AUTONOMOUS TRAFFIC THROUGH SELF-PLAY

Anonymous authors
Paper under double-blind review

ABSTRACT

Traffic rules—lanes, prescribed travel directions, signals, and right-of-way—solve a coordination problem, designed around the limitations of human drivers. We ask what organization a population of autonomous vehicles would discover if these rules were not imposed, and how it would compare with rule-based autonomous traffic. Without rules, vehicles must converge on conventions for passing, merging, and intersection traversal that no one specifies, in continuous two-dimensional road space, with each vehicle acting independently based on local observations. To learn at the required scale, we introduce OpenDrive-GPU, a batched multi-agent driving simulator, and AutoZero, a self-play method combining spatial attention with phasic policy–value learning. Trained on several road networks from hundreds of billions of vehicle transitions without human demonstrations, a single shared policy exhibits recognizable crossing, yielding, merging, and passing behaviors. Evaluated on held-out road networks, rule-free autonomous traffic sustains $1.78\times$ and $2.63\times$ the throughput of the best-performing rule-based traffic at medium and high demand, respectively. While these results demonstrate the efficiency potential of self-organized autonomous traffic, they also reveal safety challenges: collisions and road departures increase under heavy demand, and throughput shows sensitivity to failure clearance. These findings identify safe coordination as a central open challenge for rule-free autonomous traffic.

1 INTRODUCTION

Traffic rules—lanes, prescribed travel directions, signals, and right-of-way conventions—are a solution to a coordination problem, designed around the limitations of human drivers. They make behavior predictable when perception is imperfect, reactions are slow, and skill varies across individuals, but at a cost: they restrict each vehicle to a small set of discrete choices, leave road space and intersection time unused, and allow small disturbances to amplify into stop-and-go waves (Treiber et al., 2000; Helbing, 2001). Autonomous vehicles relax many of the limitations these rules were designed around: they perceive continuously, react within a single control cycle, and can exchange state with nearby vehicles. This raises a basic question: *if traffic rules were not imposed, what organization would a population of autonomous vehicles discover on its own, and how would it compare with rule-based autonomous traffic in efficiency and safety?*

Answering this question poses a difficult multi-agent learning problem. Without rules, vehicles must agree on conventions they were never given: which side to pass on, how to merge, and how to share intersections. Many such conventions are equally valid—driving on the left is no better than driving on the right (Pal et al., 2021)—so the population must coordinate on the same choice through repeated interaction (Hu et al., 2020). This problem arises in continuous two-dimensional space, among many simultaneously interacting agents, each optimizing its own return from local observations.

Prior work has relaxed parts of the conventional traffic system, but not all of it at once. Lane-free traffic lets automated vehicles use the full road width (Papageorgiou et al., 2021; Malekzadeh et al., 2021; Troullinos et al., 2021; Berahman et al., 2025), and signal-free intersection management replaces traffic lights with reservations or optimization (Dresner & Stone, 2008; Xu et al., 2022; Yan et al., 2024); both typically retain other structure, such as travel directions, reference paths, or a central coordinator. Large-scale self-play, enabled by batched simulators (Kazemkhani et al., 2025; Gulino et al., 2023), now produces robust driving policies without human demonstrations (Cusumano-Towner et al., 2025; Wang et al., 2026; Wu et al., 2026; He et al., 2026), but these

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

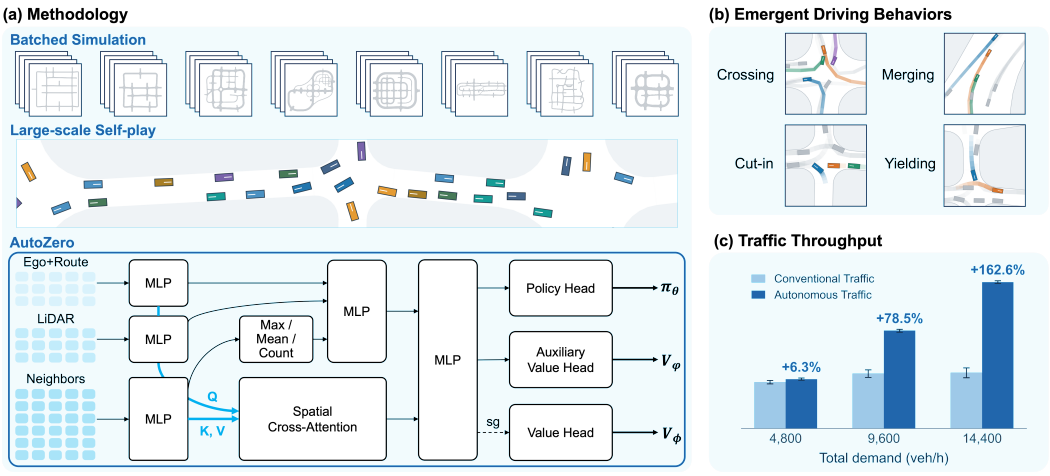

Figure 1: Overview of autonomous traffic through self-play. (a) Batched simulation supports large-scale self-play across eight road networks. AutoZero encodes local observations into a shared representation with policy and value heads; *sg* denotes stop-gradient. (b) Emergent driving behaviors including crossing, merging, cut-in, and yielding. (c) AutoZero sustains substantially higher throughput than the best-performing conventional traffic baseline at each demand level, with gains increasing at higher demand.

agents learn *how to drive within* a predefined system of lanes, directions, signals, and priorities. Closest to our goal, Pal et al. (2021) show that lane following, signal compliance, and right-of-way can emerge from multi-agent training with noisy perception, in single intersections and highway segments with about ten agents. We study emergence at the scale of complete road networks with dense traffic and no signals at all, and compare the resulting system directly with rule-based traffic in both efficiency and safety.

We study an all-autonomous setting in which vehicles drive through continuous road networks with no lane markings, prescribed travel directions, signals, or predefined right-of-way (Figure 1). Each vehicle selects its own steering and acceleration from local observations that include the states and planned routes of nearby vehicles, as with vehicle-to-vehicle communication. To learn at the required scale, we build OpenDrive-GPU, a batched simulator that runs thousands of dense multi-vehicle worlds in parallel, and AutoZero, a self-play method that combines spatial attention over neighboring vehicles with phasic policy–value learning. A single shared policy, trained on eight road networks from hundreds of billions of vehicle transitions without human demonstrations, exhibits recognizable crossing, yielding, insertion, passing, merging, and following behaviors.

Evaluated on a 4×4 grid network unseen during training, with matched demand and routes, the learned policy sustains $1.78\times$ and $2.63\times$ the throughput of the best-performing baseline at medium and high demand, respectively. However, these efficiency gains come with safety challenges: at high demand, 0.87% of offered trips end in collisions and 0.27% in road departures, and under a stress test at twice this demand, 14.8% of admitted vehicles collide. These results highlight both the efficiency potential of self-organized traffic and the challenge of maintaining safe coordination as demand grows. The evaluation compares complete traffic systems with different controllers, simulators, and information structures.

Our contributions are:

- We introduce OpenDrive-GPU, a batched multi-agent driving simulator that enables large-scale training across continuous road networks, and AutoZero, a self-play method combining spatial attention with phasic policy–value learning.
- We document recognizable passing, yielding, merging, and intersection behaviors that emerge without explicit programming, using representative trajectories and scenario-level statistics to characterize recurring coordination patterns.

- We evaluate the learned policy on held-out road networks across varying demand levels. AutoZero sustains substantially higher throughput than signal-controlled baselines, but with increased collision and road-departure rates under heavy demand. Additional experiments examine demand stress, failure-retention sensitivity, and individual rule interventions to characterize the scope and limitations of these gains.

2 Related Work

Self-play for autonomous driving Self-play has emerged as a scalable alternative to learning from human demonstrations. Early work studied structured interactive settings, including multi-agent negotiation in highway merging (Tang, 2019) and robust adaptation through population synthesis in autonomous racing (Sinha et al., 2020), while asymmetric self-play has been used to generate challenging driving curricula (Zhang et al., 2024). Recent work has scaled self-play to increasingly diverse and realistic driving environments: policies trained across large collections of real-world scenes (Cornelisse et al., 2025) or from massive simulated interaction (Cusumano-Towner et al., 2025; Wang et al., 2026; Wu et al., 2026; He et al., 2026). Subsequent work extends to end-to-end driving (Rowe et al., 2026) and explicit cooperation mechanisms such as natural-language communication (Cui et al., 2026). Related work has incorporated human data or centralized reference models to promote human-compatible behavior (Cornelisse & Vinitsky, 2024; Chang et al., 2026; Cornelisse et al., 2026), while examining generalization across heterogeneous populations (Distelzweig et al., 2026) and sensitivity to reward design (Sisask et al., 2026). Despite this progress, existing self-play approaches primarily learn *how to drive within a predefined traffic system*: even when human demonstrations are removed, the environment typically retains lane geometry, prescribed traffic directions, signals, or right-of-way conventions. Our work treats traffic organization itself as emergent, allowing conventions to arise through interaction in continuous road space without predefined lanes, directions, signals, or priority rules.

Lane-free and unsignalized autonomous traffic Lane-free traffic (Papageorgiou et al., 2021; Troullinos et al., 2021; Berahman et al., 2025) lets automated vehicles use full road width, while signal-free intersection management (Dresner & Stone, 2008; Yan et al., 2024; Antonio & Maria-Dolores, 2022) replaces traffic lights with reservations or optimization. Recent work progressively relaxes constraints through lane-free and signal-free methods (Li et al., 2021; Ahmadi et al., 2024; Naderi et al., 2025; Zeng & Ramezani, 2026). Despite removing lanes, signals, and fixed paths, these approaches typically retain prescribed travel directions, reference trajectories, or centralized coordination. Mixed-autonomy work (Wu et al., 2017; 2021) shows that learned vehicles can reshape traffic flow, but within lane-based networks with human-driven vehicles. Our setting removes these organizational priors jointly, allowing traffic organization to emerge from decentralized self-play.

3 Methodology

We first formulate autonomous traffic as a decentralized multi-agent learning problem and then introduce AutoZero, our self-play method.

3.1 Problem Formulation

We formulate autonomous traffic as a symmetric partially observable stochastic game (POSG) in which each vehicle optimizes its own expected discounted return from local observations while interacting with other vehicles. The global state includes road geometry, assigned routes, and the physical states of all vehicles, but each vehicle observes only its local neighborhood. Given local observations, vehicles sample actions from a decentralized policy and the environment transitions forward. All vehicles share the same observation, action, and reward structure, making the game symmetric, though no global team objective is provided.

We study a homogeneous setting in which all vehicles share a common policy through parameter sharing during self-play training. We assume ideal vehicle-to-vehicle (V2V) communication: each vehicle can reliably access the states and planned routes of nearby vehicles without delay or packet loss. This information-sharing assumption enables coordination through local interaction rather than explicit negotiation protocols. The complete formal definition is provided in Appendix A.

Action. Each vehicle outputs a continuous two-dimensional control action $u_t^i = (u_{a,t}^i, u_{\delta,t}^i) \in [-1, 1]^2$, where $u_{a,t}^i$ is the longitudinal action and $u_{\delta,t}^i$ is the steering action.

Observation. Each agent receives a 499-dimensional local observation containing its own kinematic state and geometry, route guidance and forward route samples, the states and routes of nearby vehicles, and LiDAR measurements of road boundaries and surrounding vehicles. The policy has access to local navigation, interaction, and free-space information, but not the global traffic state.

Reward. We use a reward function that promotes route progress and penalizes collisions, road departures, elapsed time, and abrupt control changes:

$$r_t^i = I_{g,t}^i - I_{c,t}^i - I_{o,t}^i + 0.010\,\Delta s_t^i - 0.020\,\Delta t - 0.001\left(\frac{\Delta u_{a,t}^i}{2}\right)^2 - 0.001\left(\frac{\Delta u_{\delta,t}^i}{2}\right)^2, \quad (1)$$

where $I_{g,t}^i$, $I_{c,t}^i$, and $I_{o,t}^i$ indicate goal arrival, collision, and road departure.

3.2 AutoZero

We introduce **AutoZero**, a self-play method designed for dense multi-agent traffic without predefined rules (Figure 1a). The core challenge is learning coordination from local observations in continuous space where conventions must emerge through interaction. AutoZero addresses this through spatial attention over neighboring vehicles combined with phasic policy–value learning that alternates every iteration rather than every several epochs as in standard PPG (Cobbe et al., 2021).

The encoder processes local observations through separate pathways for ego state, route guidance, LiDAR scans, and neighboring vehicles. A key design choice is how to aggregate information from a variable number of neighbors. We use two complementary approaches: a *set pathway* computes permutation-invariant statistics (elementwise maximum, mean, and normalized count) to capture overall traffic density and flow, while a *spatial attention pathway* uses multi-head attention to focus on specific vehicles relevant to the current situation—such as an oncoming vehicle at an intersection or a merging neighbor. Invalid neighbor slots are masked, ensuring the combined encoding (**SAttn**) is invariant to neighbor ordering. This feeds into a shared MLP trunk yielding features $h_\phi(o)$.

A policy head predicts Gaussian means with learned standard deviation; actions are sampled as $u = \tanh(\mu_\theta(o) + \sigma \odot \epsilon)$ with $\epsilon \sim \mathcal{N}(0, I)$. Two value heads estimate returns: an ordinary value head $V(o) = v_\psi(\mathrm{sg}[h_\phi(o)])$ with stopped gradients for advantage estimation, and an auxiliary value head $V_{\mathrm{aux}}(o) = v_\eta(h_\phi(o))$ that trains the shared features. Architecture specifications are provided in Appendix C.

Unlike standard PPG, which alternates policy and auxiliary phases every several policy epochs, AutoZero performs both phases in every iteration. After each rollout collection, we execute a policy phase that updates the policy and shared features with clipped PPO (Schulman et al., 2017) while the ordinary value head fits return targets with stopped gradients. Immediately following, an auxiliary phase retains observations and return targets $\hat{R}$ from the current rollout, snapshots the policy as π_{ref}, and minimizes

$$\mathcal{L}_{\mathrm{aux}} = \mathbb{E}_{o,\hat{R}\sim\mathcal{D}}\left[\frac{1}{2}\big(V_{\mathrm{aux}}(o) - \hat{R}\big)^2 + \frac{1}{2}\big(V(o) - \hat{R}\big)^2 + \beta\, D_{\mathrm{KL}}\big(\pi_{\mathrm{ref}}(\cdot \mid o) \,\|\, \pi_\theta(\cdot \mid o)\big)\right]. \quad (2)$$

Both the main experiments and the method ablation use auxiliary updates after every policy phase, with KL weight $\beta = 4$. The auxiliary value head sends gradients into the shared features to improve value prediction quality, while the KL term prevents large policy shifts during feature updates. This more frequent auxiliary training enables the policy to better leverage useful value features. Method ablations in Appendix D demonstrate that AutoZero achieves substantially higher sample efficiency than baseline alternatives.

3.3 Simulation and Training

OpenDrive-GPU simulator. OpenDrive-GPU batches many independent road networks and vehicle populations in parallel through a C++/CUDA simulation core and a Python tensor interface.

The road surface is represented by connected polygonal regions with traversable portals, rather than by lane-constrained vehicle motion. Routes provide navigation and progress signals, but do not prescribe lateral position or right-of-way decisions. Vehicles evolve under continuous bicycle-model controls; coordination emerges from learned policies.

The simulator is designed for large-scale parallel training through four key implementation choices: **(i)** batched, device-resident execution with contiguous arrays and CUDA kernels that connect directly to policy inference without host memory transfers; **(ii)** persistent, shared geometry where compiled maps and routes are uploaded once and reused across worlds; **(iii)** spatially restricted queries using dynamic grids and static tile indices to limit collision, neighbor, and LiDAR queries to local candidates; **(iv)** observation construction reuse that computes route geometry once per vehicle and transforms it into multiple observer frames. On a consumer-grade RTX 3090 GPU, OpenDrive-GPU achieves up to 22.93 million active agent-steps per second across 64 parallel worlds with 256 vehicles each. Implementation details and throughput benchmarks are provided in Appendices B.1 and B.2.

Training environments and traffic generation. We train across eight road networks adapted from CARLA: Town01–Town07 and Town10[1]. Each is represented as a continuous drivable region without lane divisions, prescribed travel directions, traffic signals, or right-of-way rules. We further adapt these networks by adding open boundary ports that serve as vehicle sources and sinks. At each episode reset, boundary ports are randomly assigned as *sources*, where vehicles enter, or *sinks*, where trips terminate. Origin–destination pairs and routes are sampled from a fixed resident pool containing up to 4,096 compiled routes per map. We train at nominal densities $\rho^* \in \{1, 2, 3, 4.5, 6, 8\}$ vehicles per $1000\,\mathrm{m}^2$, with Poisson arrivals calibrated to map area and reference travel time. Full training configurations are provided in Appendices B and C.

4 Evaluation and Analysis

Unless otherwise specified, evaluations use a fixed AutoZero policy trained on approximately 350 billion active vehicle transitions. We examine emergent driving behaviors, compare intersection and network efficiency, and analyze failure under increasing demand. Method and reward ablations are reported in Appendices D and E.

4.1 Emergent Driving Behaviors

Figure 2 illustrates crossing, yielding, waiting and insertion, opposing passing, merging, and following without prescribed traffic rules. The speed and position curves show how vehicles adjust their motion: waiting combines low speed with little displacement before movement resumes, while following vehicles maintain their ordering and similar speeds. Here, cut-in means insertion into a flow, and platooning describes short-term aligned following.

Cross-scenario recurrence. Across 96 recordings of the evaluation policy (12 per training map at $\rho^* = 8$), motion-based criteria identify crossing, yielding, opposing passing, merging, and following in all 12 scenarios of every map during inflow. Waiting and cut-in candidates occur less consistently, in 0–11 scenarios per map; even in Town10, its mean fraction among eligible interaction opportunities is only 0.325%. Together, scenario recurrence and opportunity-conditioned fractions characterize how widely and how often these motion patterns occur. Definitions and uncertainty estimates are given in Appendix H.

Lateral position bias. We compare 576 scenarios with paired reflections across eight maps and six densities. Left-half occupancy among eligible moving vehicles is 81.94% on the original maps and 81.26% after reflection; all 48 map–density conditions retain a left-side majority, indicating a consistent policy preference. Completion nevertheless falls from 93.10% to 82.57%, and OOB terminations increase from 1.04% to 9.64% of offered demand. Lateral preference persists despite a substantial loss of driving performance under reflection. The distributions (Figure 11) and sampling and normalization details are provided in Appendix H.

[1] https://carla.readthedocs.io/en/0.9.11/core_map/#carla-maps

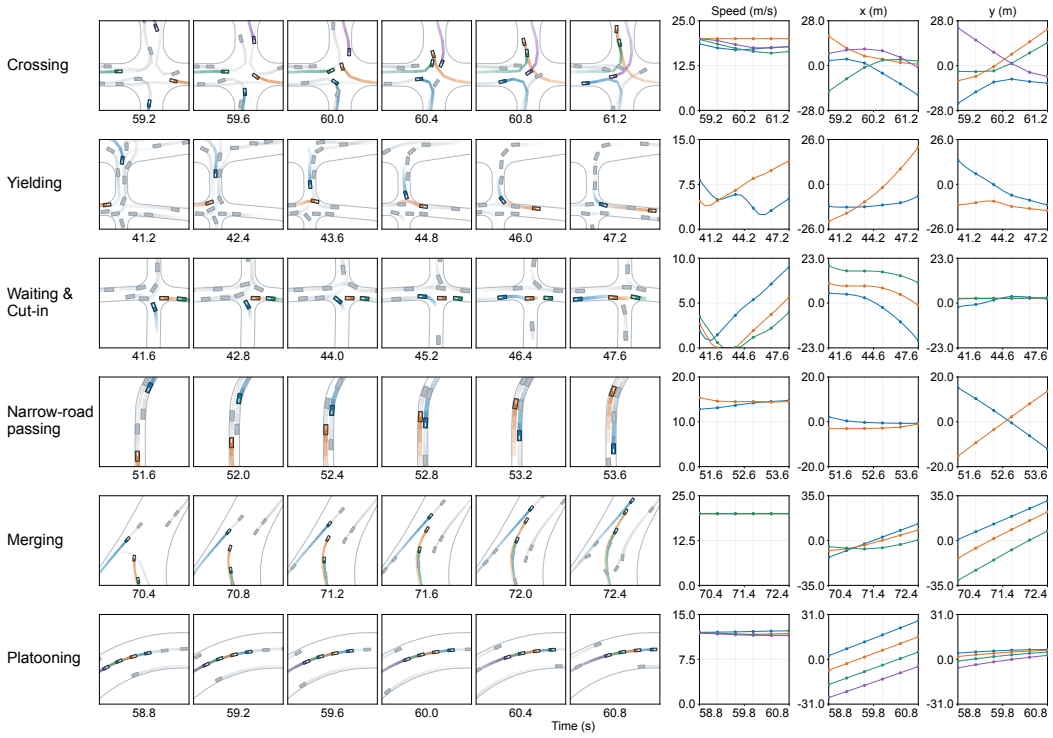

Figure 2: Representative emergent driving behaviors. Colors identify the same vehicles across snapshots and speed and position curves; surrounding vehicles are gray. Positions share a fixed origin within each row. Fading ribbons show recorded past trajectories, and curve markers identify snapshot times. Recording details appear in Appendix H.

4.2 Efficiency Evaluation

Table 1 compares finite vehicle cohorts at one intersection and continuous traffic on a road network.

4.2.1 Intersection-Level Efficiency

We compare AutoZero with FIFO-KATS and OBS-KATS (Yan et al., 2024), and lane-free MPC (Naderi et al., 2025), at an unsignalized four-way intersection with 18 m-wide roads. FIFO-KATS reserves crossing times in arrival order, whereas OBS-KATS searches crossing orders to reduce delay. Lane-free MPC jointly optimizes vehicle acceleration and steering over a receding horizon without lane constraints. All methods use the same OpenDrive vehicle dynamics and termination checks. We evaluate 12, 16, and 24 vehicles, with ten paired scenarios per size and no subsequent arrivals. AutoZero and MPC share full-width initial positions and exits; KATS uses directed lanes. Origins, destinations, longitudinal positions, headings, and initial speeds are matched across road organizations.

We measure mean travel time and cumulative stopped time ($v < 0.5$ m/s), including vehicles that never stop. All methods complete all 520 trips without collisions or OOB terminations. AutoZero reduces mean travel time relative to OBS-KATS, the strongest baseline on this metric, by 24.9%, 26.6%, and 27.4% for the three sizes. MPC stops less than OBS-KATS yet takes longer to arrive, showing that fewer stops alone do not ensure shorter trips. Appendix G gives the full protocol, controller adaptations, and representative trajectories.

4.2.2 Network-Level Efficiency

We evaluate a 4×4 intersection grid with 18 m-wide internal roads and eight sources. AutoZero uses the continuous road surface; the baselines use three 3 m lanes in each direction. Independent

Table 1: Intersection and network efficiency. Values are means ± sample standard deviations over ten paired intersection scenarios or five paired network scenarios. Intersection travel and stopped times ($v < 0.5$ m/s) are per-vehicle means in seconds; all methods complete all trips without collision or OOB. Network throughput counts successful exits during measurement; completion includes all offered requests and drainage. Bold indicates the best mean for each setting and metric.

Intersection						
# Vehicles	**12**		**16**		**24**	
	Travel time ↓	Stopped time ↓	Travel time ↓	Stopped time ↓	Travel time ↓	Stopped time ↓
FIFO-KATS	34.90±10.41	11.79±9.98	35.72±10.31	12.54±10.25	35.77±2.81	12.39±2.56
OBS-KATS	22.16±1.13	0.47±0.41	23.40±1.07	1.29±0.60	25.97±1.41	3.17±0.94
MPC	27.49±1.01	**0.00**±0.00	27.41±0.80	**0.00**±0.00	27.49±0.62	0.03±0.06
AutoZero	**16.65**±0.32	**0.00**±0.00	**17.17**±0.36	**0.00**±0.00	**18.85**±1.03	**0.01**±0.02
Network						
Demand	**Low (4,800 veh/h)**		**Medium (9,600 veh/h)**		**High (14,400 veh/h)**	
	Throughput ↑	Completion ↑	Throughput ↑	Completion ↑	Throughput ↑	Completion ↑
Actuated	4320±353	97.44±3.10	5342±483	73.34±6.71	5428±620	52.26±6.16
Max-Pressure	4514±212	**100.00**±0.00	5189±308	74.15±4.30	5191±346	50.51±4.64
MPLight	4402±217	98.98±0.27	5078±546	71.96±9.50	5311±466	49.03±6.42
AutoZero	**4798**±147	99.77±0.18	**9534**±194	**99.36**±0.20	**14253**±166	**98.85**±0.19

Poisson arrivals produce total demands of 4800, 9600, and 14400 veh/h. Five paired scenarios per demand share request times, origin–destination pairs, and road-level routes across methods. Vehicle dimensions, speed limits, and longitudinal acceleration bounds are matched.

We compare three traffic signal control methods: Actuated (Eclipse SUMO, n.d.), Max-Pressure (Varaiya, 2013), and MPLight (Chen et al., 2020), all using SUMO with CACC and LC2013 lane changing. MPLight is adapted from LibSignal (Mei et al., 2022), with one model per demand trained for 200 episodes and selected on the same five scenarios used for evaluation. AutoZero uses deterministic actions without grid fine-tuning. Baseline settings and selection procedures are given in Appendix F.

Each run includes 300 s of warm-up, 900 s of measurement, and 600 s of drainage, with new requests stopping at 1200 s. Throughput counts successful exits during measurement. Completion and failure rates include all offered requests, including those never admitted or unfinished, with outcomes observed through drainage. Vehicles unable to enter safely wait outside the network; failed vehicles are removed immediately and never count toward successful throughput. Results are equally weighted scenario means and sample standard deviations.

AutoZero improves throughput over the best-performing baseline at each demand by 6.3%, 78.5%, and 162.6%, respectively. At High demand, AutoZero achieves 14253 veh/h and 98.85% completion, compared with 5428 veh/h and 52.26% for Actuated. Its throughput closely tracks offered demand, whereas signal-controlled traffic saturates near 5200–5400 veh/h.

This efficiency advantage has a safety cost: at High demand, 0.87% of offered trips end in collisions and 0.27% in road departures. All three baselines have zero observed failures in these runs. The comparison therefore demonstrates higher successful flow under the stated conditions, rather than safer operation. Additional efficiency statistics and safety results appear in Appendix F.

4.3 Demand Stress and Failure Modes

We extend demand to 28800 veh/h in six levels, retaining the policy, five paired scenarios, and evaluation timing from Section 4.2.2. Actuated and Max-Pressure retain their calibrated settings. Completion includes all offered requests; stress-test collision and OOB rates instead use admitted vehicles as the denominator.

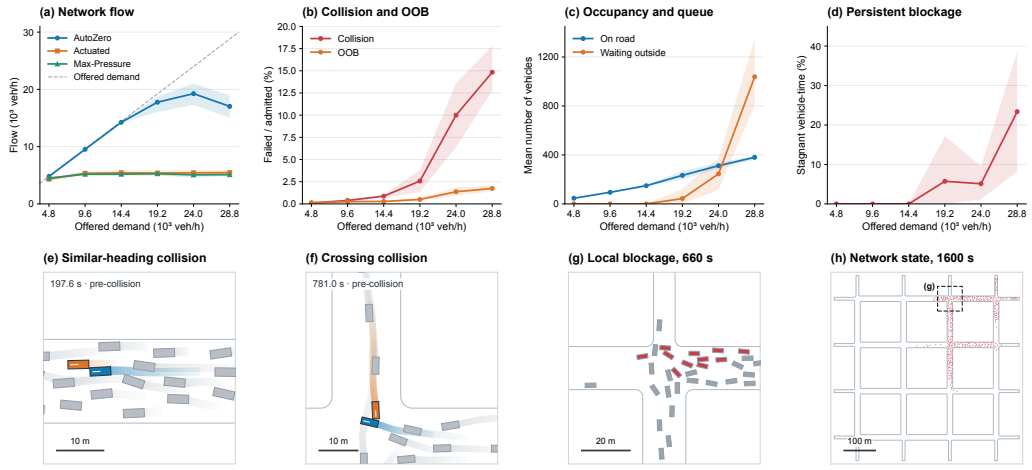

Figure 3: Demand stress and failure modes. (a) Successful flow versus offered demand (dashed line). (b–d) AutoZero failures, population and external queues, and persistent stagnation. Curves show five-scenario means with bootstrap 95% intervals. (e–f) Similar-heading and crossing collisions; colored vehicles are the participants and ribbons show past motion. (g–h) Local and later network blockage in one episode; red vehicles meet the preceding-30-s stagnation criterion.

Efficiency and safety under load. Successful flow reaches 19255.2 veh/h at demand 24000 and falls to 17032.0 at 28800 (Figure 3). This suggests saturation, although the paired confidence interval does not establish a strict decline or steady-state capacity. From demand 14400 to 28800, collision/admitted rises from 0.87% to 14.83%, OOB/admitted from 0.27% to 1.74%, and completion falls from 98.85% to 67.24%. The mean external queue reaches 1038.3 vehicles. Additional demand thus produces increasing failures and accumulation rather than proportional successful flow. AutoZero still exceeds Actuated and Max-Pressure, which achieve 5456.8 and 5104.0 veh/h at the highest demand. The remaining throughput advantage thus coexists with increasing failures and congestion.

Persistent stagnation and collisions. We distinguish temporary stops from sustained low speed and negligible route progress over 30 s. The corresponding fraction of on-road vehicle-time is zero at the three lowest demands and 23.38% at the highest, with non-monotonic changes at intermediate stress levels. Figure 3(e–f) shows collisions both between nearly aligned vehicles and along crossing paths. In the episode in (g–h), a local cluster is followed by network-wide stationary queues: all 350 on-road vehicles remain motionless over the final 300 s, with no successful exits and 708 requests still outside. These results highlight the need for future research on preventing collisions and persistent gridlock in large-scale traffic.

Sensitivity to failure clearance. At demand 14400 veh/h, retaining failed vehicles as stationary obstacles for 60 s reduces throughput from 14252.8 to 8350.4 veh/h (41.41%) and completion from 98.85% to 65.64%. Collision/offered rises from 0.87% to 10.75%. These five paired comparisons show that high throughput depends strongly on prompt failure clearance. The intervention treats failed vehicles as stationary obstacles; full results and modeling assumptions are in Appendix J.

4.4 Limitations

The network comparison evaluates complete traffic systems: simulators, controllers, information, dynamics, and collision checks differ. It does not isolate the contribution of removing lanes, directions, signals, or right-of-way rules, nor establish steady-state capacity. Within-OpenDrive rule interventions (Appendix K) compare the same policy without retraining, with completion rates varying across conditions. The shorter travel times among completed trips in rule-free settings provide complementary evidence that autonomous coordination without prescribed traffic rules has the potential to improve traffic efficiency.

The study assumes only autonomous vehicles and reliable, delay-free sharing of neighbor states and routes. Human drivers, pedestrians, communication errors, and delays are not evaluated. Most experiments also remove failed vehicles immediately; Section 4.3 shows substantial deterioration when obstacles persist. Efficiency therefore remains conditional on assumptions that require further safety and robustness work.

5 Conclusion

We studied traffic organization learned by autonomous vehicles in continuous road space without lanes, prescribed travel directions, signals, or right-of-way rules. A shared policy trained through large-scale self-play exhibits recurring coordination patterns and higher successful throughput than the tested signal-controlled autonomous systems. Separate intersection experiments compare it with reservation-based and lane-free MPC controllers. These results demonstrate an efficiency opportunity under the evaluated assumptions, alongside collisions, road departures, persistent stagnation, and sensitivity to failure clearance. Safe coordination and robustness beyond the tested information and failure-handling assumptions remain open problems.

AI Use Statement

We used generative AI tools to assist with manuscript writing and editing, data analysis, figure preparation, experiment deployment and execution, and related-work searches. The research idea, core algorithm design, experimental design, and approach to interpreting the results were developed by the authors. The authors retain responsibility for all scientific judgments and for the accuracy and integrity of the paper and accompanying materials.

Ethics Statement

We acknowledge the ICLR Code of Ethics. All experiments were conducted in simulation, without human participants or personal data. The observed collisions and persistent congestion highlight the safety risks of transferring these policies to real traffic. The results do not establish readiness for deployment on public roads; such use would require further safety validation, including interactions with pedestrians and human drivers.

Reproducibility Statement

Our anonymous repository is available at `https://anonymous.4open.science/r/autozero-iclr27-100E`. Section 3 describes the algorithm, and Appendices A, B, and C specify observations, rewards, simulation, traffic generation, and training settings. Baseline adaptations, paired evaluation protocols, and behavior-analysis procedures are detailed in Appendices F, G, and H.

References

Elham Ahmadi, Alireza Olama, Rodrigo Castelan Carlson, and Eduardo Camponogara. Signal-free path-free intersection control for connected vehicles under automated driving. *IEEE Transactions on Intelligent Vehicles*, 2024.

Guillen-Perez Antonio and Cano Maria-Dolores. Multi-agent deep reinforcement learning to manage connected autonomous vehicles at tomorrow's intersections. *IEEE Transactions on Vehicular Technology*, 71(7):7033–7043, 2022.

Mehran Berahman, Athanasia Karalakou, Majid Rostami-Shahrbabaki, and Klaus Bogenberger. Multi-task lane-free driving strategy for connected and automated vehicles: A multi-agent deep reinforcement learning approach. *Engineering Applications of Artificial Intelligence*, 154:110797, 2025.

Wei-Jer Chang, Akshay Rangesh, Kevin Joseph, Matthew Strong, Masayoshi Tomizuka, Wei Zhan, et al. Spacer: Self-play anchoring with centralized reference models. In *International Conference on Learning Representations*, volume 2026, pp. 47010–47027, 2026.

Chacha Chen, Hua Wei, Nan Xu, Guanjie Zheng, Ming Yang, Yuanhao Xiong, Kai Xu, and Zhenhui Li. Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 3414–3421, 2020. doi: 10.1609/aaai.v34i04.5744.

Karl W. Cobbe, Jacob Hilton, Oleg Klimov, and John Schulman. Phasic policy gradient. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2020–2027. PMLR, 2021. URL `https://proceedings.mlr.press/v139/cobbe21a.html`.

Daphne Cornelisse and Eugene Vinitsky. Human-compatible driving partners through data-regularized self-play reinforcement learning. *arXiv preprint arXiv:2403.19648*, 2024.

Daphne Cornelisse, Aarav Pandya, Kevin Joseph, Joseph Suárez, and Eugene Vinitsky. Building reliable sim driving agents by scaling self-play. *arXiv preprint arXiv:2502.14706*, 2025.

Daphne Cornelisse, Julian Hunt, Zixu Zhang, Waël Doulazmi, Kevin Joseph, Jaime Fernández Fisac, and Eugene Vinitsky. Human-like autonomy emerges from self-play and a pinch of human data. *arXiv preprint arXiv:2606.19370*, 2026.

Jiaxun Cui, Chen Tang, Jarrett Holtz, Janice Nguyen, Alessandro G Allievi, Hang Qiu, and Peter Stone. Coopreflect: Towards natural language communication for cooperative autonomous driving via multi-agent learning. In *International Conference on Autonomous Agents and Multi-Agent Systems*, 2026.

Marco Cusumano-Towner, David Hafner, Alex Hertzberg, Brody Huval, Aleksei Petrenko, Eugene Vinitsky, Erik Wijmans, Taylor Killian, Stuart Bowers, Ozan Sener, et al. Robust autonomy emerges from self-play. *arXiv preprint arXiv:2502.03349*, 2025.

Aron Distelzweig, Faris Janjoš, Andreas Look, Anna Rothenhäusler, Daniel Jost, Oliver Scheel, Raghu Rajan, Daphne Cornelisse, Eugene Vinitsky, and Joschka Boedecker. Beyond self-play and scale: A behavior benchmark for generalization in autonomous driving. *arXiv preprint arXiv:2605.10034*, 2026.

Kurt Dresner and Peter Stone. A multiagent approach to autonomous intersection management. *Journal of artificial intelligence research*, 31:591–656, 2008.

Eclipse SUMO. Traffic lights: SUMO documentation, n.d. URL `https://sumo.dlr.de/docs/Simulation/Traffic_Lights.html`. Actuated traffic signal control. Accessed September 26, 2026.

Cole Gulino, Justin Fu, Wenjie Luo, George Tucker, Eli Bronstein, Yiren Lu, Jean Harb, Xinlei Pan, Yan Wang, Xiangyu Chen, et al. Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:7730–7742, 2023.

Hao He, Chengcheng Hu, Zirun Su, Heng Zhang, Haisong Liu, Jinke Li, Haochen Tian, Zhenwei Shen, Hongyang Li, Zhichao Li, et al. Drivezero: End-to-end driving beyond human demonstrations. *arXiv preprint arXiv:2609.06055*, 2026.

Dirk Helbing. Traffic and related self-driven many-particle systems. *Reviews of Modern Physics*, 73(4):1067–1141, 2001.

Hengyuan Hu, Adam Lerer, Alexander Peysakhovich, and Jakob Foerster. "other-play" for zero-shot coordination. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4399–4410. PMLR, 2020.

Saman Kazemkhani, Aarav Pandya, Daphne Cornelisse, Brennan Shacklett, and Eugene Vinitsky. GPUDrive: data-driven, multi-agent driving simulation at 1 million fps. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=ERv8ptegFi.

Bai Li, Youmin Zhang, Tankut Acarman, Yakun Ouyang, Cagdas Yaman, and Yaonan Wang. Lane-free autonomous intersection management: A batch-processing framework integrating reservation-based and planning-based methods. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7915–7921. IEEE, 2021.

Milad Malekzadeh, Ioannis Papamichail, Markos Papageorgiou, and Klaus Bogenberger. Optimal internal boundary control of lane-free automated vehicle traffic. *Transportation Research Part C: Emerging Technologies*, 126:103060, 2021.

Hao Mei, Xiaoliang Lei, Longchao Da, Bin Shi, and Hua Wei. LibSignal: An open library for traffic signal control. *arXiv preprint arXiv:2211.10649*, 2022. URL https://arxiv.org/abs/2211.10649.

Mehdi Naderi, Panagiotis Typaldos, and Markos Papageorgiou. Lane-free signal-free intersection crossing via model predictive control. *Control Engineering Practice*, 154:106115, 2025.

Avik Pal, Jonah Philion, Yuan-Hong Liao, and Sanja Fidler. Emergent road rules in multi-agent driving environments. In *International Conference on Learning Representations (ICLR)*, 2021.

Markos Papageorgiou, Kyriakos-Simon Mountakis, Iasson Karafyllis, Ioannis Papamichail, and Yibing Wang. Lane-free artificial-fluid concept for vehicular traffic. *Proceedings of the IEEE*, 109(2):114–121, 2021.

Luke Rowe, Roger Girgis, Rodrigue de Schaetzen, Daphne Cornelisse, Alaap Grandhi, Felix Heide, Eugene Vinitsky, Christopher Pal, and Liam Paull. Scaling self-play for end-to-end driving. *arXiv preprint arXiv:2606.19641*, 2026.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

Aman Sinha, Matthew O'Kelly, Hongrui Zheng, Rahul Mangharam, John Duchi, and Russ Tedrake. Formulazero: Distributionally robust online adaptation via offline population synthesis. In *International Conference on Machine Learning*, pp. 8992–9004. PMLR, 2020.

Laur Sisask, Ardi Tampuu, and Tambet Matiisen. What emerges and what breaks in self-play driving. *arXiv preprint arXiv:2608.30819*, 2026.

Yichuan Tang. Towards learning multi-agent negotiations via self-play. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 2427–2435. IEEE, 2019.

Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, 62(2):1805, 2000.

Dimitrios Troullinos, Georgios Chalkiadakis, Ioannis Papamichail, and Markos Papageorgiou. Collaborative multiagent decision making for lane-free autonomous driving. In *Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1335–1343, 2021.

Pravin Varaiya. Max pressure control of a network of signalized intersections. *Transportation Research Part C: Emerging Technologies*, 36:177–195, 2013. doi: 10.1016/j.trc.2013.08.014.

Zilin Wang, Saeed Rahmani, Daphne Cornelisse, Bidipta Sarkar, Alexander David Goldie, Jakob Nicolaus Foerster, and Shimon Whiteson. Learning to drive in new cities without human demonstrations. *arXiv preprint arXiv:2602.15891*, 2026.

Cathy Wu, Aboudy Kreidieh, Eugene Vinitsky, and Alexandre M Bayen. Emergent behaviors in mixed-autonomy traffic. In *Conference on Robot Learning (CoRL)*, pp. 398–407, 2017.

Cathy Wu, Abdul Rahman Kreidieh, Kanaad Parvate, Eugene Vinitsky, and Alexandre M Bayen. Flow: A modular learning framework for mixed autonomy traffic. *IEEE Transactions on Robotics*, 38(2):1270–1286, 2021.

Zhouchonghao Wu, Akshay Rangesh, Weixin Li, Wei-Jer Chang, Zachary Lee, Saeed Bonab, Tim Wang, and Wei Zhan. Terrazero: Procedural driving simulation for zero-demonstration self-play at scale. *arXiv preprint arXiv:2607.13028*, 2026.

Huile Xu, Christos G. Cassandras, Li Li, and Yi Zhang. Comparison of cooperative driving strategies for cavs at signal-free intersections. *IEEE Transactions on Intelligent Transportation Systems*, 23 (7):7614–7627, 2022.

Zhongxia Yan, Han Zheng, and Cathy Wu. Multi-agent path finding for cooperative autonomous driving. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12361–12367, 2024.

Ruihao Zeng and Mohsen Ramezani. Deep reinforcement learning for communication-free distributed control of autonomous vehicles in unstructured intersection. *IEEE Transactions on Cybernetics*, 2026.

Chris Zhang, Sourav Biswas, Kelvin Wong, Kion Fallah, Lunjun Zhang, Dian Chen, Sergio Casas, and Raquel Urtasun. Learning to drive via asymmetric self-play. In *European Conference on Computer Vision*, pp. 149–168. Springer, 2024.

A Problem Formulation Details

A.1 Action Execution

Both components of $u_t^i = (u_{a,t}^i, u_{\delta,t}^i)$ are clipped to $[-1, 1]$ before execution by the simulator. The executed, clipped actions are also used to compute the control-change terms in the reward.

A.2 Observation Encoding

Table 2 summarizes the complete 499-dimensional local observation.

Table 2: Composition of the 499-dimensional local observation.

Component	Dimension	Description
Ego state	5	Speed, acceleration, steering angle, length, and width
Route guidance	5	Lookahead direction, relative route heading, remaining distance
Ego route samples	25	Five samples at $5/10/20/40/80$ m
Neighbor features	400	25 features for each of the nearest 16 vehicles
Road LiDAR	32	Road-boundary distances along 32 rays
Vehicle LiDAR	32	Distances along 32 rays
Total	499	

The five ego-state features are speed, longitudinal acceleration, steering angle, vehicle length, and vehicle width. The five route-guidance features contain a two-dimensional basic lookahead navigation direction, the cosine and sine of the route heading relative to the ego heading, and the remaining route distance clipped at 100 m.

The ego route is additionally sampled at 5, 10, 20, 40, and 80 m ahead of the vehicle. Each route sample contributes five features: its two-dimensional relative position in the receiving ego vehicle's coordinate frame, divided by 100 m; the cosine and sine of the route tangent relative to the ego heading; and a validity indicator. These five samples therefore contribute $5 \times 5 = 25$ dimensions.

For interactions, the agent observes the nearest 16 vehicles within a 100 m radius. Each neighbor contributes 25 features. Ten describe its instantaneous state: two-dimensional relative position, two-dimensional relative velocity, the cosine and sine of relative heading, distance, a validity indicator, longitudinal acceleration, and angular velocity. The remaining 15 features are three five-dimensional samples from the neighbor's route at 5, 20, and 80 m ahead, encoded in the receiving ego vehicle's coordinate frame using the same position, tangent, and validity representation as the ego route samples. Slots without a valid neighbor are padded and identified by the validity feature. Neighbor state and route information is communicated reliably and without delay, but observations remain local because only neighbors within 100 m are included.

Finally, two independent 32-ray LiDAR scans measure distances to road boundaries and vehicles, respectively, adding 64 dimensions.

A.3 Reward Computation

For every vehicle $i \in \mathcal{I}_t$ that is active at the beginning of step t, $I_{g,t}^i$, $I_{c,t}^i$, and $I_{o,t}^i$ in Equation 1 indicate that the vehicle reaches its goal, collides, or leaves the drivable road region, respectively, during the transition from t to $t+1$. The signed route progress Δs_t^i is measured in meters, and $\Delta t = 0.1$ seconds. Positive progress is rewarded, while backward motion produces a negative contribution.

The action-difference terms are computed from consecutive executed controls:

$$\begin{aligned} \Delta u_{a,t}^i &= \mathrm{clip}(u_{a,t}^i, -1, 1) - \mathrm{clip}(u_{a,t-1}^i, -1, 1), \\ \Delta u_{\delta,t}^i &= \mathrm{clip}(u_{\delta,t}^i, -1, 1) - \mathrm{clip}(u_{\delta,t-1}^i, -1, 1). \end{aligned} \tag{3}$$

Because each clipped difference lies in $[-2, 2]$, division by two normalizes the maximum change to unit magnitude. The control-change coefficients in Equation 1 are applied once per simulation

step and are not multiplied by Δt. We use neither trip-level reward normalization nor a penalty on control magnitude itself.

Reward accounting follows the active set at the beginning of each step. In particular, a vehicle that reaches a terminal condition during a transition still receives Equation 1 for that terminal transition; it is removed from the active set only for subsequent steps.

B Simulation Details

B.1 Simulator Representation and Execution

Geometry and vehicle motion. OpenDrive-GPU represents roads and intersections as polygonal drivable areas connected through portal segments. External portals act as trip sources or destinations. A route contains an ordered sequence of areas and portals and a reference polyline. The polyline supplies lookahead features and signed arc-length progress while allowing lateral movement across the drivable surface. The assigned route defines navigation guidance, progress rewards, and the completion target.

At each 0.1 s step, bounded longitudinal and steering commands update a kinematic bicycle model. Vehicles have oriented rectangular footprints. Collision detection first queries swept axis-aligned bounding boxes, then applies separating-axis overlap tests at interpolated poses within the step. The interpolation count depends on relative motion and vehicle extent; this is a sampled sweep, not an exact continuous-time contact solver. Road-departure checks use area polygons, boundary segments, and portal transitions. A valid exit through the assigned goal portal is distinguished from leaving the road through a solid boundary. If terminal events coincide, collision takes precedence over road departure, which takes precedence over goal completion.

Batched state and spatial queries. The CUDA backend stores vehicle state in structure-of-arrays buffers with world-major, fixed-capacity slots and active masks. Slots may be reused after a vehicle terminates, so transitions also carry vehicle and episode identities. World-indexed spatial cell keys prevent cross-world interactions. Dynamic spatial indices accelerate neighbor and vehicle-ray queries; static spatial indices restrict polygon, boundary, and route queries to relevant geometry. The 16-neighbor observation cap does not truncate collision candidates or the vehicles considered for LiDAR. Road and vehicle LiDAR use geometric ray intersections rather than image rendering. Route samples are computed once per vehicle and reused when constructing the observations of its neighbors.

Persistent geometry and environment lifecycle. Compiled maps, routes, and static indices are uploaded once and shared by worlds using the same map. Vehicle dynamics, geometric checks, observation and reward construction, and demand admission run in CUDA kernels. The training driver supplies sampled demands and resets individual worlds without rebuilding their static geometry. Simulation outputs are exposed as device tensors for policy inference and rollout collection; reused output buffers must be copied when a transition needs to outlive the next step or reset. Terminal rewards belong to the agents active at the start of the step, and episode identities separate terminal transitions from newly reset observations. These choices support asynchronous episode endings within a synchronous batched simulation.

B.2 Simulator Execution Throughput

We measure complete environment-step throughput on a shared straight-road workload with 32, 128, or 256 vehicles per world and 1, 16, or 64 parallel worlds. Within each configuration, all simulators use the same initial positions, 5 m/s speed, 0.1 s time step, and zero acceleration/steering commands on a 24 m-wide road. The figure combines two protocols: the original 32-vehicle layouts use unstaggered route starts, while the $128/256$-vehicle layouts stagger the starts to leave explicit gaps between vehicle footprints. The road grows with population, with fixed local spacing within the larger pair of configurations rather than increasing congestion.

For 32 vehicles we time 64-step blocks, requiring at least 20 timed blocks; for $128/256$ vehicles we use 16-step blocks and at least five timed blocks. Both protocols use three warm-up blocks

and at least two seconds of accumulated step time per run. All 32-vehicle measurements use the same simulator versions and initial-state construction. GPU timing is synchronized. Dynamics, observations, rewards, and detection are included; initialization, JIT compilation, reset, rendering, policy inference, and learning are excluded. No vehicles enter or terminate within a block. Active counts and agreement with the analytic constant-speed trajectories are checked independently of timing.

OpenDrive-GPU and GPUDrive v0.4.0 (Kazemkhani et al., 2025) run on one RTX 3090. PufferDrive[2] runs with one or eight CPU processes restricted to eight physical cores; eight processes distribute separate worlds, not vehicles within a world; the single-world panel uses one process only. Its block-level process dispatch is timed, but transfers to a GPU policy are excluded. The fixed GPUDrive and PufferDrive versions support 64 and 32 vehicles per world by default. The 32-vehicle measurements use these unmodified builds, counting only actual active vehicles. For the larger configurations we compile separate variants changing only their vehicle-capacity constant to 128 or 256; OpenDrive-GPU uses its configurable capacity.

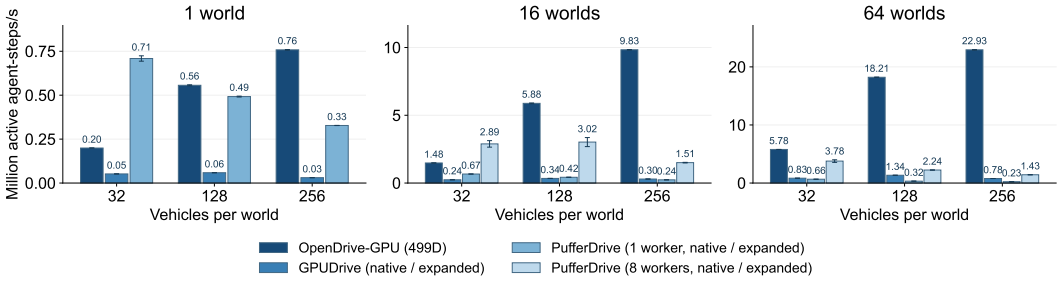

Figure 4: Simulator execution throughput on the shared moving-corridor workload. Bars show means and error bars show sample standard deviations across three independent process runs for each of 33 configurations (99 runs: 33 at 32 vehicles and 66 at 128/256 vehicles). Each panel uses its own zero-based linear vertical scale. GPUDrive and PufferDrive use native builds at 32 vehicles and capacity-expanded builds at 128/256. The comparison uses the observation workloads, CPU/GPU resources, initial-state layouts, and timing protocols described in the text.

At 64 worlds, OpenDrive-GPU achieves 18.21 and 22.93 million active agent-steps/s for 128 and 256 vehicles per world, respectively (Figure 4). The corresponding GPUDrive rates are 1.34 and 0.78 million, and eight-process PufferDrive rates are 2.24 and 1.43 million. These results describe the execution cost of each design on this workload. In the 32-vehicle, single-world case, PufferDrive instead achieves 0.71 million active agent-steps/s, compared with 0.20 million for OpenDrive-GPU and 0.05 million for GPUDrive. OpenDrive-GPU retains 499 observation features and the nearest 16 neighbors within 100 m. GPUDrive's vector observations grow to 2,951/4,103 features at the two capacities, and PufferDrive's to 1,793/2,689; both reserve neighbor slots according to world capacity. GPUDrive uses a 100 m observation radius, while PufferDrive retains its native 50 m neighbor radius. Geometry and collision semantics also differ. These workload differences limit direct attribution of throughput differences to implementation efficiency. The rates measure simulation execution under constant-speed, non-contact traffic; congested networks, frequent resets, and end-to-end training require separate measurements. Computational throughput is measured in agent-steps/s, distinct from traffic flow in veh/h.

B.3 Road Networks and Boundary Ports

We use eight road networks adapted from CARLA: Town01–Town07 and Town10 (Figure 5). They cover different network sizes, road widths, and intersection layouts. Each network is represented as a continuous drivable region without lane divisions, prescribed travel directions, traffic signals, or right-of-way rules. All traffic participants are autonomous vehicles. The geometry and sampling settings below apply to the resumed eight-map training phase in Table 3. The method and reward ablations use the earlier map versions and sampling protocols specified for those experiments.

[2] https://github.com/Emerge-Lab/PufferDrive

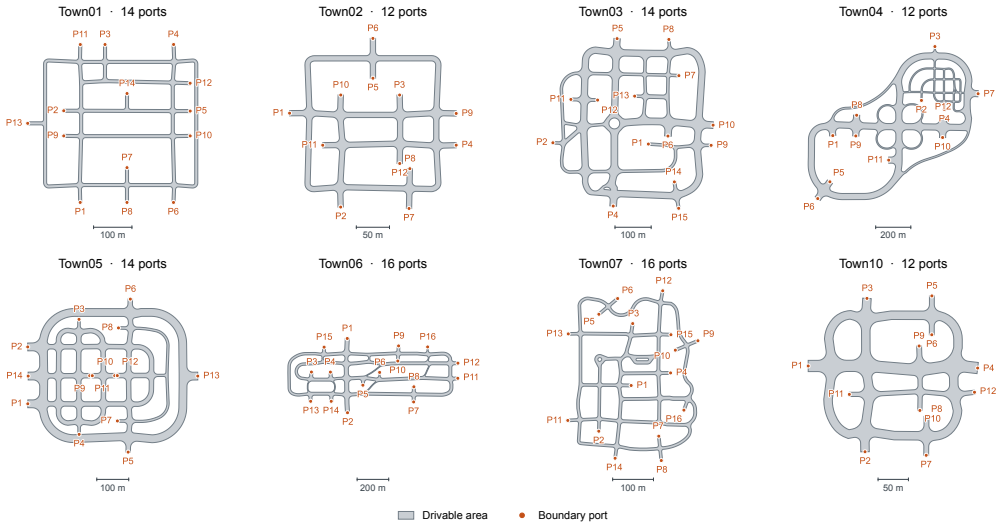

Figure 5: The eight experimental road networks. Gray regions show drivable road surfaces, and orange markers identify boundary ports, whose roles are sampled at each episode reset. Each panel uses its own scale, indicated by the scale bar.

The updated maps have 110 boundary ports in total: 14, 12, 14, 12, 14, 16, 16, and 12 for Town01–Town07 and Town10, respectively. At each episode reset, all ports are enabled and randomly divided into equally sized sets of *sources*, where vehicles enter, and *sinks*, where completed trips leave. Each source must have a legal outgoing source–sink route and each sink a legal incoming route. There are no inactive ports in this configuration; earlier ablations retain their original protocols, which allowed inactive ports. An episode starts with a small population placed along sampled routes. New arrival requests then follow a Poisson process. Vehicles spawn at their assigned sources with a randomly sampled lateral position, an initial speed of 5 m/s, and a heading aligned with the local route tangent. If there is insufficient space to spawn safely, the request waits and is retried in later steps.

B.4 Route Catalogue and Sampling

The updated offline catalogue contains 22,775,950 routes connecting ordered pairs of boundary ports across the eight maps. Routes do not revisit graph nodes or pass through other boundary ports. A training run uses a fixed, finite resident subset of this catalogue. We first retain at least one route for every legal ordered source–destination pair (OD), then distribute the remaining capacity across ODs, targeting up to 16 base routes per OD. Routes are sampled uniformly without replacement within each eligible OD. The cap is 4096 compiled routes per map, including two truncated variants per base route used for initial vehicle placement, so at most 1365 base routes fit under this cap. The cap or available routes can therefore reduce an OD's allocation below 16. Each run samples trips from this fixed subset of the offline catalogue.

Before selecting this pool, the Town04 experiment excludes routes containing either consecutive node sequence `node_0023`→`node_0025`→`node_0026` or its reverse. This exclusion applies to route selection while the drivable surface remains unchanged; the offline total above precedes the exclusion. Every legal OD must still retain at least one eligible route. For each trip, we sample a valid source–sink pair with probability proportional to the product of the two ports' effective widths, and then uniformly select a base route from the resident pool for that pair.

B.5 Traffic Density and Inflow

We use six nominal densities, $\rho^* \in \{1, 2, 3, 4.5, 6, 8\}$ vehicles per $1000\,\mathrm{m}^2$. For a map with core drivable area A_{core} in square meters, excluding the boundary access roads, we set the nominal

vehicle population and total arrival rate as

$$N^* = \frac{\rho^* A_{\text{core}}}{1000}, \qquad \lambda = \frac{N^*}{T_{\text{free}}}. \tag{4}$$

Here, T_{free} is the map's mean reference travel time, computed using the same port-width weighting and uniform route sampling. The reference vehicle starts at 5 m/s, accelerates at $3\,\mathrm{m/s^2}$ to 12.5 m/s, and then cruises to its destination. This converts each density level into a map-specific inflow in vehicles per second. Actual occupancy varies as vehicles enter, interact, and leave.

C Training Details

The S3 method ablation and the eight-map policy share the AutoZero architecture, losses, and auxiliary update schedule, with other training settings summarized in Table 3. Both use reward B, BF16 mixed precision, 256-step rollouts, up to three PPO epochs, clip parameter 0.2, entropy coefficient 0.002, $\gamma = 0.999$, and GAE $\lambda = 0.98$. The base learning rates are 10^{-4} for the policy and shared features and 6×10^{-4} for the ordinary value head and auxiliary optimizer.

Both configurations use the current rollout for auxiliary learning after every policy phase, with KL weight $\beta = 4$. The S3 AutoZero configuration uses three auxiliary epochs, fixed learning rates, and no advantage filtering. The separate eight-map training run supplies the policies used in Sections 4.1 and 4.2.2. Its resumed training phase uses at most three auxiliary epochs. After each auxiliary epoch, the full-cache weighted KL to the fixed post-PPO reference must not exceed 0.01; otherwise that epoch's model and auxiliary optimizer updates are rolled back and the auxiliary phase ends.

This stage also filters samples for both optimization phases using absolute unnormalized GAE: the threshold is 0.01 times an exponential moving average of the global maximum absolute advantage, with EMA update coefficient 0.25. If a positive threshold would discard all samples, one global maximum is retained. All base learning rates follow a cosine schedule over 10^{12} cumulative active transitions, with a minimum multiplier of 0.1. Reported transition budgets count collected active samples before filtering, counting each transition once regardless of optimization reuse. The settings above apply to the resumed phase of eight-map training.

Table 3: AutoZero configurations for the S3 ablation and the resumed eight-map training phase used for qualitative and network evaluations. The two experiments use separately trained policies.

Setting	S3 AutoZero	Eight-map resumed stage
Training maps	Town10	Town01–07, Town10
Nominal densities	1, 3, 6	1, 2, 3, 4.5, 6, 8
Training seeds	11, 22, 33	11
Parallel worlds	96 per seed	12,288 total
Minibatch size	16,384 (single GPU)	131,072 per GPU
Auxiliary interval	Every rollout	Every rollout
Auxiliary data	Current rollout	Current rollout
Auxiliary KL weight β	4	4
Auxiliary epochs	3	Up to 3; KL rollback at .01
Advantage filtering	Off	Threshold ratio .01; EMA .25
Learning-rate schedule	Constant	Cosine to $0.1\times$ at 1T

D Method Ablation

Setup and comparisons. We evaluate AutoZero on **S3**: Town10 traffic at nominal densities 1, 3, and 6 vehicles per $1000\,\mathrm{m}^2$. Each method uses 96 training worlds, 24 fixed evaluation worlds, and three training seeds (11, 22, 33). All methods use the same local observations, reward B, and tanh-squashed Gaussian actions. We compare AutoZero with four variants:

- **PPO (SAttn)**: separate actor and critic networks with the same spatial encoder design and PPO training.
- **PPG (Set)**: AutoZero with the attention pathway removed, retaining the set statistics and the same phasic training scheme.

- **Shared-PPO (SAttn)**: a shared actor–critic network trained jointly by PPO policy and value losses.
- **Two-Head PPG (SAttn)**: a shared policy/value network that preserves value predictions during policy updates and fits returns during policy-preserving distillation. For each rollout of N active transitions, it performs $6\lceil N/16384 \rceil$ distillation updates with batches of 2048.

We compare sample efficiency up to 200 million active transitions and time efficiency up to 60 minutes of training. Training time includes initialization, sampling, optimization, auxiliary updates, and checkpoint saving, and excludes evaluation, queueing, and downtime. AutoZero and the PPO (SAttn), PPG (Set), and Shared-PPO (SAttn) baselines are resumed after their 200M runs with rebuilt simulation environments and reset random-number generators; the two-head variant is trained continuously. Success is deterministic goal/offered, averaged equally over the three densities within each seed. Curves and endpoint statistics use linear interpolation between observed checkpoints, followed by the mean and sample standard deviation across seeds.

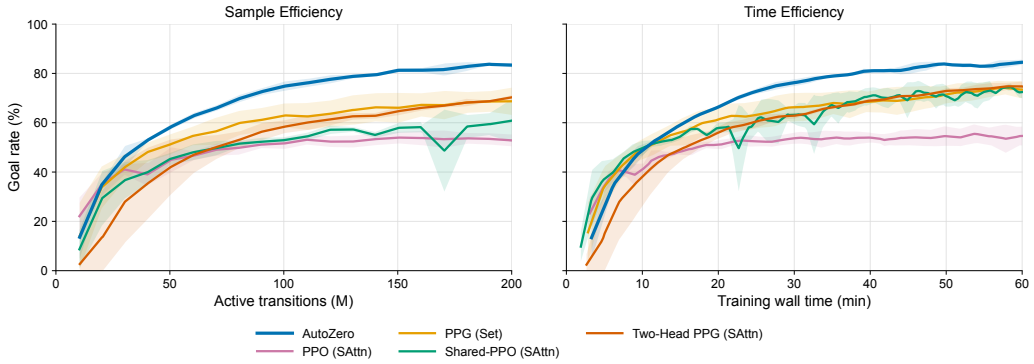

Figure 6: AutoZero and four method variants on S3. Left: success versus active environment transitions. Right: success versus training wall time, including auxiliary optimization. Lines and shaded regions show means and one sample standard deviation across three seeds, with densities 1, 3, and 6 equally weighted. Curves use observed checkpoint intervals without smoothing or extrapolation.

Table 4: S3 success (%) at exactly 200M active transitions and 60 minutes of training, using the same interpolated endpoints as Figure 6. Values are means ± sample standard deviations across three seeds.

Method	200M transitions	60 minutes
AutoZero	83.37 ± 0.90	84.52 ± 1.04
PPO (SAttn)	52.85 ± 2.40	54.64 ± 3.84
PPG (Set)	68.70 ± 5.52	73.44 ± 3.30
Shared-PPO (SAttn)	60.83 ± 8.65	72.44 ± 2.51
Two-Head PPG (SAttn)	70.29 ± 0.89	74.70 ± 1.87

Results. AutoZero achieves the highest mean success under both budgets (Figure 6 and Table 4). At 200M transitions, it reaches 83.37%, compared with 52.85% for PPO (SAttn). Removing spatial attention lowers success to 68.70%, supporting its value for learning interactions among nearby vehicles. Replacing phasic training with joint Shared-PPO (SAttn) lowers success to 60.83%; the two-head alternative reaches 70.29%.

At 60 minutes, AutoZero reaches 84.52%, compared with 74.70% for the strongest alternative, Two-Head PPG (SAttn). This supports combining spatial attention with the three-head phasic training scheme, including its gradient routing and update schedule. High-density coordination remains challenging: AutoZero's density-6 success at 60 minutes is 67.40%, and 27.45% of admitted vehicles terminate in collisions.

E Reward Ablation

E.1 Experimental Protocol

We study how progress rewards, time costs, and control-change penalties affect learning. Using the same terms as Equation 1, we vary their coefficients in

$$r_t = I_{g,t} - I_{c,t} - I_{o,t} + k_p \Delta s_t - c_t \Delta t \\ - c_{\Delta a} \left(\frac{\Delta u_{a,t}}{2} \right)^2 - c_{\Delta \delta} \left(\frac{\Delta u_{\delta,t}}{2} \right)^2 . \tag{5}$$

The progress and time coefficients apply per meter and per second, respectively; control-change coefficients apply per simulation step.

Each configuration is trained separately on three task suites: **S1**, single-vehicle navigation in Town10 with short, medium, and long routes; **S2**, small groups of vehicles passing, merging, or crossing; and **S3**, Town10 traffic at nominal densities 1, 3, and 6 vehicles per $1000\,\mathrm{m}^2$. Every run starts from scratch and uses 100 million active vehicle transitions, with three training seeds (11, 22, 33). All runs use PPO with separate actor and critic networks, each with two 1024-unit hidden layers, BF16 mixed precision, $\gamma = 0.999$, GAE $\lambda = 0.98$, and no advantage filtering.

The first sweep tests $k_p \in \{0.005, 0.010, 0.015, 0.020\}$ and $c_t \in \{0, 0.001, 0.005, 0.010, 0.020\}$, with both control-change coefficients fixed at 0.001, giving 180 training runs. The second sweep fixes $c_t = 0.020$, retains $k_p \in \{0.010, 0.015\}$, and independently varies $c_{\Delta a}, c_{\Delta \delta} \in \{0, 0.001, 0.010, 0.100\}$, giving another 288 runs.

We evaluate deterministic policies on fixed scenarios at five checkpoints spaced across the training budget. Success is the fraction of offered trips that reach their destinations, including unadmitted requests in the denominator. Within each suite, we equally weight route-length groups (S1), interaction types (S2), or densities (S3). Results report the mean and sample standard deviation across three seeds. The three-task average is computed within each seed before aggregation.

E.2 Results and Reward Selection

Progress and time costs. The balance between progress and time costs strongly affects traffic learning (Figure 7a). At $k_p = 0.010$, S3 success is below 0.1% for $c_t \leq 0.001$, but reaches 38.89% at $c_t = 0.020$. Low progress rewards combined with low time costs often produce prolonged stagnation. Increasing progress rewards alone is less effective: at $c_t = 0.020$, raising k_p from 0.010 to 0.020 reduces mean S3 success from 38.89% to 36.92%. S1 final success remains between 94.44% and 95.66% across the first sweep, although learning speed varies.

Control-change penalties. Stronger penalties do not yield consistent gains across tasks (Figure 7b–e and Table 5). Increasing the acceleration-change coefficient from 0.001 (B) to 0.010 (T) improves S3 success from 38.89% to 39.92% and improves its learning curve, but slows S1 learning and lowers final S2 success. Removing both control-change penalties (N) still achieves 38.63% on S3, with slower early learning. A stronger acceleration-change cost of 0.100 (A) gives the highest three-task final average among the tested configurations, mainly through better S2 performance, while its S3 success falls to 37.93%.

Table 5: Representative reward configurations. All use $c_t = 0.020$. Success rates (%) are final deterministic means ± sample standard deviations across three seeds. Labels match Figure 7. Repeated baselines from the two sweeps are reported once.

Config.	k_p	$c_{\Delta a}$	$c_{\Delta \delta}$	S1	S2	S3	Average
G	0.015	0.001	0.001	95.31 ± 0.52	98.61 ± 0.30	37.59 ± 1.12	77.17 ± 0.57
B (main)	0.010	0.001	0.001	94.62 ± 0.60	96.01 ± 2.96	38.89 ± 1.03	76.51 ± 1.07
T	0.010	0.010	0.001	94.44 ± 0.80	94.97 ± 3.70	39.92 ± 0.21	76.44 ± 0.92
A	0.010	0.100	0.001	94.79 ± 0.52	99.65 ± 0.30	37.93 ± 0.61	77.46 ± 0.36
N	0.010	0	0	95.14 ± 0.30	94.10 ± 2.17	38.63 ± 1.20	75.95 ± 1.11

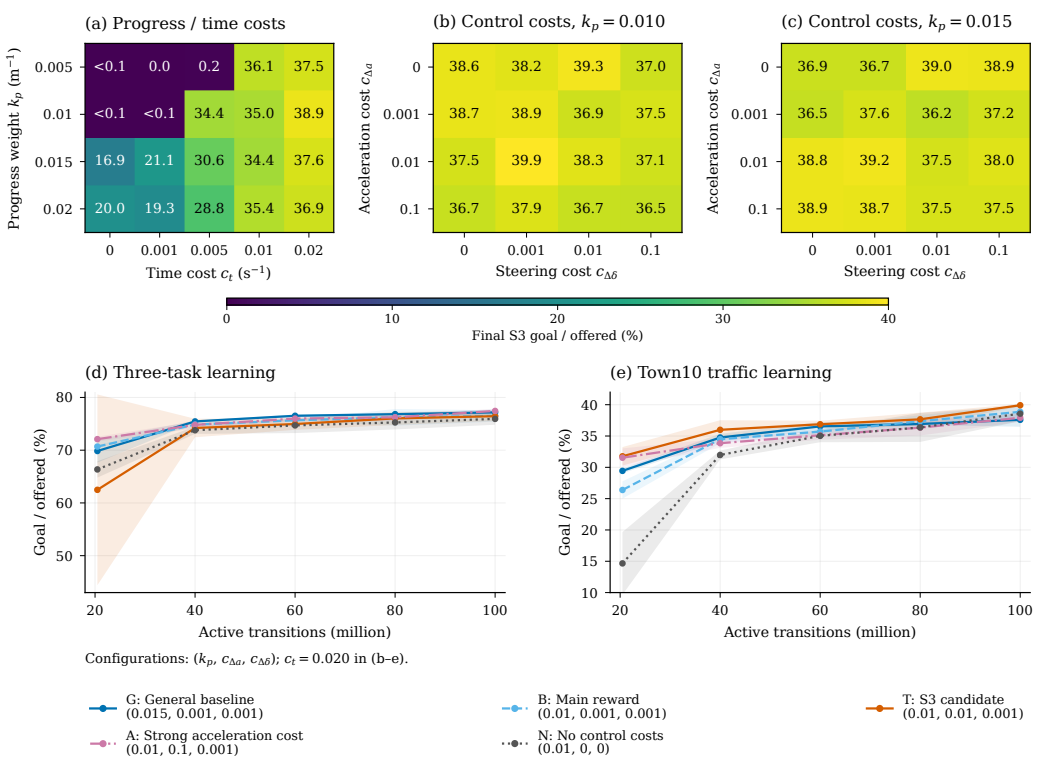

Figure 7: Reward ablation. (a) Final S3 success across progress and time coefficients, with both control-change coefficients fixed at 0.001. (b,c) Control-change sweeps at $k_p = 0.010$ and 0.015, with $c_t = 0.020$. Heatmaps show three-seed means at the final checkpoint and share a 0–40% color scale. (d,e) Learning curves for the representative configurations in Table 5; lines and shading show means and one sample standard deviation. Curves are aligned over their common observed transition range.

Main reward. We retain B for subsequent experiments. Its acceleration-change penalty is ten times smaller than T's, leaving more flexibility for control adjustments during learning in dense traffic. This is a design choice; T has the higher S3 mean in the present sweep. Dense traffic remains challenging: B achieves 63.65%, 32.98%, and 20.04% success at densities 1, 3, and 6, respectively. At density 6, 66.10% of admitted vehicles terminate in collisions. Reward tuning improves traffic learning, while safe coordination at high density remains an open challenge.

F Network Evaluation Details and Additional Results

F.1 Scenario, Controllers, and Evaluation Protocol

Matched network and demand. We compare AutoZero with conventional autonomous traffic on a 4×4 intersection grid, with 150 m between intersection centers and 75 m boundary access roads. Internal roads are 18 m wide: AutoZero uses the continuous road surface, while the conventional baselines have three 3 m lanes in each direction. All methods use 9 m boundary access roads and portals. For each scenario seed, eight of the sixteen portals are uniformly sampled as sources and the remainder become sinks, without per-side balance constraints. Each source generates independent Poisson requests at 600, 1200, or 1800 vehicles/hour (Low, Medium, and High), giving total demands of 4800, 9600, and 14400 vehicles/hour. Destinations are sampled uniformly among sinks. All methods replay identical request times, source–destination pairs, and road-level paths, sampled uniformly among equal-length shortest paths. AutoZero receives road-centerline references; the baselines follow the corresponding directed edges with legal lane changes. Thus routes agree at the road level, without prescribing identical vehicle trajectories.

Vehicles and controllers. All methods use 4.5×1.8 m vehicles, a 20 m/s speed limit, longitudinal acceleration bounds of $[-4, 3]$ m/s^2, an initial speed of 5 m/s, and a 0.1 s simulation step. The baseline, denoted *Actuated*, uses SUMO 1.24.0 with CACC, LC2013 lane changing, fixed travel directions, and actuated signals. Typical internal approaches have a dedicated left-turn lane, a through lane, and a shared through/right-turn lane, with protected left-turn phases. Insertion can use all three inbound lanes; requests that cannot enter safely remain in an external queue in all methods. SUMO departure timeouts and congestion teleportation are disabled.

Actuated parameters are selected from seven candidates on scenario seeds 101 and 202, checked on held-out seeds 303 and 404, and fixed before the reported evaluation. Selection minimizes mean request-to-success time, charging failed or unfinished requests through the evaluation horizon, subject to a non-increasing collision rate. The selected profile uses 9–60 s through greens, 5–30 s pure left-turn greens, a 3 s maximum detection gap, and a 2 s detector-gap setting; yellow intervals are 6 s and existing all-red phases are 1 s. AutoZero uses the fixed evaluation checkpoint from eight-map self-play training, evaluated deterministically without fine-tuning on this grid.

We additionally evaluate *Max-Pressure* with the same SUMO vehicles, lane connections, and admissible green states. Its lane-based pressure sums upstream-minus-downstream stopped-vehicle counts over served lane pairs, with each incoming lane's contribution divided among its legal downstream lanes. Decisions are made every 1 s; ties retain the current phase. Every phase change uses 6 s of yellow followed by 1 s of all-red. A single minimum green of 15 s is selected from 10, 15, 20, and 30 s using seeds $1101/2202$, then validated against 10 s on $3303/4404$, each at all three demands. Selection uses the same full-request cost and collision constraint as above; validation additionally requires non-decreasing mean throughput. These 24 calibration and 12 validation runs are excluded from Table 1.

Our reinforcement-learning signal baseline is *MPLight* (Chen et al., 2020), adapted from LibSignal (Mei et al., 2022) to the same legal phase library and SUMO CACC + LC2013 vehicles. The 16 intersections share a FRAP/DQN controller with incoming-lane vehicle counts and the current phase as input, and negative mean incoming-lane queue length as reward. Decisions occur every 30 s with a 15 s minimum green and the same 6 s yellow/1 s all-red clearance. We train one model per demand for 200 episodes using training seed 11. Among the checkpoints evaluated every 20 episodes across both the initial 100 episodes and the 100-episode continuation, we select the one with lowest mean full-request cost subject to zero observed collisions across the five scenarios. The selected episodes are 100, 60, and 80 for Low, Medium, and High demand, respectively. Final evaluation uses deterministic GPU inference with the same numerical settings as training.

Evaluation protocol. Each run starts empty and consists of 300 s of warm-up, 900 s of measurement, and 600 s of drainage; new requests stop at 1200 s. We evaluate five paired scenario seeds (11, 22, 33, 44, 55) at each demand for four methods, totaling 60 evaluation runs. All methods use identical demand and route sequences. Throughput counts successful exits during $[300, 1200)$ s, including trips requested during warm-up, and is converted to vehicles/hour. Completion, collision, and out-of-bounds rates use all requests generated during $[0, 1200)$ s as their denominator, with outcomes observed through 1800 s. Never-admitted and unfinished requests therefore remain in the denominator. Collision rates count involved vehicles rather than events; failed vehicles are removed and do not count as successful exits. Table 1 reports equally weighted scenario means and sample standard deviations.

F.2 Efficiency and Safety Results

Additional efficiency statistics. At High demand, completion is 98.85% versus 52.26%; the mean external queue during measurement is 0.01 vehicles for AutoZero versus 1747.2 for Actuated. Paired scenario-bootstrap 95% confidence intervals for the throughput differences relative to Actuated are $[268.8, 735.2]$, $[3900.0, 4470.4]$, and $[8338.4, 9311.2]$ vehicles/hour, respectively. Calibrated Max-Pressure achieves mean throughputs of 4513.6, 5188.8, and 5191.2 vehicles/hour. MPLight achieves 4402.4, 5077.6, and 5311.2 vehicles/hour, with completion rates of 98.98%, 71.96%, and 49.03%. Despite its demand-specific training and selection on these scenarios, MPLight does not consistently outperform the calibrated conventional controllers.

Safety and comparison scope. Table 6 reports the safety metrics for the same 18 m network evaluations as Table 1. Collision and out-of-bounds (OOB) rates use all offered requests as their denominator, including never-admitted and unfinished requests, with outcomes observed through the end of the 600 s drainage period. Collisions count involved vehicles rather than events; failed vehicles are removed. All three SUMO baselines record zero collisions and OOB terminations in these runs. SUMO uses native lane-following dynamics and collision checks, rather than OpenDrive's bicycle dynamics and collision detection, and does not enforce AutoZero's ± 0.5 rad steering-actuator bound. These differences remain part of the system-level comparison despite matched vehicle dimensions and longitudinal bounds.

Table 6: Network safety on 18 m roads with 9 m portals. Demand denotes the total arrival rate across eight sources. Values are means $\pm$ sample standard deviations across the same five paired scenario seeds as Table 1; rates (%) use all offered requests. Bold means indicate the lowest rate for each demand and metric, including ties.

Demand	**Low (4,800 veh/h)**		**Medium (9,600 veh/h)**		**High (14,400 veh/h)**	
	Collision ↓	OOB ↓	Collision ↓	OOB ↓	Collision ↓	OOB ↓
Actuated	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00
Max-Pressure	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00
MPLight	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00	**0.00**±0.00
AutoZero	0.10±0.16	0.14±0.09	0.38±0.12	0.25±0.09	0.87±0.19	0.27±0.10

F.3 Demand Stress Protocol and Failure Diagnostics

Demand and evaluation protocol. We extend the network experiment using the same fixed policy, grid geometry, and deterministic FP32 evaluation. Six offered demands, 4800–28800 veh/h in increments of 4800, are each evaluated on five scenario seeds $(11, 22, 33, 44, 55)$, giving 30 AutoZero runs. We additionally run Actuated and Max-Pressure at 19200, 24000, and 28800 veh/h on the identical request sequences (30 additional SUMO runs), and reuse their original three-demand results. Both controllers retain their previously selected parameters and controller definitions, including the 15 s minimum green for Max-Pressure; no high-demand tuning is performed. Each run has 300 s of warm-up, 900 s of measurement, and 600 s of drainage. New requests stop at 1200 s; queued requests continue attempting admission until 1800 s. Successful flow, mean on-road population, and mean external queue use the measurement window. Completion is measured over all requests, while collision and OOB rates use admitted vehicles, with outcomes tracked through the full horizon.

Results and uncertainty. Figure 3(a–d) shows diminishing AutoZero throughput gains alongside increasing failures and external accumulation. Mean successful flow is 14252.8 veh/h at demand 14400, reaches 19255.2 veh/h at 24000, and is 17032.0 veh/h at 28800. The last change has a paired scenario-bootstrap 95% interval of $[-4817.6, 371.2]$ veh/h; thus the point estimates suggest saturation and decline, without establishing a strict decrease or steady-state capacity. From demand 14400 to 28800, collision/admitted increases from 0.87% to 14.83% and OOB/admitted from 0.27% to 1.74%, while completion falls from 98.85% to 67.24%. Mean external queue grows from approximately zero to 1038.3 vehicles. Additional demand therefore does not translate into proportional successful flow.

The added controls in Figure 3(a) show that Actuated and Max-Pressure plateau at substantially lower successful flow under the same offered requests. Across the three high-demand levels, their means remain within 5388.0–5456.8 and 5076.8–5272.0 veh/h, respectively. At 28800 veh/h, AutoZero achieves 17032.0 veh/h versus 5456.8 for Actuated and 5104.0 for Max-Pressure, while the baselines' mean external queues reach 4708.4 and 4745.8 vehicles. This establishes a higher observed successful flow in the tested setting, despite AutoZero's deterioration under stress. Actuated records no collision terminations in the additional runs, and Max-Pressure records two vehicles in one highest-demand scenario during drainage (mean collision/admitted 0.01465%).

Persistent stagnation and failure cases. We distinguish sustained lack of progress from temporary stopping. At 1 s sampling intervals, a qualifying 30 s window requires speed below 0.5 m/s for at least 90% of intervals and a route-progress range below 2 m. Overlapping qualifying windows are merged; their overlap with the measurement window, summed over vehicles and divided by total on-road vehicle-time, defines Figure 3(d). The fraction is zero at the three lowest demands and 23.38% at 28800 veh/h; it is not monotonic between the two intermediate stress levels. Snapshot colors instead require a complete qualifying window in the preceding 30 s, without using future states.

The two collision examples illustrate distinct local geometries: nearly aligned vehicles with lateral offset in (e), and intersecting motions in (f). In the episode shown in (g–h), a local cluster is followed by extensive stationary queues: 11 vehicles in the zoom qualify at 660 s, while all 350 on-road vehicles qualify at 1600 s and 708 requests remain outside. A separate check of every 0.1 s frame confirms that these 350 vehicles keep identical positions, headings, and route progress, with zero speed, throughout the final 300 s; no successful exits occur in that interval. We classify this observed outcome as persistent policy stagnation. Identifying its causal onset and feasible escape maneuvers would require controlled interventions.

Failure-case visualization. Demand stress and observed failure modes of the AutoZero policy. (a) Successful flow for AutoZero, Actuated, and Max-Pressure versus offered demand; the dashed line is offered demand. (b–d) AutoZero failure rates, mean occupancy and external queue, and persistent-stagnation vehicle-time fraction. Lines are means over five scenario seeds; shading is the scenario-bootstrap 95% confidence interval. (e–f) Similar-heading and crossing collisions at 28800 veh/h (seeds 11 and 22). Recorded poses precede native collision events by 0.1 s; physical-width ribbons show the preceding 3 s, fading toward the past. Colored vehicles are the collision participants; gray vehicles provide context, and white marks indicate heading. (g–h) Local and later

network states of the same 19200 veh/h, seed-11 episode. The dashed box locates (g). Red vehicles satisfy the preceding-30-s stagnation criterion; external queues are not drawn on the road.

G Intersection Evaluation Details

G.1 Scenario and Controller Adaptations

Scenario and pairing. We evaluate 12, 16, and 24 vehicles at an unsignalized four-way intersection with 18 m wide roads, 150 m arms beyond the 12 m junction cutback, and the same OpenDrive bicycle dynamics and termination checks for all methods. Each approach contains an equal number of vehicles at $t = 0$; there are no subsequent arrivals. Straight, left-turn, and right-turn demands are sampled with equal probability. Initial longitudinal distances from the intersection center are randomized within $[65, 145]$ m, with at least 10 m between consecutive vehicle centers on each approach. AutoZero and MPC share identical initial states, with lateral positions randomized across the full road width (center offsets in $[-7.8, 7.8]$ m), and use full-width exits. FIFO-KATS and OBS-KATS share lane-center starts on three 3 m inbound lanes and directed outbound lanes. Across these two road organizations, each vehicle retains its origin, destination, longitudinal position, heading, and initial speed. All vehicles start at 5 m/s, with a 20 m/s speed limit, acceleration bounds $[-4, 3]$ m/s^2, and steering bounds ± 0.5 rad; the simulation step is 0.1 s and the horizon is 90 s.

Controllers. AutoZero uses deterministic actions from the selected policy in Section 4.2.2, without intersection fine-tuning. We implement FIFO-KATS and OBS-KATS following Yan et al. (2024), and adapt the lane-free MPC formulation of Naderi et al. (2025) to full-width initial states and exits. The following describes their adaptations to OpenDrive. The KATS controllers share fixed geometric paths and a reservation-based trajectory planner. For nearby starts, we cap each vehicle's constant crossing speed by its available braking and reacceleration distance and rescale its subzone reservation durations accordingly. Both replan every 2 s; OBS searches at most 64 orders and 4096 nodes. Simultaneous arrivals use a fixed tie-breaking order while preserving within-lane precedence. MPC jointly plans vehicle controls with a 6 s horizon, 0.2 s control knots, 0.4 s replanning interval, and at most 60 solver iterations. Soft guides use smooth lateral transitions and are checked against the road, full vehicle body, and steering bounds; infeasible turn cutbacks are shortened geometrically. Passing initializations retain the original inlet pose after a turn. These guides are soft objectives, not lane constraints. Its straight/left/right preferred speeds are $12/10/8$ m/s; KATS crossing speeds also depend on route curvature and reachability. Thus, common physical bounds do not imply identical speed preferences. The reported times are simulation times and exclude solver wall-clock latency.

Evaluation protocol. We evaluate ten scenario seeds (111, 122, 133, 144, 155, 211, 222, 233, 244, 255) per size, yielding 30 scenarios and 520 vehicles per method. All MPC runs use the controller and geometric guides described above. Table 1 reports mean travel time from initialization to successful arrival and mean cumulative stopped time per vehicle, defined by speed below 0.5 m/s and including vehicles with zero stopped time. Each cell averages a scenario-level metric over ten seeds, with sample standard deviation.

All methods complete $120/120$, $160/160$, and $240/240$ vehicles across the ten scenarios at the three respective sizes, with zero collisions and zero OOB terminations.

The results compare controller and road-organization combinations on these finite cohorts. The keyframes below visualize a matched scenario for all four methods using the MPC version in the table.

G.2 Interaction Keyframes

Figure 8 illustrates scenario seed 211 at the largest cohort size (24 vehicles). All four runs complete $24/24$ trips with zero collisions and zero OOB terminations. We use a common spatial view and an explicit temporal selection rule to show each method's busy crossing period, with timestamps reported separately for each method.

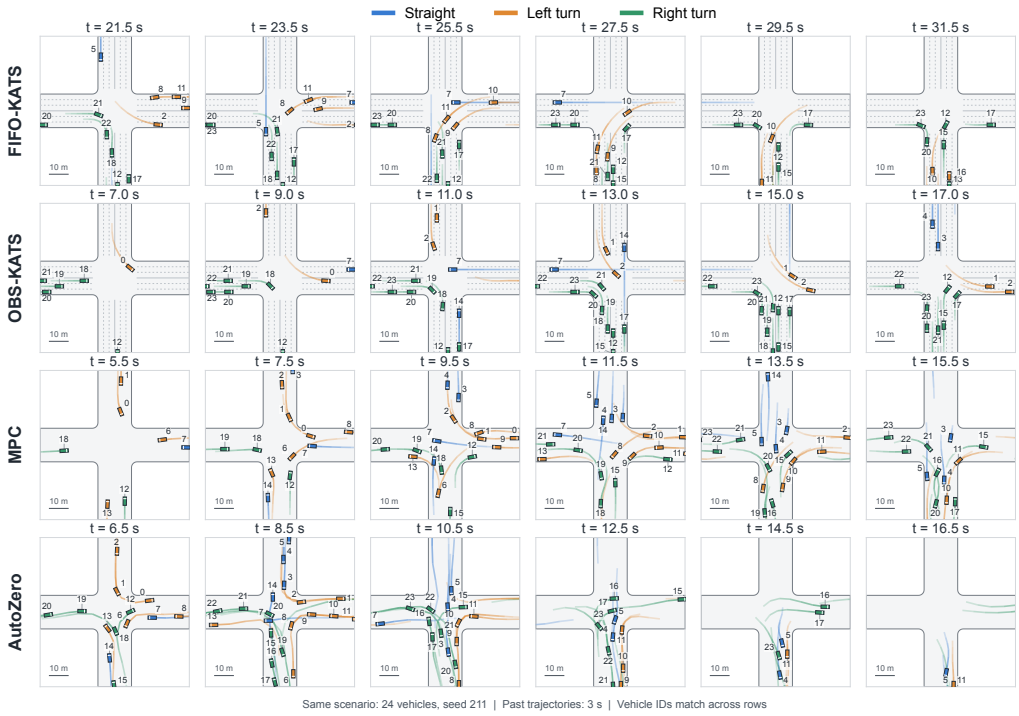

Figure 8: Four-method interaction snapshots in the intersection core. Rows show FIFO-KATS, OBS-KATS, MPC, and AutoZero; each contains six recorded frames spaced 2 s apart. Every panel covers the same 80×80 m view centered on the junction. For each method, the third frame is the earliest recorded maximum of vehicle-center occupancy in the central 40×40 m square; the other frames use offsets $-4, -2, +2, +4, +6$ s. Timestamps are simulation time from initialization. Colors encode straight, left-turn, and right-turn demand; numeric IDs correspond across methods. Fading lines show the preceding 3 s of sampled trajectories, with states saved every 0.5 s and no state interpolation. White front marks indicate vehicle heading. Faint lane guides appear only for KATS. AutoZero/MPC share full-width initial states, and FIFO/OBS share lane-center initial states; origins, destinations, longitudinal placement, initial speeds, and headings are paired across road organizations. This is a selected local example; aggregate efficiency is reported in Table 1.

H Behavior Quantification Protocol and Additional Results

Representative interactions and recording. The representative examples in Figure 2 use deterministic actions from an earlier stage of the eight-map training run, separate from the S3 ablation. Requests enter for 60 s before drainage; the merging example occurs during drainage. The rows use Town05, Town10, Town10, Town06, Town04, and Town05, respectively, all at nominal density $\rho^* = 8$. Each row uses a fixed viewpoint; times below the six snapshots are in seconds. Colors consistently identify two to four representative vehicles in the snapshots and the speed, x-position, and y-position curves; other vehicles are gray. Thin gray lines mark road boundaries. Vehicle-width ribbons connect recorded past positions and fade toward earlier times. Colored and gray vehicles use the same history window: 1.6 s for narrow-road passing and 3.2 s otherwise. Gray ribbons have lower opacity. Curve dots mark the displayed frames. Positions use the crop center as a shared fixed origin, with x rightward and y upward. This common origin preserves the relative positions of vehicles within each row. Origins differ between rows. Positions and speeds are recorded every 0.4 s; ribbons show recorded history. The plots are generated from recorded vehicle states and road geometry.

Recording cohorts. The quantitative behavior figures use the 350.230B checkpoint, with BF16 model inference and TF32 enabled to match the illustrative behavior-recording protocol. The network comparison uses the same weights with FP32 inference and TF32 disabled. The density-8 candidate cohort contains eight maps $\times$ twelve fixed scenarios at density 8 under deterministic actions. Its 96 recordings cover 0–120 s at 0.4 s intervals, with inflow ending at 60 s, and contain 396,995 admitted pair opportunities. The maps were used during training. Scenario specifications, demands, simulator and model definitions, and detector thresholds match the earlier recording cohort; only the model weights differ. Different weights can change realized interactions and the opportunity denominators. The main representative figure and additional candidate collections use an earlier training stage and are analyzed separately from the quantitative cohort. All quantitative distributions below use the same 96 deterministic recordings; the cross-density analysis in Figure 11 is specified separately below. The analysis measures scenario repeatability on training maps for one trained policy.

Scope of behavioral measures. Behavior names denote motion patterns defined by the criteria below. Their precision and recall have not been assessed against independent human annotations. The measures characterize observed interactions; identifying causal yielding responses or assessing platoon string stability requires additional controlled experiments. This interpretation applies to both the quantitative summaries and the illustrative cases in Appendix I.

Opportunities and observation windows. At each saved state, vehicle pairs within 30 m are tested using current positions, headings, and speeds; at least one vehicle must move at 2 m/s or faster. Crossing opportunities require headings separated by 35–145 degrees, both speeds at least 2 m/s, and a straight-line intersection reached by both vehicles within 6 s with an arrival-time difference at most 3 s. Opposing opportunities require heading separation at least 150 degrees, each vehicle ahead of the other, and predicted closest approach within 6 s and 6 m. Converging opportunities use a heading separation below 145 degrees, lateral separation at least 3.5 m in at least one vehicle frame, and the same closest-approach bounds. Following opportunities require heading separation at most 25 degrees, both speeds at least 2 m/s, a nonoverlapping longitudinal ordering, and lateral separation no larger than half the summed widths plus 1.5 m in either vehicle frame. Tests have crossing/opposing/converging/following precedence, so each opportunity has one admission family.

The first eligible state admits a pair encounter without requiring a successful future passage. Proximity gaps of up to 0.8 s are bridged; each uninterrupted encounter is admitted once and observation windows for the same pair do not overlap. Labels inspect up to 12 s after admission and 3.2 s of preceding context. Labels supported only before admission are excluded; following must persist for at least 2 s after admission. Recorded collisions, out-of-bounds terminations, early departures, and recording cutoffs do not remove opportunities. Failure means either focal vehicle has a recorded terminal failure in the window, not necessarily that the two focal vehicles collided with each other. A window admitted during inflow can extend into drainage.

Table 7: Operational candidate definitions. Names match the figures; these are descriptive proxies with unmeasured detection error.

Candidate	Recorded evidence and opportunity denominator
Crossing	Intersecting center trajectories at 35–145 degrees, estimated center-passage gap at most 2.4 s, and minimum center distance at most 12 m. Denominator: crossing opportunities.
Yielding	A crossing whose later vehicle slows by at least 2 m/s before the earlier passage and recovers at least 1 m/s afterward; passage separation is at least 0.8 s. Denominator: crossing opportunities.
Waiting & Cut-in	Merging evidence below, preceded by at least 0.8 s at speed at most 0.5 m/s for either vehicle. Denominator: crossing or converging opportunities; three-vehicle insertion is not checked.
Narrow-road passing	Opposing headings before an interior closest approach within 12 m, a change from ahead to behind, and subsequent separation. Denominator: opposing opportunities; no road-width filter.
Merging	Following after incoming paths are separated by at least 3.5 m, with lateral convergence of at least 2.5 m or prior relative heading at least 35 degrees. Denominator: crossing or converging opportunities.
Platooning	Aligned moving vehicles maintain a stable front/back ordering for at least 2 s, within 30 m and the lateral tolerance above. Denominator: following opportunities; nearest-neighbor adjacency and string stability are not required.

Aggregation and uncertainty. For behavior b and scenario s, let O_{bs} be the number of opportunities from its eligible admission families and C_{bs} the number with the candidate label. The reported fraction is the mean of C_{bs}/O_{bs} over scenarios with $O_{bs} > 0$, with 2000 scenario-level bootstrap resamples and percentile 95% intervals. Resampling at the scenario level preserves dependence among vehicle pairs and frames within each run. In contrast, scene occurrence is the number of scenarios with $C_{bs} > 0$ divided by all twelve scenarios, including those without opportunities, with a Wilson 95% interval (Figure 10). Labels can overlap, and a different admission family can lead to similar later motion; an illustrative case is therefore not automatically a member of a particular fraction's numerator. Bootstrap uncertainty reflects scenario sampling within this fixed protocol, not model-training variability or detector bias.

Candidate fractions and cross-scenario recurrence. Figure 9 reports candidate/opportunity ratios computed within each scenario and then averaged with equal scenario weight. During inflow, Crossing, Yielding, opposing passing, Merging, and Platooning candidates each occur in all 12 scenarios of every map. Their map-wise mean fractions range from 17.29–26.89%, 1.88–5.58%, 48.65–60.87%, 7.11–18.59%, and 37.30–54.83%, respectively. Waiting & Cut-in is less widespread: its inflow scene counts are $(0, 5, 3, 2, 0, 0, 1, 11)$ out of 12 for Town01, 02, 03, 04, 05, 06, 07, and 10, respectively. Even in Town10 its mean opportunity-conditioned fraction is only 0.325%. Thus repeated occurrence and interaction frequency answer different questions. The labels are not mutually exclusive and use different opportunity families, so the percentages cannot be summed or interpreted as preferences among six alternative actions. Scene-occurrence intervals are shown in Figure 10; additional examples are in Appendix I.

Cross-density lateral distribution. Figure 11 uses all six nominal densities, with 12 deterministic scenarios per map–density cell (576 original recordings and 576 paired mirrored recordings). The original cohort includes the same 96 density-8 scenarios, rather than 96 additional independent repetitions. The original group retains 356,017 vehicle-time samples during $0 \leq t < 60$ s, measured every 2.0 s, with speed at least 2 m/s, distance at least 25 m from intersection nodes, and heading change at most 10 degrees across the adjacent saved frames. The same filters retain 347,921 samples in the mirrored group; all 576 pairs have eligible samples. If d_L, d_R are distances from the vehicle center to the first boundary along its left and right heading normals, respectively, we use $u = (d_L - d_R)/(d_L + d_R)$: negative values denote the vehicle-relative left side. Forty equal-width bins on $[-1, 1]$ are normalized within each scenario, averaged within each map–density cell, and then averaged equally across the 48 cells. Bin-center densities are joined without additional smoothing. The left-half statistic is computed directly from samples with $u < 0$, using the same weights. Original condition means range from 68.48% to 97.27%. The original aggregate

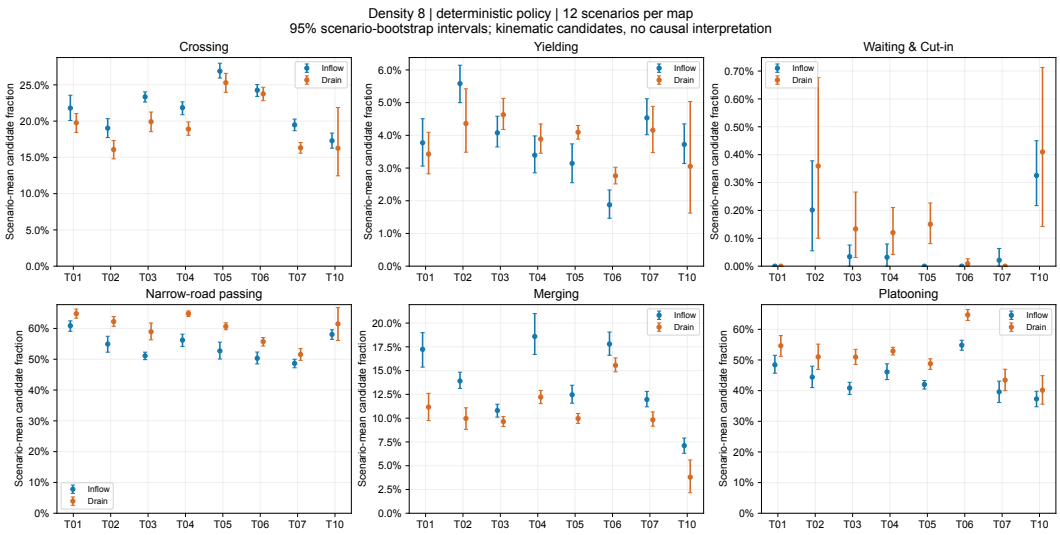

Figure 9: Opportunity-conditioned candidate fractions for the 350.230B policy, using 12 scenarios per map at density 8. Points are equally weighted means of within-scenario ratios among scenarios with eligible opportunities; bars are 95% percentile intervals from 2000 scenario bootstrap resamples. Blue/orange denote inflow/drainage, assigned by opportunity admission time; an observation window may cross the 60 s boundary. Panel scales differ. Names match the representative figure, but the automated proxies are broader: Narrow-road passing is not filtered by road width, Waiting & Cut-in checks a two-vehicle waiting/convergence sequence, and Platooning checks short-term aligned following. Fractions use the motion-based definitions in the text.

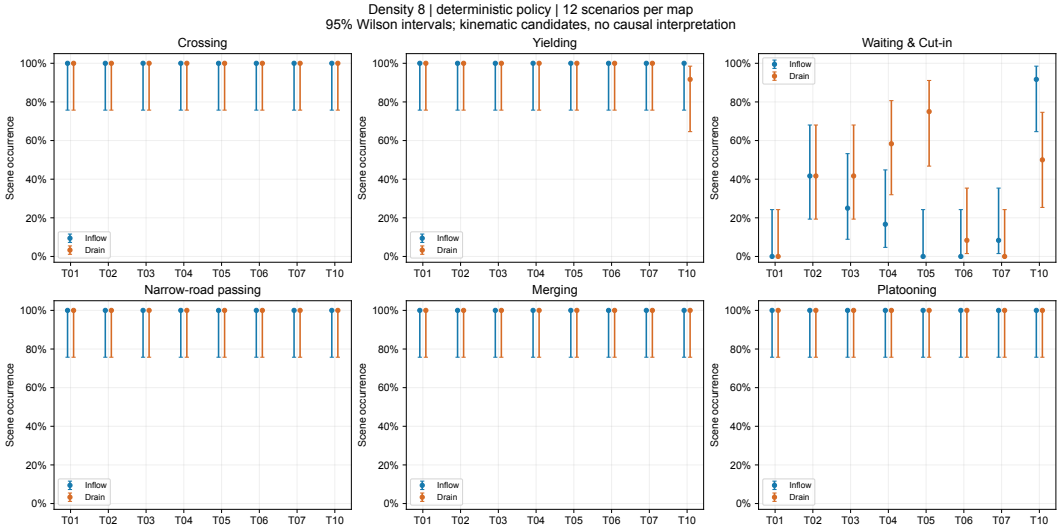

Figure 10: Cross-scenario recurrence in the 96-recording cohort: fraction of twelve scenarios per map with at least one eligible candidate, with Wilson 95% intervals. Blue/orange are inflow/drainage. A value of one means all twelve observed scenarios contain a candidate. Waiting & Cut-in has substantially less uniform map coverage than the other proxies. This occurrence statistic complements the opportunity-conditioned fractions in Figure 9.

95% percentile interval is $[81.61\%, 82.28\%]$, obtained from 5000 scene-bootstrap resamples within each fixed cell; this interval is not drawn in the figure. The shaded area only fills the distribution. This uncertainty excludes training-seed variation and does not remove geometric or route-selection effects.

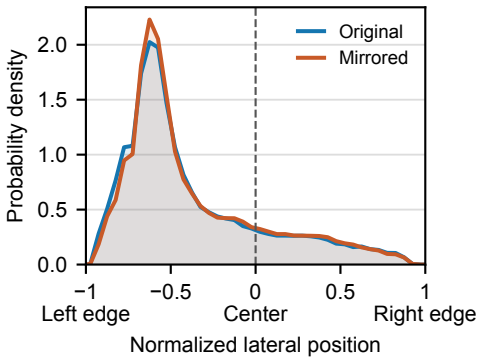

Figure 11: Lateral position on original and mirrored maps, with equal scenario and map–density weighting. Negative positions indicate the vehicle-relative left side; shading fills the distributions.

Paired reflection control. We evaluate the fixed policy in a reflected environment using the coordinate map $M(x, y, \theta) = (x, -y, -\theta)$. Reflected observations are supplied to the policy, and its steering commands are transformed back to the simulator coordinates. Each pair shares routes, arrival times, and initial-placement randomness, while subsequent admissions and outcomes evolve through closed-loop interaction. The reflection procedure was validated against independently constructed mirrored geometry over 40 steps, with observation error below 9×10^{-7} and matching positions, rewards, and termination states. All 576 initial states satisfy the reflection mapping. Repeated evaluations of 192 original scenarios confirmed trajectory and outcome reproducibility.

Mirrored condition means range from 66.77% to 97.70% left-half occupancy. The paired aggregate change is -0.68 percentage points (95% interval $[-1.20, -0.15]$), using 5000 bootstrap resamples of scenario pairs within each fixed map–density cell. These intervals are not drawn in Figure 11. Completion, collision, and out-of-bounds rates use the full 60 s inflow plus 240 s drainage, with offered demand as denominator and the same scenario/cell weighting: original versus mirrored rates are 93.10% versus 82.57%, 2.83% versus 4.41%, and 1.04% versus 9.64%, respectively. Performance degradation is concentrated in Town03, Town06, and Town07. Lateral occupancy is conditional on eligible moving samples, whose composition can change with failures. The results show persistent lateral preference alongside reduced completion and increased failures. Their scope is one trained policy evaluated on reflections of its training geometries; the origin of the preference remains an open question.

Lateral occupancy. For the density-8 view in Figure 12, states are sampled every six saved frames (2.4 s) and restricted to vehicles moving at least 2 m/s, at least 25 m from intersection nodes, and changing heading by no more than 10 degrees between adjacent saved states around the sample. Rays along the heading normals measure the first road-boundary distances d_R, d_L. The normalized position $z = (d_R - d_L)/(d_R + d_L)$ is relative to each vehicle's direction: negative values indicate proximity to its right boundary and positive values to its left. This coordinate has the opposite sign to u in Figure 11. These filtered samples form the lateral-position distributions in Figure 12.

Figure 12 pools the filtered inflow vehicle-time samples within each map, distinguishing central from side-biased occupancy. It provides a complementary spatial description of the same cohort used in the preceding interaction summaries.

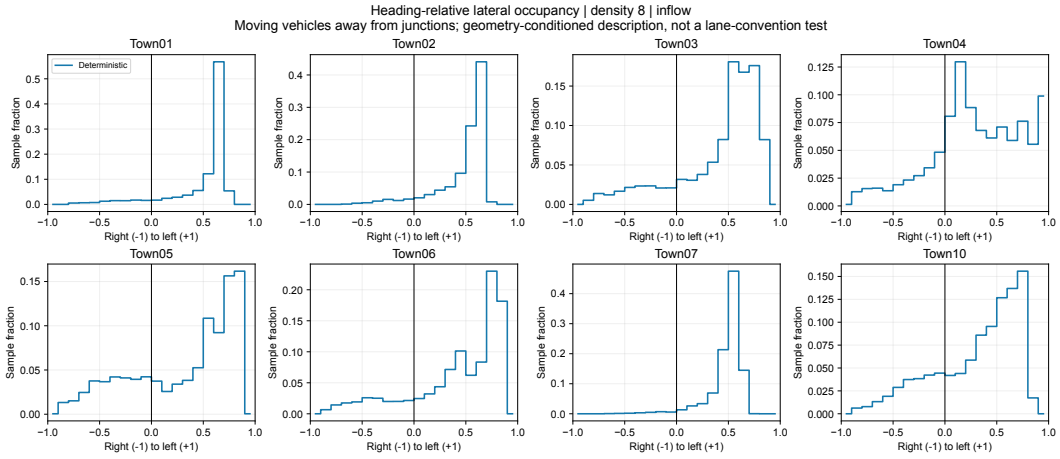

Figure 12: Heading-relative lateral occupancy during inflow for the 350.230B policy, pooling the twelve deterministic density-8 recordings per map. Negative/positive values indicate proximity to the vehicle-relative right/left road boundary; zero indicates equal boundary distances. Histograms pool filtered vehicle-time samples within each map.

I Additional Behavior Candidates

The following collections show two illustrative candidates per behavior, each with six recorded frames. They use an earlier policy than the 350.230B quantitative cohort. Headers identify map, nominal density, and vehicle IDs; these examples include densities 4.5 and 8 and multiple recording cohorts. Examples are selected to illustrate the motion patterns defined in Appendix H and contain no recorded focal-vehicle failure within the event window. In both checkpoints' analyses, unshown failures remain in their respective opportunity statistics. A displayed interval can cover only part of the full event used by the detector.

Blue and orange identify vehicles A and B and their speed curves. The rightmost panel instead plots B's position relative to A: solid dark lines are longitudinal (positive ahead), and dashed purple lines are lateral (positive to A's left). This frame rotates with A's instantaneous heading, so turning contributes to changes in the curves. It differs from the fixed crop-centered x, y coordinates in Figure 2; crossing zero in one component does not establish collision or overtaking. Curve time limits match the displayed frames, and all frames are saved states.

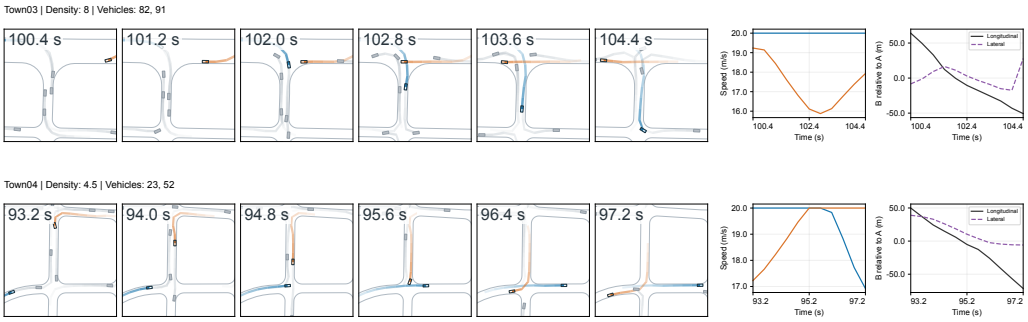

Figure 13: Additional Crossing candidates. The six-frame sequences show differently directed paths traversing a shared region. Relative position and speed show how the vehicles pass through the intersection in sequence.

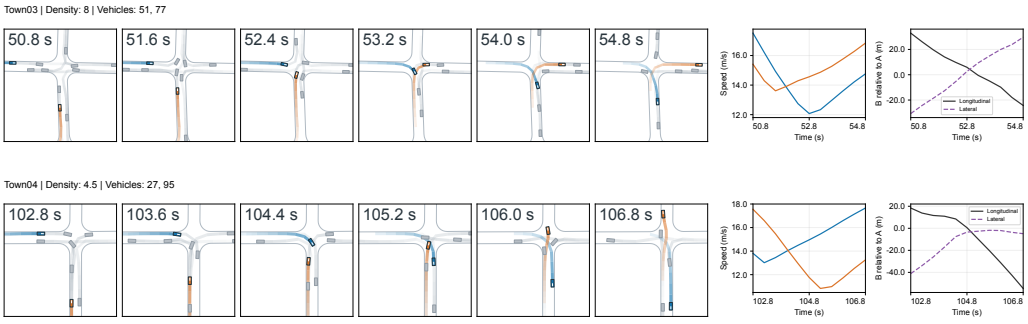

Figure 14: Additional Yielding candidates, selected for slowing before another vehicle's passage followed by recovery within the full event. Speed and relative-position traces show the timing of slowing and recovery.

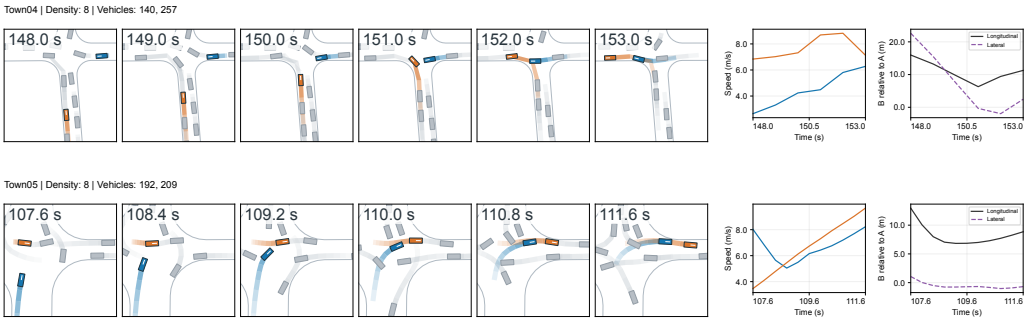

Figure 15: Additional Waiting & Cut-in candidates. The full events contain a low-speed wait followed by convergence and following. The snapshots show a selected portion of this two-vehicle interaction.

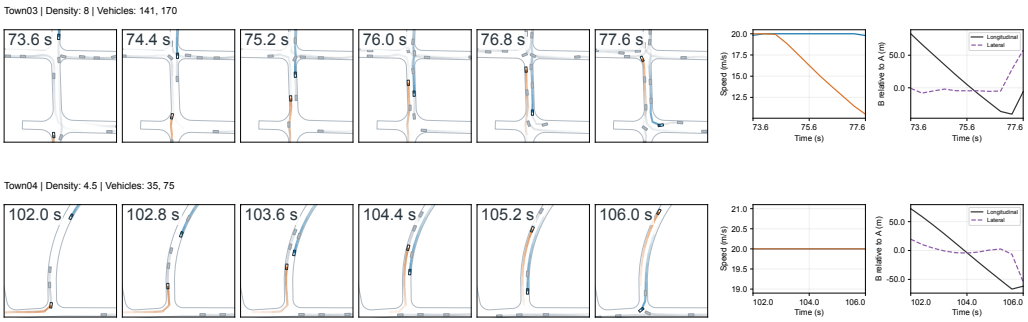

Figure 16: Additional opposing-passing candidates, displayed under the shared Narrow-road passing label. The detector checks relative heading, approach, and separation, without conditioning on road width.

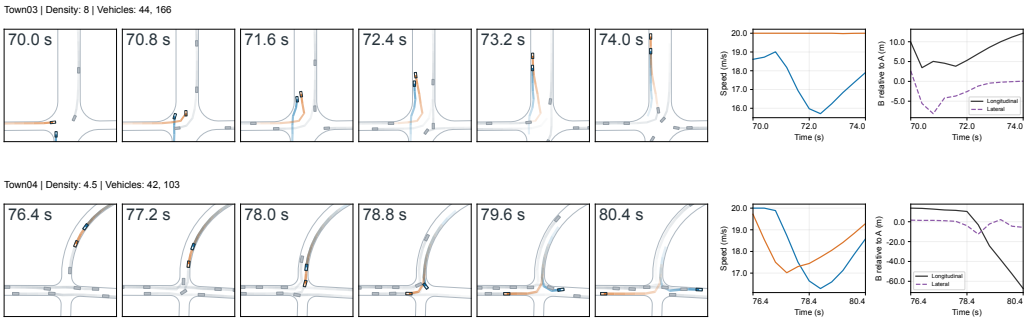

Figure 17: Additional Merging candidates. Initially separated incoming trajectories subsequently align in a following configuration. Relative lateral convergence supports this description but can also reflect rotation of the reference vehicle's heading.

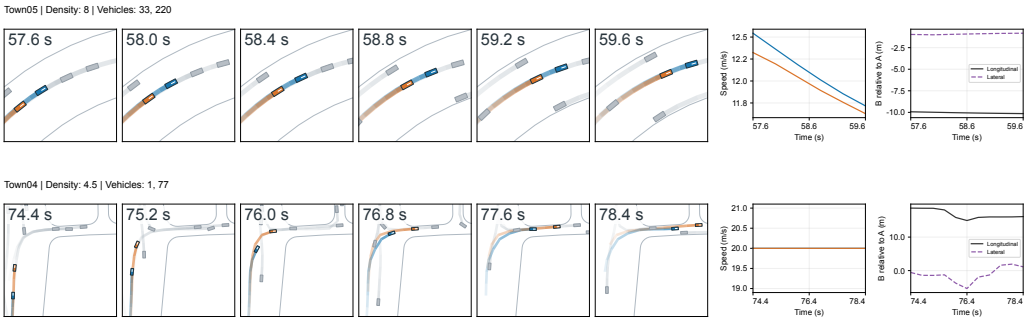

Figure 18: Additional Platooning candidates showing short-term aligned following. The first example uses Town05 vehicles 33 and 220 at 57.6–59.6 s; recorded trajectories through 66 s maintain the same branch and vehicle order. It shows the later following stage of an encounter classified as a crossing opportunity and remains in that opportunity group for the quantitative analysis.

J SENSITIVITY TO FAILED-VEHICLE CLEARANCE

Paired evaluation. We evaluate the fixed 350.230B AutoZero policy without retraining under immediate removal (0 s) and 60 s retention of failed vehicles. Both conditions use the 18 m, 4×4 network with 9 m portals and total demand 14,400 veh/h from Section 4.2.2. The five scenario seeds $(11, 22, 33, 44, 55)$ share identical source/sink assignments, arrival requests, destinations, and road-level routes across conditions. Each run uses 300 s warm-up, 900 s measurement, and 600 s drainage, with 0.1 s steps and deterministic FP32 inference.

Failure handling. At the first collision or out-of-bounds (OOB) termination, the request is counted as failed exactly once. In the retention condition, its vehicle polygon is frozen at the detected end-of-step position and heading for 600 steps; OOB vehicles are not moved back onto the road. Failed vehicles receive no policy actions but remain visible to neighbor observations and vehicle LiDAR, and participate in collision and insertion checks. A later impact does not restart an existing obstacle's timer; a newly colliding vehicle receives its own failure and retention timer. Successful vehicles are removed immediately. The intervention applies throughout the episode, with the original observation representation and policy unchanged. Instantaneous stopping is an imposed failure response, rather than a policy braking maneuver or a model of crash dynamics. The 60 s duration is a sensitivity parameter, not an estimate of real clearance time.

Table 8: Sensitivity to failed-vehicle retention at high demand. Values are equally weighted means $\pm$ sample standard deviations over five paired scenarios. Throughput is successful measurement-window exits in veh/h. All other columns are percentages of all offered requests, evaluated through the end of drainage. Unfinished includes vehicles still active and requests still queued; collisions count involved vehicles, not events.

Retention	Throughput	Completion	Collision	OOB	Unfinished
0 s	14252.8 ± 165.8	98.85 ± 0.19	0.87 ± 0.19	0.27 ± 0.10	0.00 ± 0.00
60 s	8350.4 ± 3342.9	65.64 ± 19.70	10.75 ± 3.18	0.99 ± 0.28	22.62 ± 18.57

Results and scope. Retaining failed vehicles reduces mean throughput by 41.41% (ratio of scenario means) and completion by 33.21 percentage points (Table 8). The mean external queue during measurement rises from 0.01 to 305.05 requests. The effect varies substantially across scenarios: retention-condition completion ranges from 35.19% to 83.79% (Table 9). Thus, the evaluated policy's high throughput is sensitive to the immediate-clearance assumption. The paired experiment holds the policy and simulator fixed to measure sensitivity to obstacle persistence. The 0 s condition uses the same five scenarios as the main network evaluation.

Table 9: Individual paired scenarios for Table 8. Throughput is in veh/h and completion is in percent.

	Throughput		Completion	
Seed	0 s	60 s	0 s	60 s
11	14352	3456	98.64	35.19
22	14044	7056	98.92	60.64
33	14248	8364	98.98	66.51
44	14152	11468	99.07	83.79
55	14468	11408	98.66	82.07

K TRAVEL TIME UNDER TRAFFIC-RULE INTERVENTIONS

Controlled comparison. We evaluate individual traffic constraints using the fixed 350.230B AutoZero policy in OpenDrive. Both conditions share policy weights, 499-dimensional observations, neighbor-route sharing, vehicle dynamics, and deterministic FP32 inference. Paired scenarios share demand, routes, and entry geometry, with 0.1 s steps and no retraining or warm-up. Runs end when all requests have entered and no active vehicles remain, or at 120 s. Successful, colliding, and out-of-bounds (OOB) vehicles are removed immediately in both conditions.

Table 10: Travel time under individual traffic-rule interventions. Travel time is the pooled mean over successfully completed vehicles, measured from actual insertion to arrival, in seconds (lower is better). It includes on-road signal waiting but excludes external entry queues. Completion counts include all requested vehicles across the paired scenarios; failures are excluded from the time mean but retained in these denominators.

		Mean travel time (s)		Completed / requested	
Intervention	Pairs	Open	Constrained	Open	Constrained
Signals, four vehicles	20	8.99	24.73	80/80	80/80
Direction, 30:30	5	13.72	14.97	264/300	256/300
Direction, 48:12	5	12.92	13.70	257/300	278/300
Fixed 3 m corridors	5	11.83	12.90	292/300	276/300

Traffic signals. A four-way intersection has 18 m-wide approaches, a 12 m junction radius, and 50 m arms measured from the junction boundary. Four vehicles are present at initialization, one per approach, with no later arrivals. Each independently chooses straight, left, or right with equal probability. We use 20 paired scenarios (seeds 1–20), with identical initial vehicle states within each pair. The constrained condition adds a longitudinal signal supervisor to AutoZero. North–south and east–west phases each have 30 s green, 6 s yellow, and 2 s all-red, giving a 76 s cycle with a scenario-seeded initial offset. All movements share their approach signal; there is no separate protected-left phase. The stop line is 16 m from the center. The supervisor preserves steering and uses bounded braking (at most 4 m/s^2 deceleration) with a 0.5 m stopping margin. Vehicles unable to stop safely at yellow may proceed; vehicles already inside the intersection continue. Signals are not added to the policy observation. Both conditions complete all 80 trips, with mean travel times of 8.99 s without signals and 24.73 s with this fixed signal schedule. No red-light entries are detected in the constrained runs.

Prescribed directions and fixed corridors. The direction experiment compares an open 18 m-wide road against two non-crossable 9 m one-way regions. Each run has 60 requests, split 30:30 or 48:12 between opposing directions. The corridor experiment compares an open 9 m-wide one-way road against three non-crossable 3 m corridors, with 20 requests assigned to each entry. Both experiments use 150 m roads and five paired seeds $(11, 22, 33, 44, 55)$ per setting. All requests queue at $t = 0$; insertion uses the same native safety checks, and blocked requests retry at the next step without being discarded. Both conditions retain identical 12 m entry/exit regions and route references. Completion targets precede the downstream partition, giving a nominal reference trip of 114 m and avoiding an artificial exit funnel. Crossing a constrained region's boundary triggers native OOB termination; the corridor condition prohibits lane changes rather than modeling ordinary dashed lane markings.

Interpretation. Open conditions have shorter mean travel times among completed trips in all four settings (Table 10). The signal result has full completion but depends on the specified, unoptimized timing. Straight-road completion differs: at 48:12, the open condition is faster among successful trips but completes fewer requests (257 versus 278). All non-completions are collision or OOB failures, not timeouts. The fixed-corridor condition also tests adaptation to a restricted operating space: in isolated-vehicle evaluations, the policy left all three 3 m corridors, although prescribed trajectories traversed them successfully. These comparisons hold the policy fixed without retraining. Together, the travel-time reductions support the efficiency potential of open road space, while differences in completion highlight the importance of policy adaptation when traffic constraints change.