreidieh, A. R.; Chekroun, R.; Barbour, W.; et al. 2024a. Hierarchical speed planner for automated vehicles: A framework for lagrangian variable speed limit in mixed autonomy traffic. *arXiv preprint arXiv:2402.16993*.

Wang, J.; Zheng, Y.; Dong, J.; Chen, C.; Cai, M.; Li, K.; and Xu, Q. 2023a. Implementation and experimental validation of data-driven predictive control for dissipating stop-and-go waves in mixed traffic. *IEEE Internet of Things Journal*.

Wang, S.; Shang, M.; Levin, M. W.; and Stern, R. 2023b. A general approach to smoothing nonlinear mixed traffic via control of autonomous vehicles. *Transportation Research Part C*, 146: 103967.

Wang, S.; Wang, Z.; Jiang, R.; Zhu, F.; Yan, R.; and Shang, Y. 2024b. A multi-agent reinforcement learning-based longitudinal and lateral control of CAVs to improve traffic efficiency in a mandatory lane change scenario. *Transportation Research Part C*, 158: 104445.

Wu, C.; Bayen, A. M.; and Mehta, A. 2018. Stabilizing Traffic with Autonomous Vehicles. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 1–7. IEEE Press.

Yanakiev, D.; and Kanellakopoulos, I. 1995. Variable time headway for string stability of automated heavy-duty vehicles. In *Proceedings of 1995 34th IEEE Conference on Decision and Control*, volume 4, 4077–4081 vol.4.

Yang, M.; Wang, X.; and Quddus, M. 2019. Examining lane change gap acceptance, duration and impact using naturalistic driving data. *Transportation Research Part C*, 104: 317–331.

Zhang, Y.; Macke, W.; Cui, J.; Hornstein, S.; Urieli, D.; and Stone, P. 2023a. Learning a robust multiagent driving policy for traffic congestion reduction. *Neural Computing and Applications*, 1–14.

Zhang, Y.; Quiñones-Grueiro, M.; Barbour, W.; Zhang, Z.; Scherer, J.; Biswas, G.; and Work, D. B. 2023b. Cooperative Multi-Agent Reinforcement Learning for Large Scale Variable Speed Limit Control. *2023 IEEE International Conference on Smart Computing (SMARTCOMP)*, 149–156.

A Simulation Details

This section describes the details of simulations shown in various figures in the paper.

Figure 1b, lane-change behavior tuning This figure presents three examples of time-space diagrams, illustrating the throughput along the road during the simulations. These simulations were conducted in a multi-lane scenario over a 500-second period, with a 50-second merge starting after 200 seconds and an inflow rate of 1800 vehicles per hour per lane. In the aggressive example (left), the lcAssertive parameter was set to 5, while in the tuned example (right), it was set to 3. In the no-lane-change scenario (middle), the lcSpeedGain and lcStrategic parameters were set to 0, and the lcCooperative parameter was disabled by setting it to -1.

Figure 2c, comparison of constant speed and constant time-headway control This figure compares two simulations: one with fixed-value speed limit control and one with fixed-value time-headway control. The results are presented in two pairs of time-space diagrams. The top pair illustrates the throughput along different road segments over time, while the bottom pair shows the average velocity along the road. The simulated scenario involves a single-lane highway without any merging road inflow. After 200 seconds, control signals are applied to all vehicles within a 200-meter road segment for a duration of 100 seconds.

B Computing Infrastructure

Hardware Desktop with 12 Intel Xeon W-2133 3.6GHz CPU cores, 64 GB RAM.

Operating system Ubuntu 20.04.

Software To recreate our software environment, install Eclipse SUMO 1.17 simulator, and create a Conda environment using the file environment.yml in our code repository.

C Hyperparameters

This section describes the parameters and hyperparameters used for the vehicle behavior in the SUMO simulator, for the MDP actions and rewards, and RLlib's PPO algorithm, listed in Table 1.

SUMO parameters The step length was chosen to be 0.5 seconds to allow more resolution than the default 1 second step, while keeping simulation time low. Most IDM car following parameters were kept default. The default time-headway parameter (tau) value was empirically tuned to 1.5 seconds, to obtain a maximal incoming vehicle flow of 1800 vehicles per hour per lane. The continuous sub-lane lane-change model was used to capture the complex dynamics of lane changes better than the default model. Lane-change behavior parameters were empirically tuned to mitigate spurious disturbances once all vehicles from the merging road had entered the highway. The empirical tests were analyzed using a time-space diagram of the simulation, as illustrated in Figure 2 of the paper. Additionally, the lane change parameter that required vehicles to drive in the rightmost lane was disabled. This parameter caused vehicles to move to the rightmost lane whenever there were sufficiently large gaps, even if the vehicles in that lane were moving slower, which is not realistic behavior.

MDP parametes Reward normalization was selected such that returns will be approximately in the range [-1, 0]. For safety reasons, the minimum action value of 1.5 seconds was chosen to be equal to the default time headway parameter of the IDM car following model. This guarantees that the RL agent can only increase time-headway. The maximum action value was selected to narrow the action space, thereby reducing training time while still accommodating a sufficiently large time-headway range. We tested different values for the numbers of control segments, ranging from 2 to 5. We chose to use 2 since the performance of all values was similar, while training with 2 control segments was faster.

PPO hyperparameters The number of rollout workers was chosen based on the number of available CPU cores, and the train batch size was adjusted accordingly to include one full simulation episode for each rollout worker. Other PPO hyperparameter values were kept as RLlib's defaults, since they resulted in both speed and stability in the learning process.

SUMO parameters	
step-length	0.5 seconds
lateral-resolution	0.4 meters
extrapolate-departpos	True
tau (default time-headway)	1.5 seconds
lcKeepRight	0
lcAssertive	3
lcSpeedGain	5
MDP parameters	
$\frac{1}{C}$ (reward normalization)	10^{-5}
action range	[1.5, 6] seconds
num_control_segments	2
RLlib PPO parameters	
num_rollout_workers	10
train_batch_size	2000
sgd_minibatch_size	128
clip_param	0.3
num_sgd_iter	30
use_gae	True
lambda	1
vf_loss_coeff	1
kl_coeff	0.2
entropy_coeff	0
learning_rate	$5 \cdot 10^{-5}$

Table 1: SUMO, MDP, and PPO hyperparameters

D Reward Function Analysis

The average velocity of vehicle i is

$$\bar{v}_i = \frac{L_i}{T_i}, \tag{4}$$

where T_i and L_i are the total travel time and the total travel distance of vehicle i, respectively. Thus, assuming the travel distance (route) of each of the vehicles is fixed, increasing the average speed amounts to minimizing T_i.

For a vehicle i in the simulation, the total travel time can be computed by:

$$T_i = \sum_{t=1}^{T_{sim}} a_i\left(t\right) \mathrm{d}t, \tag{5}$$

where T_{sim} is the total number of simulation timesteps, $\mathrm{d}t$ is the duration of each timestep, and $a_i\left(t\right)$ is a boolean parameter which is 1 if vehicle i was planned to enter the simulation before time t and it did not yet exit the simulation before time t, and 0 otherwise. This boolean parameter is crucial, since without it the controller can just prevent vehicles from entering the simulation, and thus decrease the road density and increase average velocity for a smaller number of vehicles without penalty. Note, that since we measure L_i only on the travel distance within the simulated road, we assume here that the velocity of delayed vehicles is 0. While this is not accurate, this still provides a better approximation for the average speed of the full road than disregarding delayed vehicles completely. This will also provide higher penalties on delaying vehicles from entering the simulation.

Therefore, the function to minimize is the average total travel time of vehicles in the simulation:

$$\bar{T} = \frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T_{sim}} a_i\left(t\right) \mathrm{d}t, \tag{6}$$

where N is the number of vehicles that are expected to enter the simulation.

Reinforcement learning algorithms train a controller to maximize the expected value of the return, which is the timeaggregation of the (discounted) reward:

$$\max_{\pi} \lim_{T_{sim} \to \infty} E \left\{ \sum_{t=1}^{T_{sim}} \gamma^t r\left(t\right) \right\}, \tag{7}$$

where π is the policy, and $\gamma < 1$ is the discount factor. For this theoretical analysis, we assume $\gamma = 1$, and note that by replacing the return in Equation 7 with the (negative) average travel time in Equation 6, we derive the following reward function:

$$r(t) = -\frac{1}{N} \sum_{i=1}^{N} a_i\left(t\right) \mathrm{d}t. \tag{8}$$

However, using the total travel time reward has a few drawbacks:

1. Delayed reward – The reward for an action is delayed since it only depends on the number of vehicles exiting the simulation. Therefore, only when a vehicle leaves the simulation, the penalty is reduced. In this case, distinguishing between actions that decrease travel time and actions that increase it is a challenge, since the total travel time depends on the entire time series of actions for each vehicle.

2. Start / end point dependent – Positive reward is achieved only for vehicles that exited the simulation. This means that if we increase the number of simulated road segments, the reward will be further delayed.

To deal with that, we seggest a time-delay reward function:

$$r(t) = -\frac{1}{N} \sum_{i=1}^{N} a_i\left(t\right) \left(1 - \frac{v_i\left(t\right)}{v_{free}\left(x_i\left(t\right)\right)}\right) \mathrm{d}t, \tag{9}$$

where $v_i\left(t\right)$ and $x_i\left(t\right)$ are the velocity and position of vehicle i at time t respectively, and v_{free} is the free-flow velocity at a certain location (i.e., the speed limit there). This is an immediate reward that uses real-time velocity readings of the vehicles. Therefore, actions get immediate feedback based on the magnitude of the travel delay they caused.

The sum of these rewards over the entire simulation duration gives the average time delay measured from the free flow completion time:

$$\begin{aligned}
\sum_{t=1}^{T} r\left(t\right) &= -\frac{1}{N} \sum_{t=1}^{T_{sim}} \sum_{i=1}^{N} a_i\left(t\right) \left(1 - \frac{v_i\left(t\right)}{v_{free}\left(x_i\left(t\right)\right)}\right) \mathrm{d}t \\
&= -\frac{1}{N} \sum_{t=1}^{T_{sim}} \sum_{i=1}^{N} a_i\left(t\right) \left(1 - \frac{v_i\left(t\right) \mathrm{d}t}{v_{free}\left(x_i\left(t\right)\right)}\right) \\
&= -\frac{1}{N} \sum_{t=1}^{T_{sim}} \sum_{i=1}^{N} a_i\left(t\right) \left(1 - \frac{\mathrm{d}x_i\left(t\right)}{v_{free}\left(x_i\left(t\right)\right)}\right) \\
&= -\frac{1}{N} \sum_{t=1}^{T_{sim}} \sum_{i=1}^{N} a_i\left(t\right) \left(1 - \mathrm{d}t_{free}^{(i)}\left(t\right)\right) \\
&= -\frac{1}{N} \sum_{i=1}^{N} \left(T_i - T_{free}^{(i)}\right) \\
&= -\frac{1}{N} \sum_{i=1}^{N} \Delta T_i,
\end{aligned} \tag{10}$$

where $\mathrm{d}x_i\left(t\right)$ is the distance vehicle i traveled during timestep t, $T_{free}^{(i)}$ is the travel time of vehicle i assuming it drove at free-flow speed, and ΔT_i is its time delay. $\mathrm{d}t_{free}^{(i)}\left(t\right)$ is the time it would take vehicle i to travel a distance of $\mathrm{d}x_i\left(t\right)$ in free-flow. The last steps in the derivation above assume vehicle i completed its entire journey. Since each vehicle's travel distance is its route length

$$\sum_{t=1}^{T_{sim}} a_i\left(t\right) \mathrm{d}x_i\left(t\right) = L_i, \tag{11}$$

the free flow completion time of the entire route is

$$T_{free}^{(i)} = \sum_{t=1}^{T_{sim}} a_i\left(t\right) \frac{\mathrm{d}x_i\left(t\right)}{v_{free}\left(x_i\left(t\right)\right)} = \sum_{t=1}^{T_{sim}} a_i\left(t\right) \mathrm{d}t_{free}^{(i)}(t) \tag{12}$$

which is constant for all vehicles with the same route. This computation further assumes that free flow velocity is constant between $x_i\left(t\right)$ and $x_i\left(t\right) + \mathrm{d}x_i\left(t\right)$ for all values of t.

It is important to note that as long as vehicle i completes its entire route, maximizing the expected value of the above objective leads to maximizing the expected value of the (weighted) relative change in average velocity compared to a baseline simulation:

$$
\begin{aligned}
&E\left\{-\frac{1}{N}\sum_{i=1}^{N}\left(T_i - T_{free}^{(i)}\right)\right\} \\
=&E\left\{-\frac{1}{N}\sum_{i=1}^{N}\left(T_i - T_{base}^{(i)} + T_{base}^{(i)} - T_{free}^{(i)}\right)\right\} \\
=&E\left\{-\frac{1}{N}\sum_{i=1}^{N}\left(T_i - T_{base}^{(i)}\right) \underbrace{-\frac{1}{N}\sum_{i=1}^{N}\left(T_{base}^{(i)} - T_{free}^{(i)}\right)}_{\text{constant}}\right\} \\
\propto&E\left\{-\frac{1}{N}\sum_{i=1}^{N}\left(T_i - T_{base}^{(i)}\right)\right\} \\
=&E\left\{-\frac{1}{N}\sum_{i=1}^{N}T_i\left(1 - \frac{T_{base}^{(i)}}{T_i}\right)\right\} \\
=&E\left\{-\frac{1}{N}\sum_{i=1}^{N}T_i\left(1 - \frac{\bar{v}_i}{\bar{v}_{base}^{(i)}}\right)\right\} \\
=&E\left\{\frac{1}{N}\sum_{i=1}^{N}T_i\left(\frac{\bar{v}_i - \bar{v}_{base}^{(i)}}{\bar{v}_{base}^{(i)}}\right)\right\},
\end{aligned}
\tag{13}
$$

where $\bar{v}_{base}^{(i)}$ and $T_{base}^{(i)}$ are the average velocity of vehicle i, and its total travel time in the baseline (human-only) simulation, respectively. This is similar to the metric used in the paper, up to a weight factor for each vehicle which is equal to the total travel time for that vehicle. This means that the reward function puts more weight on vehicles whose total travel time is large. However, since the total travel time of the vehicles depends on the policy, it cannot be easily normalized in the reward function.