

Stabilizing Mixed Traffic with Autonomous Vehicles: A Multi-Lane Mean Field Game and Reinforcement Learning Approach

Yitian Liang^{1*}

¹Department of Applied Physics and Applied Mathematics, Columbia University, New York, 10027, NY, USA.

Corresponding author(s). E-mail(s): yl5433@columbia.edu;

Abstract

Autonomous vehicles (AVs) hold promise for reducing congestion and improving safety in mixed traffic flows where human-driven vehicles (HVs) remain dominant. This paper presents a multi-lane framework that couples the Aw–Rascle–Zhang (ARZ) model for HVs with a Mean Field Game (MFG) formulation for AVs. We also explore a reinforcement learning (RL) alternative to the MFG approach, treating the partial differential equation (PDE) environment as a Markov decision process. Our experiments cover three scenarios: (1) a multi-lane ring road to study fundamental wave suppression, (2) a highway segment with an on-ramp merging problem, and (3) a multi-agent RL setup. In all cases, we show that a moderate or high AV penetration rate can significantly reduce stop-and-go waves, merging bottlenecks, and collision likelihood. Our findings highlight the role of AV-based controls in stabilizing traffic under various inflow conditions and illustrate how RL can serve as a flexible method for large-scale control. These results provide insights for future AV integration strategies in real-world transportation networks.

Keywords: Autonomous vehicles, Mean Field Game, Aw–Rascle–Zhang model, Multi-lane traffic flow, Reinforcement learning, Traffic stability

1 Introduction

Autonomous vehicles (AVs) are expected to transform current transportation systems by reducing congestion, enhancing road safety, and improving mobility. Yet, many

open questions remain about how AVs interact with human-driven vehicles (HVs), particularly in mixed traffic flows under realistic demand conditions. A growing body of work has explored these interactions using macroscopic models, numerical simulations, and theoretical analyses. Inspired by [1] and [2], we focus on understanding the role of AVs in stabilizing traffic when both AVs and HVs share the road.

In purely HV-dominated traffic, stop-and-go waves often arise from small driver-induced errors or disturbances, eventually leading to severe congestion if the flow is dense. Studies such as [1] and [2] demonstrate that by modeling AVs as rational agents who adjust their speeds or headways to minimize costs (e.g., travel time, fuel use, or collisions), one can attenuate or eliminate these instability waves. Most existing models analyze such effects under either a ring-road setting (for fundamental wave propagation research) or a highway segment with merging on-ramps (to examine bottleneck formation). Despite these advances, more work is needed to clarify the broader conditions—such as AV penetration rates, lane-changing dynamics, and inflow intensities—that determine whether traffic flow remains stable or breaks down.

In this paper, we extend and unify these ideas by presenting three complementary scenarios, each highlighting a different aspect of mixed traffic flow with AVs and HVs. First, we investigate a multi-lane ring road to see how AV penetration and control laws affect wave suppression. Next, we consider a main highway segment with an on-ramp, where merging can create localized congestion if inflows exceed capacity. Finally, we introduce a reinforcement learning (RL) framework that treats the partial differential equation (PDE) environment as an interactive setting in which AV agents learn speed and lane strategies through trial and error.

Our key contributions include:

- Proposing a combined mean field game (MFG) and Aw–Rascle–Zhang (ARZ) framework that models both AV and HV sub-populations in a multi-lane configuration.
- Demonstrating through numerical experiments how AVs can significantly improve stability and reduce stop-and-go waves, especially at modest to high penetration levels.
- Exploring a reinforcement learning approach for AV control, offering a flexible alternative when the system’s dimensionality or complexity challenges a purely analytical MFG solution.

The remainder of this paper is structured as follows. We first describe our modeling approach, which couples ARZ-based HV equations with MFG or RL-based AV controls in multi-lane traffic. We then present numerical experiments that illustrate how AVs mitigate traffic instability in both ring-road and merging scenarios. Finally, we compare the PDE-MFG solution against a PDE-plus-RL approach, highlighting their respective advantages and limitations.

2 Model and Methodology

In this section, we propose an **extended multi-lane MFG-ARZ model** designed to capture the lane-changing and merging/diverging behaviors in a road network with

mixed human-driven vehicles (HV) and autonomous vehicles (AV). We detail the governing PDEs, define the terms corresponding to lane changes and on/off-ramp merges, and specify the Mean Field Game (MFG) cost function along with additional penalty/reward components that reflect safety, comfort, and efficiency constraints. We then introduce a reinforcement learning (RL) framework that interprets the PDE-based environment as an MDP, linking the MFG-ARZ cost structure to an RL reward.

2.1 Multi-Lane MFG-ARZ PDE Formulation

Let N be the number of lanes. For each lane $i = 1, 2, \dots, N$, we introduce:

- $\rho_i(x, t)$ as the density of vehicles in lane i .
- $u_i(x, t)$ as the average speed of vehicles in lane i .

In a mixed traffic flow scenario, one can further distinguish between ρ_i^{AV} , ρ_i^{HV} , u_i^{AV} , and u_i^{HV} . For brevity, we will first present the standard lane-based notation, then incorporate MFG aspects for AV.

2.1.1 Conservation of Mass with Lane-Changing

Each lane i satisfies a **continuity equation** with additional source/sink terms Γ_{ij} that model the flow of vehicles changing from lane i to lane j [3] [4] [5]:

$$\frac{\partial \rho_i}{\partial t} + \frac{\partial}{\partial x}(\rho_i u_i) = \sum_{j \neq i} (\Gamma_{ji} - \Gamma_{ij}),$$

where $\Gamma_{ij} = \Gamma_{ij}(\rho_i, \rho_j, u_i, u_j)$ represents the vehicle flow rate from lane i to lane j . If on/off-ramp merges are present, similar source terms can be added at specific spatial locations.

Note: $\Gamma_{ij} \geq 0$ effectively *decreases* ρ_i and *increases* ρ_j . The function Γ_{ij} typically depends on *relative speeds* (i.e., u_i vs. u_j), as well as densities ρ_i, ρ_j and driver/vehicle preferences.

2.1.2 Lane-Specific ARZ Momentum Equation (HV)

For *human-driven vehicles*, we adopt a lane-based Aw-Rascle-Zhang-type momentum equation [3, 4]. In lane i :

$$\frac{\partial}{\partial t}(u_i + h(\rho_i)) + u_i \frac{\partial}{\partial x}(u_i + h(\rho_i)) = -\frac{1}{\tau}(U(\rho_i) - u_i),$$

where:

- $h(\rho_i)$ is the lane-specific hesitation function (often non-decreasing in ρ_i),
- $U(\rho_i)$ is the desired speed function, typically *decreasing* in ρ_i ,
- τ is a relaxation time parameter controlling how quickly vehicles adjust their speed toward $U(\rho_i)$.

If we separate HV densities and speeds explicitly, let ρ_i^{HV} and u_i^{HV} denote the HV density and velocity in lane i . Then, in the *multi-lane* setting, the ARZ momentum

equation for HV becomes [6]:

$$\frac{\partial}{\partial t} \left(u_i^{HV} + h(\rho_i^{HV}) \right) + u_i^{HV} \frac{\partial}{\partial x} \left(u_i^{HV} + h(\rho_i^{HV}) \right) = -\frac{1}{\tau} \left(U(\rho_i^{HV}) - u_i^{HV} \right). \quad (2)$$

(If desired, one can replace ρ_i^{HV} by the total density $\rho_i^{TOT} = \rho_i^{HV} + \rho_i^{AV}$ inside $h(\cdot)$ or $U(\cdot)$.)

2.1.3 MFG Extension for Autonomous Vehicles (AV)

Now consider AV in lane i , with density ρ_i^{AV} and velocity u_i^{AV} . We model AV as *rational agents* aiming to minimize an individual cost (or maximize utility). In a continuum limit, following [7]:

1. **Continuity Equation** (for ρ_i^{AV}), with lane-changing terms Γ_{ij}^{AV} :

$$\frac{\partial \rho_i^{AV}}{\partial t} + \frac{\partial}{\partial x} (\rho_i^{AV} u_i^{AV}) = \sum_{j \neq i} (\Gamma_{ji}^{AV} - \Gamma_{ij}^{AV}). \quad (3)$$

2. **HJB Equation:** Each AV seeks to minimize a cost functional that depends on speed, density, and possibly lane-changing penalties. The value function $V_i(x, t)$ in lane i satisfies:

$$\frac{\partial V_i}{\partial t} + u_i^{AV} \frac{\partial V_i}{\partial x} + f(\rho_i, u_i^{AV}) + \Phi_{\text{lane}}(\rho_i) = 0,$$

where $f(\cdot)$ is a running cost for speed/acceleration, and Φ_{lane} is an additional penalty (or reward) term representing the lane-changing cost and congestion. The *optimal velocity* for AV is then:

$$u_i^{AV} = \arg \min_a \left\{ a \partial_x V_i + f(\rho_i, a) \right\}. \quad (5)$$

3. **Lane-Changing Cost:** To switch from lane i to lane j , an AV may incur a penalty $\Psi_{ij}(\rho_i, \rho_j)$. This can be integrated into the HJB framework or modeled as a jump process.

Putting these together, we obtain a *multi-lane MFG-ARZ system* of PDEs describing how AV and HV densities evolve, how vehicles choose lane changes, and how AV compute their optimal speed (and possibly lane) strategies in a mean-field sense [7].

2.2 MFG Cost Function and Additional Terms

We construct a **composite cost function** for AV that balances safety, comfort, and travel efficiency. A possible example is [7]:

$$f(\rho_i, u) = \alpha_1 \left(\frac{u}{u_{\max}} \right)^2 - \alpha_2 \left(\frac{u}{u_{\max}} \right) + \alpha_3 \left(\frac{\rho_i}{\rho_{\max}} \right) \left(\frac{u}{u_{\max}} \right),$$

where:

- $\alpha_1 > 0$ penalizes excessive speeds (energy usage),
- $\alpha_2 > 0$ rewards higher speeds for efficiency,
- $\alpha_3 > 0$ models a safety cost coupling speed with density.

Lane-specific penalty/reward $\Phi_{\text{lane}}(\rho_i)$ might reflect the congestion level in the lane (e.g., higher penalty for denser lanes) and encourage AV to choose less congested lanes. **Lane-changing cost** Ψ_{ij} further penalizes frequent or abrupt lane switches [5].

In a **Mean Field Game (MFG)** setting, we obtain a *coupled PDE system*: the HJB equation plus the continuity equation. This pair enforces that each AV's best-response speed collectively matches the evolving density distribution.

2.3 Lane-Changing and Merging Terms

For multi-lane traffic, Γ_{ij} or Γ_{ij}^{AV} is crucial. We adopt a generic form:

$$\Gamma_{ij}^{(\text{HV or AV})}(x, t) = \gamma_{ij} \Phi(\rho_i, \rho_j, u_i, u_j),$$

where γ_{ij} is a tunable rate constant and $\Phi(\cdot)$ is a function reflecting the *motivation* to move from lane i to lane j (e.g., higher if $u_j > u_i$, lower if ρ_j is large). If **on/off-ramps** exist, lane j could represent a ramp, and Γ_{ij} becomes an *entry/exit* flow. Merging from multiple lanes into one (e.g., from lane 1 and lane 2 merging into lane 3) can be viewed as a capacity-constrained inflow [5].

2.4 Complex Road Network Nodes and Boundary Conditions

Real-world highways, ramps, and intersections can be represented by a *directed graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each edge $e \in \mathcal{E}$ is a lane-segment (possibly multi-lane), and nodes $v \in \mathcal{V}$ are junctions. In the **multi-lane MFG-ARZ** context [3, 5]:

1. **Per-Edge PDEs:** Each edge e contains one or more lanes. We solve continuity and momentum (ARZ or MFG-ARZ) for HV/AV in each lane i , with appropriate boundary conditions at $x = 0$ and $x = L_e$.
2. **Node Coupling:** At a node v with incoming edges $\{e_1^-, \dots\}$ and outgoing edges $\{e_1^+, \dots\}$, we impose flow conservation:

$$\sum_{e^- \in \text{In}(v)} F_{e^-}(t) = \sum_{e^+ \in \text{Out}(v)} F_{e^+}(t),$$

plus capacity or supply–demand constraints. HV/AV splits can be handled similarly, ensuring that the total HV or AV inflow equals outflow at the node.

3. **Boundary Conditions:**

- **Inflow** boundaries (e.g., external ramp) might fix or cap ρ or flows F .
- **Outflow** boundaries can be “free” or capacity-limited.
- **Periodic** if the domain is a ring.

By combining *lane-specific PDE solutions* on each edge with node constraints, we obtain a *network-level multi-lane MFG-ARZ model* capturing merges, diverges, and more complex topologies.

2.5 Putting It All Together

Summarizing the final system of PDEs:

- **HV (ARZ) Continuity and Momentum** for lane i on edge e :

$$\begin{aligned} \rho_{e,i,t}^{HV} + (\rho_{e,i}^{HV} u_{e,i}^{HV})_x &= \sum_{j \neq i} (\Gamma_{e,ji}^{HV} - \Gamma_{e,ij}^{HV}), \\ (u_{e,i}^{HV} + h(\rho_{e,i}^{HV}))_t + u_{e,i}^{HV} (u_{e,i}^{HV} + h(\rho_{e,i}^{HV}))_x &= -\frac{1}{\tau} (U(\rho_{e,i}^{HV}) - u_{e,i}^{HV}). \end{aligned}$$

- **AV (MFG) Continuity and HJB** (lane i on edge e):

$$\rho_{e,i,t}^{AV} + (\rho_{e,i}^{AV} u_{e,i}^{AV})_x = \sum_{j \neq i} (\Gamma_{e,ji}^{AV} - \Gamma_{e,ij}^{AV}), \quad (C1)$$

$$V_{e,i,t} + u_{e,i}^{AV} V_{e,i,x} + f(\rho_{e,i}^{AV}, u_{e,i}^{AV}) = 0, \quad u_{e,i}^{AV} = \operatorname{argmin}_a \left\{ a V_{e,i,x} + f(\rho_{e,i}^{AV}, a) \right\}. \quad (HJB)$$

- **Lane-Changing / Merging:** $\Gamma_{e,ij}^*$ defined via a suitable model or node-based constraints.
- **Network Node Conservation:**

$$\sum_{e^- \in \text{In}(v)} F_{e^-}(t) = \sum_{e^+ \in \text{Out}(v)} F_{e^+}(t),$$

plus capacity constraints, etc.

This *coupled PDE system* captures:

- **Human Driving (HV):** the ARZ part for HV [3, 4, 6].
- **Optimal AV Driving:** the MFG part for AV [7].
- **Multi-Lane Interactions:** through lane-specific PDEs and cross-lane fluxes Γ_{ij} .
- **Network Complexity:** by imposing node conditions at merges, diverges, and interchanges [5].

2.6 RL/Deep Learning Framework with PDE-Based Cost

In this final subsection, we present a **reinforcement learning (RL)** approach to solving the *multi-lane MFG-ARZ* control problem for autonomous vehicles (AV). Instead of explicitly solving the Hamilton–Jacobi–Bellman (HJB) equation, we interpret the *traffic PDE environment* as a *Markov Decision Process (MDP)* where agents (AVs) learn policies (speeds, lane changes) to minimize a cost (or maximize reward).

Below, we show the *reward function* derivation, the *Bellman equations*, *policy gradient* methods, and a typical *implementation workflow*. Such an RL-based approach for high-dimensional PDEs is discussed in [5].

2.6.1 PDE-Derived Reward Function

Recall that in the MFG-ARZ framework, each AV aims to minimize a running cost:

$$f(\rho, u) = \alpha_1 \left(\frac{u}{u_{\max}} \right)^2 - \alpha_2 \left(\frac{u}{u_{\max}} \right) + \alpha_3 \left(\frac{\rho}{\rho_{\max}} \right) \left(\frac{u}{u_{\max}} \right) + \dots,$$

where ρ is local density, u is the AV's chosen speed, and $\alpha_1, \alpha_2, \alpha_3$ reflect energy use, speed preference, and safety risk. In *reinforcement learning*, we convert this cost into a reward at each time step t , typically by taking its *negative*:

$$r_t = -f(\rho_t, u_t) = -\alpha_1 \left(\frac{u_t}{u_{\max}} \right)^2 + \alpha_2 \left(\frac{u_t}{u_{\max}} \right) - \alpha_3 \left(\frac{\rho_t}{\rho_{\max}} \right) \left(\frac{u_t}{u_{\max}} \right) - \dots$$

Maximizing cumulative reward in RL then aligns with *minimizing* the MFG-ARZ cost [7].

2.6.2 Markov Decision Process (MDP) Formulation

We treat each AV (or a representative “mean AV”) as an RL agent in a discrete-time MDP:

- **State s_t** : A collection of traffic variables extracted at time t . Could include local densities $\rho_i(x, t)$, current AV speeds u_i^{AV} , neighbor-lane states for lane-change decisions, etc.
- **Action a_t** : The control input chosen by the AV (target speed, lane-change command).
- **Transition**: The PDE environment updates from t to $t+1$ based on ARZ or MFG-ARZ.
- **Reward r_t** : Negative of the PDE-based cost.
- **Discount Factor $\gamma \in [0, 1]$** .

The *Bellman equation* for the state-value function $V^\pi(s)$ under policy π is:

$$V^\pi(s) = \mathbb{E}_{a \sim \pi, s' \sim p} [r(s, a) + \gamma V^\pi(s')], \quad (6)$$

and one often uses the action-value $Q^\pi(s, a)$:

$$Q^\pi(s, a) = \mathbb{E} [r(s, a) + \gamma V^\pi(s') \mid s, a]. \quad (7)$$

2.6.3 Policy Gradient Methods

Because the *action space* (speed, lane change) can be continuous, we adopt algorithms like **DDPG** or **PPO**. A *policy* $\pi_\theta(a|s)$ is parameterized by θ . We seek θ^* that

maximizes expected return:

$$J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta} \left[\sum_{t=0}^{T-1} \gamma^t r_t \right],$$

where $\tau = (s_0, a_0, s_1, a_1, \dots)$ is a trajectory from the environment. The *policy gradient* theorem yields updates like:

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{t=0}^{T-1} \left[\nabla_\theta \log \pi_\theta(a_t | s_t) \right] (R_t - b), \quad (8)$$

with discounted return R_t and baseline b . More advanced methods (DDPG, PPO) stabilize or clip these gradients and use a *critic* to approximate Q^π .

2.6.4 Neural Network Architectures

- **Policy Network π_θ :**

- Input: the current traffic state s_t .
- Output: continuous actions (speed, etc.) or discrete lane-change decisions.
- Typically an MLP or CNN, possibly with mean-field aggregate $\bar{\rho}$.

- **Critic / Value Network $Q_\phi(s, a)$ or $V_\phi(s)$:**

- For better sample efficiency and stable training (actor-critic methods).
- Another neural net that outputs an estimate of the expected return given (s, a) .

- **Mean-Field / Aggregated Inputs:**

- In many MFG scenarios, the average density $\bar{\rho}$ or global congestion metric is fed into each agent's policy.

2.6.5 Implementation Workflow in a PDE Environment

A typical **algorithmic flow** that unifies PDE and RL [5]:

1. **Initialize** the policy π_θ and any value/critic networks.
2. **Reset** the PDE environment with initial $\rho_i(x, 0)$, $u_i(x, 0)$.
3. **For each episode:**
 - (a) Observe the initial state s_0 .
 - (b) **For** $t = 0, \dots, T_{\max}$:
 - The *AV agent* picks action $a_t \sim \pi_\theta(\cdot | s_t)$.
 - The *PDE solver* evolves for Δt , applying a_t as speed/lane commands for AV, while HV follow ARZ.
 - A new state s_{t+1} is measured.
 - Reward $r_t = -f(\rho_t, u_t)$ is computed.
 - The transition (s_t, a_t, r_t, s_{t+1}) is stored.
 - (c) **Update** θ (and possibly ϕ) using RL gradient steps.

This approach allows *learning-based* or data-driven policies to approximate the optimal control strategies that would otherwise require solving high-dimensional HJB equations directly, as noted in [5, 7].

We have now presented a *comprehensive Model and Methodology* for multi-lane traffic flow with HV and AV. The PDE-based Aw–Rascle–Zhang [3, 4, 6] and Mean Field Game [7] formulations handle lane changes, merges, and node constraints; the *reinforcement learning* extension interprets the PDE environment as an MDP, bridging the MFG cost structure with RL-based control [5].

3 Numerical Experiments

Below is a **unified description** of how the experiments are set up, including the *physical parameters, initial conditions, boundary conditions, what is being measured, and which figures* are generated in each of the three scenarios. We only *describe* how the experiments are conducted and what each figure shows, without discussing stability outcomes or numerical results.

3.1 Experimental Settings

We adopt similar physical and numerical setups across the three extended scenarios (multi-lane ring, multi-lane merging, and multi-agent RL), each tailored to reflect the problem at hand.

1. **Free-Flow Speed:**

$$u_{\max} = 30 \text{ m/s.}$$

2. **Jam Density:**

$$\rho_{\text{jam}} = \frac{1}{7.5} \text{ vehicles/m.}$$

3. **Hesitation Function:**

For PDE-based simulations (Scenarios 1 & 2), we use

$$h(\rho) = 9 \text{ m/s} \times \sqrt{\frac{\rho/\rho_{\text{jam}}}{1 - \rho/\rho_{\text{jam}}}},$$

mirroring the form in Ref. [6] of the original paper.

4. **Road Length / Domain:**

- **Scenario 1:** A ring of length $L = 1 \text{ km}$.
- **Scenario 2:** A main road of length 1000 m and an on-ramp of length 300 m, each with multiple lanes. The ramp merges at 80% of the main.
- **Scenario 3:** A ring of length 200 m for multi-agent RL, discretized only for visualization bins.

5. **Time Horizon:** Typically $T = \frac{2L}{u_{\max}}$, e.g. 66.7 s for a 1 km ring at 30 m/s, or 50–60 s in the merging scenario. For RL, we run each episode for a certain number of steps (e.g., 50 steps), across multiple episodes.

6. **Density or Inflow:**

- **Scenario 1:** We set a uniform base density $\bar{\rho}^{\text{TOT}}$ and vary it along with the AV ratio $\bar{\rho}^{\text{AV}}/\bar{\rho}^{\text{TOT}}$.
- **Scenario 2:** We specify an inflow boundary for the main road (0–0.05 veh/m) and ramp inflow (0–0.025 veh/m).
- **Scenario 3:** We fix total vehicles in the ring (e.g. 6–12) and vary the fraction that is AV (0%–100%).

7. Initial Conditions:

- **PDE:** Typically, a near-uniform density plus a 10% sine-wave or random perturbation. HV velocities are initialized near the desired speed \bar{u} , and AV velocities likewise start near \bar{u} .
- **RL:** Each vehicle's position (x , lane) and speed is randomized at reset, with partial or no knowledge of other vehicles.

8. Error Function / Stability Criterion:

For PDE runs, we define

$$E(t) = \sum_{i \in \{\text{AV}, \text{HV}\}} \left\| \rho^i(\cdot, t) - \bar{\rho}^i \right\| + \left\| u^i(\cdot, t) - \bar{u} \right\|,$$

measured over the spatial domain (e.g. in an L^1 or L^2 norm). For RL, we use collision counts or speed distributions as an indicator of stable vs. unstable behavior.

3.2 Implementation and Figure Generation

Below we outline *what each figure* contains and *how* it is generated, without interpreting the results:

3.2.1 Scenario 1 (Multi-Lane Ring)

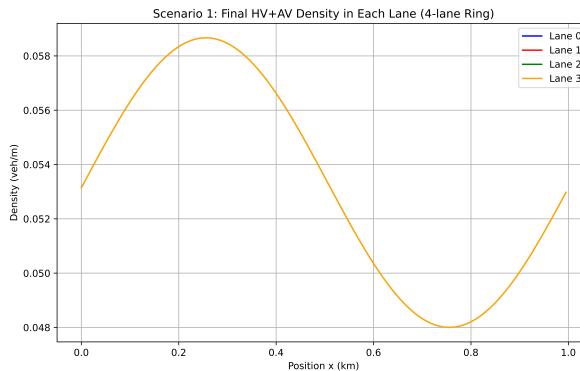


Fig. 1 Final HV+AV Density in Each Lane (4-Lane Ring)

- **Figure 1:** Plots the *final HV+AV density* in each lane $\rho_i(x)$ over the ring's spatial range $[0, L]$. Each lane is a colored curve (e.g. lane 0 in blue, lane 1 in red). This figure is produced after the PDE simulation ends, sampling ρ_i in each lane.

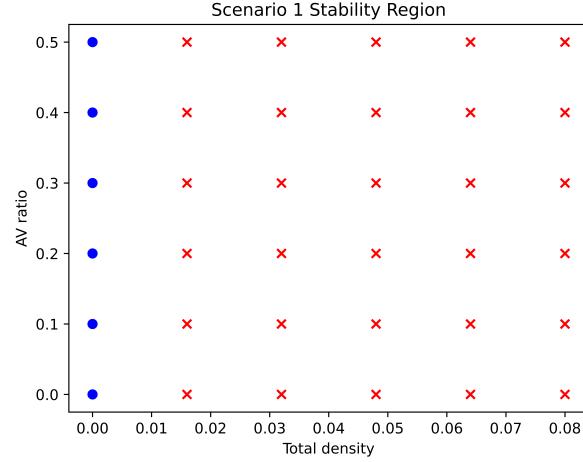


Fig. 2 Stability Region (AV Ratio vs. Total Density)

- **Figure 2:** Shows a *stability region* in a 2D parameter space—e.g. total density vs. AV ratio. Each point is stable (marked by one symbol/color) or unstable (marked differently). We run the PDE code for each parameter pair and record if $\max_{0 \leq t \leq T} E(t)$ surpasses a threshold.

Scenario 1: Density Evolution (Lane 0)

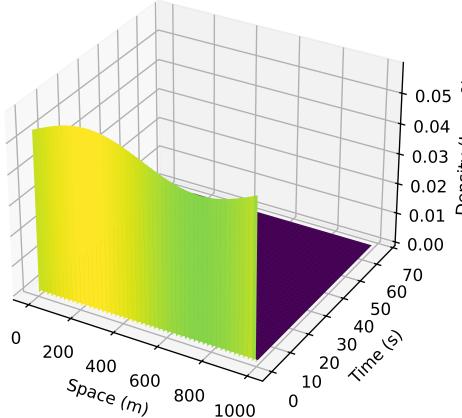


Fig. 3 3D Density Evolution

- **Figure 3:** A 3D time-space-density surface for a single lane. After each PDE time step, we store $\rho(\cdot, t)$ across space and stack them in a 2D array. Plotting ρ in the vertical axis vs. space on one horizontal axis and time on the other yields the surface.

3.2.2 Scenario 2 (Two-Lane Main Road + Two-Lane Ramp)

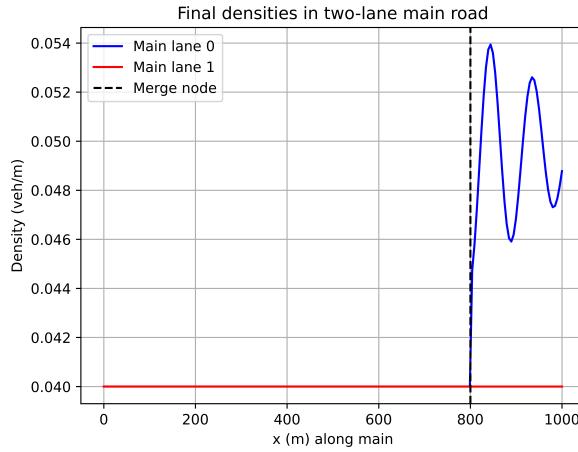


Fig. 4 Final Densities in Two-Lane Main Road

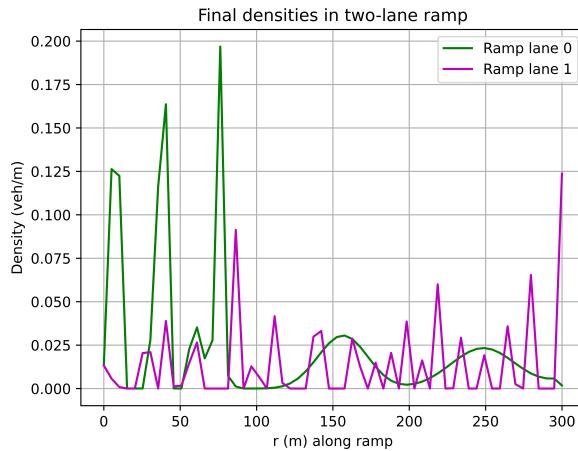


Fig. 5 Final Densities in Two-Lane Ramp

- **Figure 4 and Figure 5:** Plots the *final densities* in the main-lanes (top) and ramp-lanes (bottom). Each lane has a separate line. The main-lane plot also includes a

dashed line indicating the merge node location. We gather $\rho_i(x)$ from each PDE lane array at the final time.

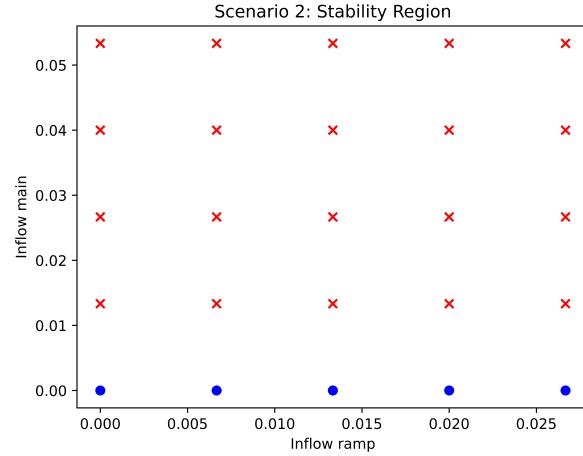


Fig. 6 Stability Region (Inflow Main vs. Inflow Ramp)

- **Figure 6:** A *stability map* over ramp inflow (x-axis) vs. main inflow (y-axis). For each $(\text{inflow}_{\text{ramp}}, \text{inflow}_{\text{main}})$ pair, the PDE solver runs up to time T . We mark stable vs. unstable points as different symbols.

Scenario 2: Lane0 density evolution

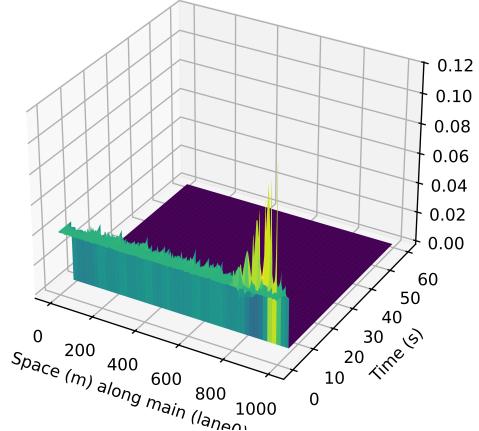


Fig. 7 Stability Region (Inflow Main vs. Inflow Ramp)

- **Figure 7:** A *3D time-space-density* plot for one main-lane (e.g. lane 0), similar to Scenario 1's surface plot. We record $\rho(x, t)$ across time steps to visualize wave propagation or local accumulation.

3.2.3 Scenario 3 (Multi-Agent RL)

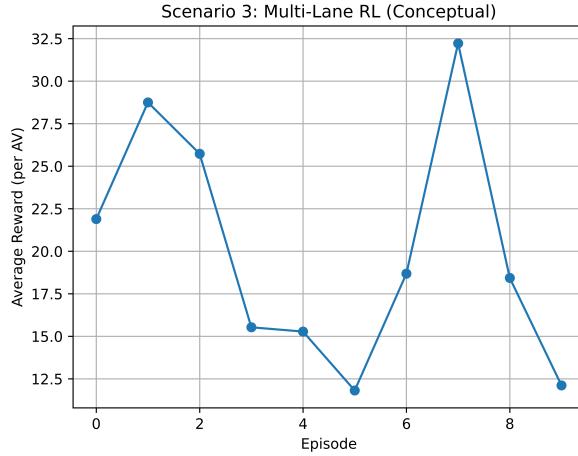


Fig. 8 Stability Region (Inflow Main vs. Inflow Ramp)

- **Figure 8:** Plots the *episode reward* vs. the episode index. After each RL episode, we compute the average or total reward across the AV agents and store it. This figure is a simple line chart showing how reward evolves over multiple episodes.

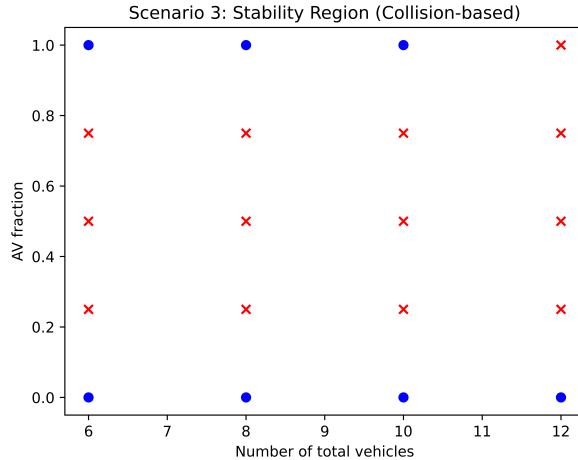


Fig. 9 Stability Region (Inflow Main vs. Inflow Ramp)

- **Figure 9:** Depicts a *collision-based stability region*, e.g. number of total vehicles (x-axis) vs. AV fraction (y-axis). Each run is stable if collisions remain below a threshold, marked in one color, or unstable otherwise.

Scenario 3: 'Density' Evolution (Discretized Bins)

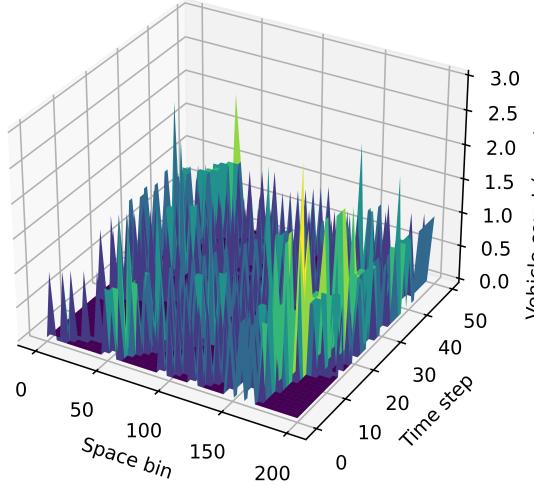


Fig. 10 Stability Region (Inflow Main vs. Inflow Ramp)

- **Figure 10:** Shows a “density” evolution in discretized spatial bins over time. We partition the ring into bins, count how many vehicles are in each bin at each time step, and produce a 3D surface (bin index, time, vehicle count). This provides a PDE-like “visualization” of vehicle clustering.

4 Conclusion

Based on the observations from the three scenarios, we can draw several main points:

4.1 Autonomous Vehicles (AVs) Help Stabilize Traffic

In both the ring (Scenario 1) and merging (Scenario 2) simulations, when the fraction of AVs is higher, the flow is more stable. In the multi-agent RL (Scenario 3), adding more AVs often reduces collisions and makes the flow smoother.

4.2 High Density or Inflow Often Leads to Instability.

Large total density in the ring or large inflow at the merge node creates stop-and-go waves or spikes in density. The RL scenario shows more collisions if there are many vehicles and not enough AVs.

4.3 Merging bottlenecks can form quickly.

If ramp inflow is high and the main road is already near capacity, we see local congestion at the merge node. This matches real-world observations of highway ramp bottlenecks.

4.4 Simplified models offer insight but have limits.

We use reduced PDE forms of the ARZ model and basic RL strategies. While they capture essential features (e.g., wave growth, collision rates), more advanced methods would be needed for real-world accuracy.

Overall, our results suggest that well-designed AV control and moderate inflows can keep traffic stable under a range of conditions. Future work may refine node models, use more realistic driver or agent behaviors, and explore richer RL approaches. By doing so, we could better predict and manage high-density traffic flows in complex networks.

References

- [1] Huang, K., Di, X., Du, Q., *et al.*: Scalable traffic stability analysis in mixed-autonomy using continuum models. *Transportation Research Part C: Emerging Technologies* **111**, 616–630 (2020)
- [2] Huang, K., Di, X., Du, Q., Chen, X.: Stabilizing traffic via autonomous vehicles: A continuum mean field game approach. In: Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, p. (2019)
- [3] Aw, A., Rascle, M.: Resurrection of ‘second order’ models of traffic flow. *SIAM Journal on Applied Mathematics* **60**(3), 916–938 (2000)
- [4] Zhang, H.M.: A non-equilibrium traffic model devoid of gas-like behavior. *Transportation Research Part B: Methodological* **36**(3), 275–290 (2002)
- [5] Fan, S., Work, D.B.: A heterogeneous multiclass traffic flow model with creeping. *SIAM Journal on Applied Mathematics* **75**(2), 813–835 (2015)
- [6] Seibold, B., Flynn, R.M., Kasimov, A.R., Rosales, R.R.: Constructing set-valued fundamental diagrams from jamiton solutions in second order traffic models. [arXiv: 1201.3804](https://arxiv.org/abs/1201.3804) (2012)
- [7] Huang, K., Xi, S., Du, Q., Chen, X.: A game-theoretic framework for autonomous vehicles velocity control: Bridging microscopic differential games and macroscopic mean field games. [arXiv:1903.06053](https://arxiv.org/abs/1903.06053) (2019)