TRAJECTORY-CONDITIONED OBJECTIVE FOR MULTI-INTERSECTION TRAFFIC SIGNAL CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-intersection traffic signal control requires balancing feedback relevance with spatiotemporal coverage. Local, region-shared, and global rewards aggregate costs over spatial sets whose contributing vehicles change over time. Motivated by the observation that vehicle trajectories link feedback for the same vehicles over time and across intersections, we propose a Trajectory-Conditioned Objective (TCO). TCO aggregates queueing costs along the realized trajectories of vehicles currently observed at an intersection within a bounded spatiotemporal horizon. Experiments on real-world road networks show that TCO, as a simple plug-and-play objective, consistently improves representative baselines without otherwise changing their settings and remains robust across traffic loads and network scales.

1 INTRODUCTION

Multi-intersection traffic signal control (TSC) requires coordination among controllers whose decisions affect the same vehicles at successive stages of their journeys. Reinforcement learning methods commonly use queue lengths or pressure to construct rewards for these controllers (Chu et al., 2020b; Wei et al., 2019a; Chen et al., 2020). A local queue-based reward accounts for waiting costs on the incoming lanes of one intersection, with a vehicle set that changes over time. The currently observed vehicles may continue queueing there or incur further costs downstream. Following their trajectories links the learning signal to both outcomes.

Regional and global rewards incorporate cross-intersection feedback by aggregating costs over a fixed neighborhood or the entire network (Chu et al., 2020b; Rashid et al., 2018). Broader coverage includes downstream congestion together with variation from other traffic flows and signal decisions. The choice of spatial scope therefore creates a trade-off between feedback relevance and cross-intersection coverage. Temporal extensions further incorporate delayed outcomes; for example, γ-Reward revises current rewards using future feedback from neighboring intersections (Liu et al., 2021). The vehicles contributing to such regional feedback change as traffic enters and leaves the region.

Vehicle trajectories provide a natural path for feedback over time and across intersections (Figure 1). They connect currently observed vehicles to their subsequent queueing costs, whether they remain at the same intersection or move downstream. We therefore propose a Trajectory-Conditioned Objective (TCO) that aggregates discounted queueing costs along their realized trajectories, within a finite temporal horizon and a bounded number of traversed intersections. The resulting surrogate reward unifies within-intersection and cross-intersection feedback for each initially observed vehicle set.

TCO constructs rewards retrospectively from vehicle trajectories recorded during interaction. At the end of each episode, these rewards are computed and stored with the corresponding transitions for training. The construction can be integrated into existing controllers while preserving their model architectures, observations, and policy update procedures. Execution remains decentralized.

The main contributions of this paper are as follows:

- We propose a trajectory-conditioned local objective that aggregates the subsequent queueing costs of currently observed vehicles, within and across intersections, through bounded trajectory tracing.
- In fixed-configuration experiments, TCO reduces mean travel time in all 24 combinations of three controllers and eight traffic settings.

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.

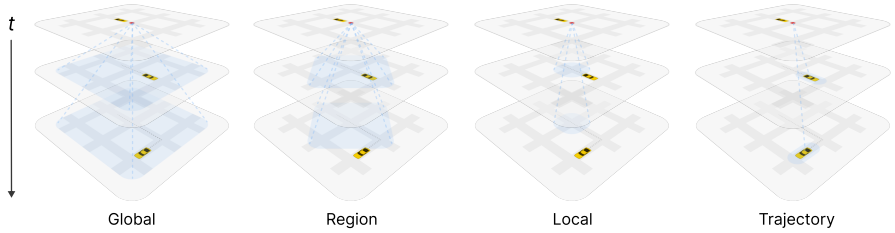


Figure 1: Comparison of four reward constructions. Each subplot shows three consecutive decision steps. Global aggregates current network-wide feedback; Region aggregates current feedback from the target intersection and its neighborhood; Local uses only current local feedback; Trajectory aggregates queueing feedback along the realized paths of vehicles observed at the target intersection.

2 Related Work

Multi-intersection signal control involves coordination with physical delays. RL approaches address it through centralized training, traffic state representations, and reward design.

2.1 CTDE Frameworks

Centralized training with decentralized execution (CTDE) uses joint information during training to learn policies that act on local observations. VDN (Sunehag et al., 2018) decomposes the team action-value function additively, while QMIX (Rashid et al., 2018) combines agent value estimates through a monotonic mixing network. COMA (Foerster et al., 2017) uses a centralized critic and a counterfactual baseline to estimate each agent's advantage. MAAC (Iqbal & Sha, 2019) uses attention to select information from other agents when constructing centralized critics. In traffic signal control, SocialLight (Goel et al., 2023) conditions its critics on neighborhood states and actions and uses counterfactual reasoning for local policy updates. CoordLight (Zhang et al., 2025) combines queue dynamics state encoding with neighbor-aware policy optimization, using attention to model dependencies among neighboring agents in advantage estimation.

2.2 Network Architectures and State Representations

Traffic state representations determine which aspects of traffic flow are available to the controller. Efficient Pressure (EP) (Wu et al., 2021) provides a compact pressure representation, and Advanced Traffic State (ATS) (Zhang et al., 2022b) combines efficient pressure with information about approaching vehicles. AttentionLight (Zhang et al., 2022a) uses queue lengths to represent traffic conditions and self-attention to model correlations between signal phases. Graph-based architectures incorporate dependencies across intersections: CoLight (Wei et al., 2019b) uses graph attention for inter-intersection communication, and STMARL (Wang et al., 2022) models spatial and temporal dependencies in traffic observations.

2.3 Reward Design

PressLight (Wei et al., 2019a) and MPLight (Chen et al., 2020) draw on Max-Pressure control (Varaiya, 2013) to construct local rewards from differences in traffic loads on incoming and outgoing lanes. MA2C (Chu et al., 2020b) and NeurComm (Chu et al., 2020a) incorporate rewards from other intersections with weights discounted by network distance. DenseLight (Lin et al., 2023) defines dense rewards using the gap between ideal and actual vehicle travel distances. Liu et al. (2021) introduce γ-Reward and γ-Attention-Reward, which retrospectively modify rewards using future neighborhood feedback, with attention providing adaptive neighbor weights. TCO uses vehicle identity to select the subsequent queueing costs of vehicles currently observed at an intersection. Their recorded trajectories determine which costs enter the local reward within a bounded window of observations and intersection hops. This construction extends local learning objectives over time and across intersections through the vehicles they serve.

3 PRELIMINARIES

3.1 TRAFFIC TERMINOLOGY

Vehicle Position In a road network with intersection set $\mathcal{I}$, the position p_v^t identifies the downstream intersection of the road occupied by vehicle v. The set of vehicles on the incoming lanes of intersection i is denoted by $\mathcal{V}_i^t$.

Vehicle Trajectory For a vehicle $v \in \mathcal{V}_i^t$, let $\tau_v(t,h)$ be the control-step index of its h-th subsequent recorded observation, with $\tau_v(t,0) = t$. Its trajectory window contains up to H timestamped positions:

$$\mathcal{T}_v^t = \left(\tau_v(t,h),\, p_v^{\tau_v(t,h)}\right)_{h=0}^{H-1}. \tag{1}$$

The window ends after H observations or at the last available record. For consecutive observations, $\tau_v(t,h) = t+h$. When observations are intermittent, $\tau_v(t,h)$ retains the original control-step index.

Vehicle Reward and Intersection Reward The instantaneous reward of vehicle v at time t is

$$u_v^t = -q_v^t, \tag{2}$$

where q_v^t is a binary indicator of whether vehicle v is queued. The intersection reward typically sums the instantaneous rewards of all vehicles on its incoming lanes:

$$r_i^t = \sum_{v \in \mathcal{V}_i^t} u_v^t.$$

This is the Local reward used in our experiments and equals the negative local queue length. We express rewards in units of queued vehicles.

3.2 RL FORMULATION

We model multi-intersection traffic signal control as a decentralized partially observable Markov decision process (Dec-POMDP) (Oliehoek & Amato, 2016), specified by

$$\langle \mathcal{I}, \mathcal{S}, \{\mathcal{A}_i\}, \{\mathcal{O}_i\}, P, \Omega, R, \gamma, \rho_0 \rangle .$$

Here, $\mathcal{I}$ is the set of intersection agents, $\mathcal{S}$ is the global traffic state space, and $\mathcal{A}_i$ and $\mathcal{O}_i$ are the action and observation spaces of agent i. The models P and Ω describe state transitions and joint observations, respectively. The shared reward function R specifies the network-level objective, $\gamma \in [0,1)$ is the discount factor, and ρ_0 is the initial-state distribution.

At the start of control interval t, agent i receives a local observation o_i^t and selects a signal phase a_i^t. The joint action $\mathbf{a}^t = (a_i^t)_{i \in \mathcal{I}}$ induces a transition from s^t to s^{t+1} according to $P(s^{t+1} \mid s^t, \mathbf{a}^t)$. Vehicle positions and rewards are sampled at the end of the interval and indexed by the corresponding transition t. For queue-based control, the shared reward is the sum of the local rewards, $\sum_i r_i^t$. The experiments compare learning signals constructed from these costs, including Local rewards and the trajectory-conditioned surrogate rewards defined below.

4 METHOD

TCO extends local objectives with the subsequent queueing costs of currently observed vehicles.

4.1 TRAJECTORY-CONDITIONED REWARD

Local, regional, and global rewards aggregate vehicle costs over spatial sets of different sizes. For a fixed intersection set $\mathcal{S}_i$, this construction has the form $\sum_{j \in \mathcal{S}_i} r_j^t$. The contributing vehicles change as traffic moves through that set. TCO follows the recorded vehicles along their subsequent paths.

For $v \in \mathcal{V}_i^t$, let $\ell_v^{t,h}$ count the intersection transitions along its trajectory from the initial record at t to the record at $\tau_v(t,h)$. We define the trajectory reward for vehicle v as

$$U_v^t = \sum_{h=0}^{H-1} \gamma_\tau^h u_v^{\tau_v(t,h)} \mathbb{I}\left(\ell_v^{t,h} \le K\right), \qquad H \in \mathbb{N}_+, \tag{3}$$

where the trajectory horizon H bounds the number of recorded vehicle observations, including the initial one, and the hop limit K bounds the number of intersection transitions from that observation. Unavailable records after vehicle exit or episode termination contribute zero. The trajectory discount $\gamma_\tau \in [0, 1]$ weights later records; the RL discount γ weights intersection rewards across control steps.

The trajectory-conditioned reward assigned to intersection i at transition t is

$$\tilde{r}_i^t = \sum_{v \in \mathcal{V}_i^t} U_v^t. \tag{4}$$

Vehicle identity specifies whose costs enter the reward, and the realized trajectory specifies when and where those costs are collected. The same queueing cost may enter rewards for several earlier observations at the same or different intersections, giving those local objectives a shared cost component.

With $H = 1$ or $\gamma_\tau = 0$, Eq. (4) reduces to the local queue-based reward. With $K = 0$, it retains delayed costs at the source intersection; larger K also permits downstream costs.

4.2 Aggregate Discounted Return

The aggregate discounted return sums the trajectory-conditioned rewards over intersections and control steps, using the RL discount γ. Overlapping trajectory windows can include a recorded queueing cost in several rewards. Let $w_v^{t,s}$ denote the coefficient of u_v^s in U_v^t, with zero weight when v is unobserved at t. It includes the trajectory discount and the H, K cutoffs (Appendix F).

Proposition 1 (Coverage-weighted queueing cost) *For a recorded episode of length T, suppose each observed vehicle belongs to one intersection at each time. For $\gamma \in (0, 1]$, define*

$$m_v^s = \sum_{t=0}^{s} \gamma^{t-s} w_v^{t,s}. \tag{5}$$

Then the aggregate discounted return satisfies

$$\sum_{t=0}^{T-1} \gamma^t \sum_i \tilde{r}_i^t = -\sum_{s=0}^{T-1} \gamma^s \sum_v m_v^s q_v^s. \tag{6}$$

The proof exchanges the finite sums over source records and subsequent costs (Appendix F). With consecutive observations and $\gamma_\tau = \gamma$, m_v^s counts the historical windows covering the cost at (v, s). Such windows can originate at different times at the same intersection. Vehicle entry, exit, and window truncation make these weights vary across a trajectory. Equation (6) characterizes the aggregate discounted return under TCO. In training, each controller uses the trajectory-conditioned rewards as empirical surrogate rewards for value estimation and policy learning.

4.3 Reward Construction and Training

Algorithm 1 describes training for replay-based learners. Transitions and vehicle observations are stored in $\mathcal{B}$ and $\mathcal{L}$, respectively. At episode end, the trajectory-conditioned rewards are computed and completed samples enter replay $\mathcal{D}$. Updates during collection use completed samples from earlier episodes, following the baseline schedule. Appendix A.2 details replay insertion and warm-up.

CoordLight holds its policy fixed throughout each episode. At episode end, it fills in the trajectory-conditioned rewards, computes advantages, and then performs PPO updates. All variants retain the baseline architecture and use the baseline observations at execution.

The trajectory-conditioned rewards depend on subsequent actions and traffic conditions under the data-collection policy. Replay can therefore contain rewards constructed under different behavior policies.

Algorithm 1 Replay-Based Training with TCO

Require: Replay-based learner, $\mathcal{D}$; H, K, γ_τ
for each episode **do**
  Reset environment and buffers $\mathcal{B}, \mathcal{L}$
  for each control step t **do**
    Act; store the transition in $\mathcal{B}$
    Log (t, v, p_v^t, u_v^t) in $\mathcal{L}$
    if the episode ends **then**
      Compute U_v^s via Eq. (3)
      Assign $\tilde{r}_i^s$ via Eq. (4)
      Add labeled transitions to $\mathcal{D}$
    end if
    if update due and sufficient samples **then**
      Apply baseline updates from $\mathcal{D}$
    end if
  end for
end for

5 Experiments

5.1 Experimental Setup

We conduct experiments in CityFlow (Zhang et al., 2019) using the Jinan, Hangzhou, and Manhattan benchmarks (Wei et al., 2019b), with 12, 16, and 196 intersections, respectively (Figure 2).

Jinan and Hangzhou are evaluated under Moderate (Mod.), Heavy, and Extreme (Ext.) demands, and Manhattan under Moderate and Extreme demands. Relative to Moderate demand, Heavy demand increases traffic volume by 70% in Jinan and 50% in Hangzhou. Extreme demand approximately doubles the Moderate volume in these two networks. Performance is measured by CityFlow's average travel time (ATT) in seconds over vehicles generated by the simulation cutoff. Travel time includes entry waiting, and unfinished trips contribute their elapsed time at the cutoff.



(a) Jinan (12) (b) Hangzhou (16) (c) Manhattan (196)

Figure 2: Road networks used in the experiments.[1]

Each TCO variant in Table 1 uses the same seeds as its corresponding baseline. TCO changes the training reward while retaining the baseline observations, network architecture, update rule, and

[1]Map data © OpenStreetMap contributors: https://www.openstreetmap.org/copyright.

Table 1: ATT in seconds (lower is better). RL results are mean ± SD over ten seeds.

Algorithm	Jinan			Hangzhou			Manhattan	
	Mod.	Heavy	Ext.	Mod.	Heavy	Ext.	Mod.	Ext.
Advanced-MP	354.10	656.30	740.64	481.81	720.60	870.81	473.79	1083.17
G2P Control	353.10	819.60	896.01	469.81	714.93	886.00	439.11	1046.53
Advanced-QMIX	736.13 ± 51.67	1035.83 ± 91.37	1080.29 ± 91.07	753.17 ± 30.08	981.03 ± 65.17	1103.60 ± 63.37	854.48 ± 69.65	1639.61 ± 60.22
Advanced-MPLight	295.18 ± 2.65	494.35 ± 13.50	601.43 ± 14.86	394.41 ± 7.96	596.85 ± 11.24	748.15 ± 7.58	509.59 ± 76.74	1060.56 ± 28.73
Advanced-MPLight-TCO	$\mathbf{289.18 \pm 1.71}$	452.98 ± 6.85	$\mathbf{552.37 \pm 8.13}$	$\mathbf{361.22 \pm 3.61}$	$\mathbf{556.01 \pm 5.40}$	$\mathbf{711.93 \pm 8.69}$	428.08 ± 22.88	1005.28 ± 25.40
G2P-MPLight	299.47 ± 4.52	534.32 ± 13.44	657.88 ± 33.45	394.71 ± 8.66	613.09 ± 8.98	767.60 ± 10.61	444.16 ± 28.83	1016.66 ± 41.64
G2P-MPLight-TCO	290.05 ± 1.93	456.57 ± 7.69	560.04 ± 15.01	363.66 ± 1.93	560.08 ± 7.30	718.17 ± 6.42	411.51 ± 11.72	$\mathbf{963.59 \pm 31.53}$
CoordLight	304.06 ± 5.07	542.37 ± 35.18	707.96 ± 17.82	407.01 ± 4.50	623.76 ± 4.78	765.73 ± 4.05	383.81 ± 12.15	980.80 ± 39.70
CoordLight-TCO	289.58 ± 0.66	$\mathbf{446.43 \pm 13.51}$	679.88 ± 47.92	369.03 ± 3.77	567.29 ± 8.21	719.69 ± 5.14	$\mathbf{376.29 \pm 23.20}$	966.34 ± 26.26

training budget. Advanced-MPLight, G2P-MPLight, and Advanced-QMIX train for 125 episodes, and CoordLight for 500. Each episode lasts 3,600 seconds. We evaluate the final checkpoint with exploration disabled.

We follow the notation in Section 4.1: the trajectory horizon H limits the number of recorded vehicle observations, including the initial one; the hop limit K bounds intersection transitions; and the trajectory discount γ_τ weights later records. The learner accumulates the resulting rewards using the RL discount γ. All main TCO variants use a trajectory horizon of 24 records, a hop limit of 3, and a trajectory discount of 0.8. The lightweight configuration uses ten records and one hop with the same trajectory discount. Appendix B.1 reports its results; Table 9 lists the remaining settings.

Baselines We compare against two rule-based controllers and four RL methods:

- **Advanced-MP** (Zhang et al., 2022b): selects signal phases by comparing the demand of approaching vehicles in the current phase with the pressure of competing phases.
- **G2P Control** (Liao et al., 2025): selects the phase maximizing generalized pressure, defined from upstream demand and downstream receiving state.
- **Advanced-MPLight** (Zhang et al., 2022b): uses independent Q-learning with a FRAP network. Its inputs are the current phase, movement effective pressure, and vehicle counts within the effective range. In our experiments, it uses a negative local queue reward.
- **G2P-MPLight** (Liao et al., 2025): uses Advanced-MPLight with movement-level generalized pressure as its traffic representation and the same local queue reward.
- **Advanced-QMIX**: combines QMIX (Rashid et al., 2018) with Advanced-MPLight observations. It learns a joint action-value function through value decomposition, with a global reward.
- **CoordLight** (Zhang et al., 2025): uses attention and a GRU to encode queue dynamics. Its critic uses neighboring actions; the reward is the negative sum of local and neighboring queue lengths.

5.2 Main Results

Table 1 shows that TCO reduces mean ATT for all three RL backbones across all eight traffic settings. Relative to the corresponding baselines, the reductions range from 1.5% to 17.7%. Across 80 paired runs per backbone, TCO improves ATT in all runs with Advanced-MPLight, in 76 with G2P-MPLight, and in 75 with CoordLight.

5.3 Comparison of Reward Constructions

Using Advanced-MPLight, we compare TCO with Future-Region, which aggregates future queue-based rewards from the source intersection and its fixed one-hop neighbors. Sum adds the rewards within the region, while Mean divides by the number of included intersections. Table 2 compares Sum with TCO, using a horizon parameter of 24 and a trajectory discount of 0.8 for both methods; TCO uses a hop limit of 3.

The horizon parameter H counts recorded vehicle observations for TCO and consecutive control steps for Future-Region. Both apply γ_τ along their respective sequences. Gaps in vehicle records can therefore produce different elapsed window durations.

Table 2: Future-Region (Sum) and TCO: mean±SD ATT over five paired seeds.

Jinan			Hangzhou		
Demand	Future-Region	TCO	Demand	Future-Region	TCO
Mod.	301.30 ± 2.87	$\mathbf{289.35 \pm 1.71}$	Mod.	407.20 ± 15.25	$\mathbf{362.52 \pm 2.62}$
Heavy	731.49 ± 88.79	$\mathbf{455.52 \pm 7.47}$	Heavy	679.81 ± 67.88	$\mathbf{559.76 \pm 4.69}$
Ext.	984.18 ± 229.06	$\mathbf{552.57 \pm 8.84}$	Ext.	876.17 ± 46.26	$\mathbf{711.75 \pm 8.18}$

TCO obtains lower mean ATT in every setting and lower ATT in all 30 paired runs in Table 2. A further comparison uses the lightweight configuration with three seeds per setting. TCO yields lower ATT in all 18 paired runs against each of the Sum and Mean variants.

We also compare two reward constructions on Advanced-MPLight. Scaled Local applies a fixed rescaling to the local reward, and γ-Reward (Liu et al., 2021) applies the original recursive construction using future neighborhood feedback. Table 3 reports their results; Appendix C.5 describes the reward definitions. Relative to the Local results in Table 1, Scaled Local reduces mean ATT in seven settings, and γ-Reward reduces it in all eight. TCO has lower mean ATT than both constructions in each setting, with a small difference from γ-Reward on Manhattan under Extreme demand.

Table 3: Additional reward constructions with Advanced-MPLight (ATT, mean $\pm$ SD over five seeds). Local and TCO results are given in Table 1.

Scenario	Scaled Local	γ-Reward
Jinan (Mod.)	292.29 ± 1.17	293.44 ± 1.60
Jinan (Heavy)	484.50 ± 5.87	476.30 ± 10.95
Jinan (Ext.)	590.76 ± 20.64	598.16 ± 15.77
Hangzhou (Mod.)	398.51 ± 6.60	376.54 ± 4.03
Hangzhou (Heavy)	588.67 ± 1.71	573.98 ± 7.67
Hangzhou (Ext.)	725.34 ± 5.83	725.98 ± 2.91
Manhattan (Mod.)	478.96 ± 13.60	466.36 ± 40.91
Manhattan (Ext.)	1020.74 ± 10.98	1006.70 ± 16.90

5.4 Controlled Network Scaling

Table 4 compares Advanced-MPLight with its TCO variant on Hangzhou-style networks with 4, 9, and 16 intersections. The networks share road and lane parameters, demand density, and training protocol. TCO uses the main trajectory configuration. It reduces mean ATT at every tested size and improves all ten paired runs at each size (two-sided sign test, $p = 0.00195$).

Table 4: Controlled scaling: ATT over ten seeds (mean and standard deviation).

Network	Local	TCO
2×2	213.75 ± 1.97	$\mathbf{208.42 \pm 0.96}$
3×3	255.81 ± 2.30	$\mathbf{246.89 \pm 0.80}$
4×4	394.41 ± 7.96	$\mathbf{361.22 \pm 3.61}$

Table 5: Cross penalties on Hangzhou (ATT, mean $\pm$ SD over five seeds).

Demand	+Cross	+Gated
Moderate	362.24 ± 2.38	$\mathbf{360.83} \pm 5.20$
Heavy	550.90 ± 3.37	$\mathbf{549.60} \pm 3.52$
Extreme	714.98 ± 4.52	$\mathbf{714.24} \pm 2.21$

Appendix E analyzes reward variability and action–reward dependence under fixed policies.

5.5 Trajectory Parameters and Aggregation

We use Advanced-MPLight-TCO to examine the trajectory horizon H, hop limit K, trajectory discount γ_τ, and reward aggregation. Each parameter sweep varies one parameter while holding the other two fixed. Bold entries in the ablation tables indicate the lowest ATT within each scenario.

Hop Limit (K). Table 6 reports the hop-limit sweep with a horizon of 64 records and a trajectory discount of 0.8. The $K = 0$ variant, which retains delayed local costs but excludes downstream feedback, yields higher reported ATT than $K = 3$ in seven of the eight settings. ATT does not decrease monotonically with K.

Trajectory Discount (γ_τ). At zero trajectory discount, the reward contains only the immediate local queue cost while retaining TCO's episode-end replay insertion (Table 7), providing a Local reward control under the same insertion schedule. Discounts of 0.3, 0.5, and 0.8 improve ATT over this condition in all six settings. The best discount depends on the scenario. Under Extreme demand, Jinan and Hangzhou achieve their lowest ATT at 0.5 and 0.8, respectively; giving equal weight to all retained costs increases ATT in both networks.

Table 6: Hop limit K, with $\gamma_\tau = 0.8$, $H = 64$.

K	Jinan			Hangzhou			Manhattan	
	Mod.	Heavy	Ext.	Mod.	Heavy	Ext.	Mod.	Ext.
0	**287.5**	454.8	567.9	361.9	559.9	714.8	436.7	1012.4
1	289.8	446.3	550.7	361.7	**541.5**	718.5	437.1	1035.1
2	288.2	**439.8**	551.1	356.9	557.3	714.5	434.0	993.9
3	288.6	444.8	**540.7**	**355.6**	542.6	**705.7**	430.5	**972.7**
4	289.8	458.6	556.9	357.9	557.7	713.7	**419.4**	995.0

Table 7: Trajectory discount γ_τ, with $H = 64$ and $K = 8$.

γ_τ	Jinan			Hangzhou		
	Mod.	Heavy	Ext.	Mod.	Heavy	Ext.
0	295.17	493.52	600.71	398.19	614.37	745.07
0.3	295.06	466.76	586.57	375.87	571.43	727.09
0.5	290.94	467.48	**565.87**	374.90	568.66	718.84
0.8	**288.45**	459.06	566.71	**356.46**	**559.22**	**716.02**
1.0	289.08	**440.60**	671.27	362.71	565.15	738.34

Trajectory Horizon (H). We vary the trajectory horizon from 5 to 64 records, with the hop limit fixed at 8 and the trajectory discount at 0.8. Across the settings in Table 13, longer horizons yield no consistent improvement.

Cross-Trajectory Penalty. We examine two extensions that incorporate downstream queue costs from other vehicles. Table 5 compares a cross-trajectory penalty (+Cross) with a gated variant (+Gated), which activates when the downstream queued-vehicle fraction reaches 0.5. Both variants use the lightweight configuration. +Gated gives slightly lower mean ATT than +Cross at each demand level, with differences below 1.5 seconds. Appendix C.3 defines the additional penalty. Main experiments use trajectory-only rewards.

Reward Aggregation. Table 8 examines two changes to the vehicle sum in Eq. (4). Temporal normalization (Time norm.) divides each vehicle's trajectory reward by the sum of the discount weights within its valid trajectory window. Vehicle averaging (Veh. mean) divides the summed intersection reward by the number of vehicles observed there, with zero reward for an empty vehicle set. Both variants use the lightweight configuration. Vehicle averaging yields substantially higher mean ATT than temporal normalization in all four settings. Appendix C.4 gives the definitions.

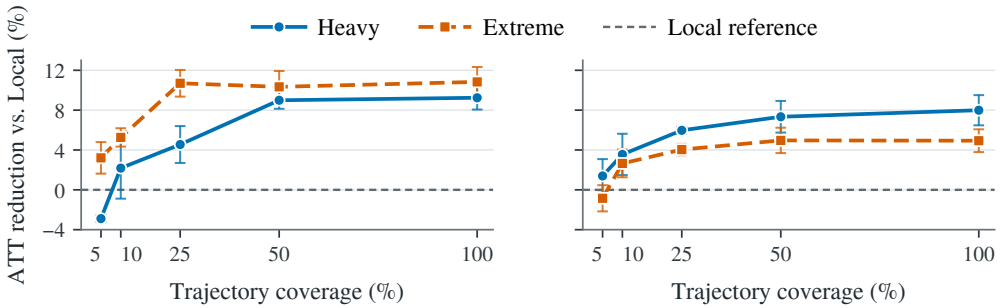
Table 8: Reward aggregation (ATT, mean $\pm$ SD over five seeds).

Scenario	Time norm.	Veh. mean
Jinan (Mod.)	$\mathbf{302.30} \pm 1.13$	706.63 ± 81.18
Jinan (Ext.)	$\mathbf{624.71} \pm 30.00$	969.86 ± 93.19
Hangzhou (Mod.)	$\mathbf{403.26} \pm 8.15$	586.83 ± 37.65
Hangzhou (Ext.)	$\mathbf{747.09} \pm 9.72$	934.13 ± 29.61

5.6 Learning with Partial Trajectory Coverage

In practical deployments, complete trajectories may be available for only a subset of vehicles. We vary the fraction of vehicles with trajectory feedback using Advanced-MPLight and the lightweight configuration. Immediate local queue costs are retained for all vehicles, and delayed feedback from the selected trajectories is scaled by the reciprocal of the coverage fraction. Figure 3 reports ATT reductions relative to Local. Appendix D defines the partial-coverage reward and reports the means and standard deviations.

At 10% trajectory coverage, mean ATT is lower than Local in all four settings, with reductions ranging from 2.2% to 5.3%. Full coverage gives the lowest reported mean in three settings; Hangzhou under Extreme demand attains its lowest mean at 50% coverage.



(a) Jinan (b) Hangzhou

Figure 3: ATT reduction with partial coverage over five seeds. Error bars show ± 1 SD of trajectory-variant ATT, normalized by the fixed Local mean. Local is trained independently with the same seeds.

6 Discussion and Conclusion

Limitations and Future Work. Practical deployment depends on reliable vehicle association; the effects of association errors on learning remain to be evaluated. Continuous operation would require finalizing rewards as trajectory windows close and accommodating the resulting feedback delay in online updates. The current evaluation uses fixed trajectory parameters. Adapting these parameters to changing traffic conditions is a direction for future work.

Conclusion. TCO constructs local learning objectives from the subsequent queueing costs of currently observed vehicles, combining continued waiting at the same intersection and downstream queueing through a common trajectory-based construction. Its coverage-weighted interpretation makes explicit how overlapping trajectory windows affect the aggregate discounted return. Across the evaluated road networks and traffic demands, TCO improves the three learning backbones while retaining their architectures and decentralized execution.

Reproducibility Statement

Section 4.1 defines the trajectory-conditioned reward, and Algorithm 1 describes its integration into training. Appendix F gives the assumptions and proof of Proposition 1. Section 5.1 and Appendix A document the evaluation protocol, training settings, and benchmark sources. Appendices C and D provide the reward-variant definitions and supplementary results.

References

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 *AAAI Conference on Artificial Intelligence*, volume 34, pp. 3414–3421, 2020.

T. Chu, S. Chinchali, and S. Katti. Multi-agent reinforcement learning for networked system control. *arXiv preprint arXiv:2004.01339*, 2020a.

Tianshu Chu, Jie Wang, Lara Codecà, and Zhaojian Li. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, 21(3): 1086–1095, 2020b. doi: 10.1109/TITS.2019.2901791.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926*, 2017. URL https://arxiv.org/abs/1705.08926.

H. Goel, Y. Zhang, M. Damani, and G. Sartoretti. Sociallight: Distributed cooperation learning towards network-wide traffic signal control. *arXiv preprint arXiv:2305.16145*, 2023.

Shariq Iqbal and Fei Sha. Actor-attention-critic for multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 2961–2970. PMLR, 2019.

Xiao-Cheng Liao, Yi Mei, Mengjie Zhang, and Xiang-Ling Chen. Generalized phase pressure control enhanced reinforcement learning for traffic signal control, 2025. URL https://arxiv.org/abs/2503.20205.

Junfan Lin, Yuying Zhu, Lingbo Liu, Yang Liu, Guanbin Li, and Liang Lin. Denselight: Efficient control for large-scale traffic signals with dense feedback. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pp. 6058–6066. ijcai.org, 2023. doi: 10.24963/IJCAI.2023/672. URL https://doi.org/10.24963/ijcai.2023/672.

Junjia Liu, Huimin Zhang, Zhuang Fu, and Yao Wang. Learning scalable multi-agent coordination by spatial differentiation for traffic signal control. *Engineering Applications of Artificial Intelligence*, 100:104165, 2021. doi: 10.1016/j.engappai.2021.104165.

Frans A. Oliehoek and Christopher Amato. *A Concise Introduction to Decentralized POMDPs*. SpringerBriefs in Intelligent Systems. Springer, 2016. doi: 10.1007/978-3-319-28929-8.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 4295–4304. PMLR, 2018.

Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2018.

Pravin Varaiya. The max-pressure controller for arbitrary networks of signalized intersections. In *Advances in Dynamic Network Modeling in Complex Transportation Systems*, pp. 27–66. Springer, 2013.

Yanan Wang, Tong Xu, Xin Niu, Chang Tan, Enhong Chen, and Hui Xiong. Stmarl: A spatio-temporal multi-agent reinforcement learning approach for cooperative traffic light control. *IEEE Transactions on Mobile Computing*, 21(6):2228–2242, 2022. doi: 10.1109/TMC.2020.3033782.

Hua Wei, Chacha Chen, Guanjie Zheng, Kan Wu, Vikash Gayah, Kai Xu, and Zhenhui Li. Presslight: Learning max pressure control to coordinate traffic signals in arterial network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1290–1298, 2019a.

Hua Wei, Nan Xu, Huichu Zhang, Guanjie Zheng, Xinshi Zang, Chacha Chen, Weinan Zhang, Yanmin Zhu, Kai Xu, and Zhenhui Li. Colight: Learning network-level cooperation for traffic signal control. In *ACM International Conference on Information and Knowledge Management*, pp. 1913–1922, 2019b.

Qiang Wu, Liang Zhang, Jun Shen, Linyuan Lü, Bo Du, and Jianqing Wu. Efficient pressure: Improving efficiency for signalized intersections. *arXiv preprint arXiv:2112.02336*, 2021. URL https://arxiv.org/abs/2112.02336.

Huichu Zhang, Siyuan Feng, Chang Liu, Yaoyao Ding, Yichen Zhu, Zihan Zhou, Weinan Zhang, Yong Yu, Haiming Jin, and Zhenhui Li. CityFlow: A multi-agent reinforcement learning environment for large scale city traffic scenario. In *The World Wide Web Conference*, pp. 3620–3624. ACM, 2019. doi: 10.1145/3308558.3314139.

Liang Zhang, Qiang Wu, and Jianming Deng. AttentionLight: Rethinking queue length and attention mechanism for traffic signal control. *arXiv preprint arXiv:2201.00006v2*, 2022a. URL https://arxiv.org/abs/2201.00006v2.

Liang Zhang, Qiang Wu, Jun Shen, Linyuan Lü, Bo Du, and Jianqing Wu. Expression might be enough: representing pressure and demand for reinforcement learning based traffic signal control. In *International Conference on Machine Learning*, pp. 26645–26654. PMLR, 2022b. URL https://proceedings.mlr.press/v162/zhang22ah.html.

Yifeng Zhang, Harsh Goel, Peizhuo Li, Mehul Damani, Sandeep Chinchali, and Guillaume Sartoretti. Coordlight: Learning decentralized coordination for network-wide traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, 26(6):8034–8049, 2025. doi: 10.1109/TITS.2025.3546261. URL https://doi.org/10.1109/TITS.2025.3546261.

A Experimental Details

A.1 Networks and Training Settings

Table 9 lists network sizes and experimental settings. Section 5.1 describes the baseline methods, evaluation metric, seed counts, and trajectory configurations.

Table 9: Network configurations and experimental settings.

Category	Value
Jinan grid	3×4
Hangzhou grid	4×4
Manhattan grid	7×28
Episode Length	3600 seconds
Decision Interval	10 seconds
Yellow Interval	5 seconds
Action Space Size	8
Optimizer	Adam
Learning Rate	1×10^{-3}
Hidden Size	128
Batch Size	128
RL discount (γ)	0.99

A.2 Training and Evaluation Protocol

For each training seed, we evaluate the checkpoint saved at the end of the prescribed training budget, with exploration disabled. Training and evaluation use the same traffic flow file for each scenario. Reported means and standard deviations summarize the resulting per-seed ATT values.

The replay-based Local baselines insert transitions after each control step. TCO stores the current episode's transitions until its trajectory rewards are computed, then inserts them into replay at episode end. Both use the baseline update schedule and begin updates once replay contains enough samples for a minibatch. TCO therefore waits for its first episode to finish before replay updates can begin; subsequent episodes draw on completed earlier episodes during collection. CoordLight collects each episode with a fixed policy, then computes advantages and performs PPO updates after the rewards are available.

A.3 Evaluation Metric

For a simulation ending at time T, let $\mathcal{F}(T)$ contain the vehicles that have completed their trips and $\mathcal{P}(T)$ those still present, including vehicles waiting to enter the road network. CityFlow computes

$$\mathrm{ATT}(T) = \frac{\sum_{v \in \mathcal{F}(T)} (t_v^{\mathrm{exit}} - t_v^{\mathrm{gen}}) + \sum_{v \in \mathcal{P}(T)} (T - t_v^{\mathrm{gen}})}{|\mathcal{F}(T)| + |\mathcal{P}(T)|}, \tag{7}$$

where t_v^{gen} and t_v^{exit} denote generation and exit times. Unfinished trips contribute elapsed time up to the cutoff, and future demand that has yet to generate a vehicle is excluded. The value is zero when the denominator is zero.

A.4 Traffic Demand

Table 10 reports the vehicle counts specified by the flow files. Each admitted, recorded vehicle yields one trajectory; the number of recorded trajectories can be smaller when some vehicles are not admitted before the simulation cutoff.

A.5 Compute and Runtime

The reported runtime measurements use a reference machine equipped with a 6-core/12-thread CPU, 16 GB of RAM, and an NVIDIA GeForce RTX 3060 Laptop GPU. Depending on the map and

Table 10: Vehicle counts specified by the experimental flow files.

Map	Moderate	Heavy	Extreme
Syn-3	5,675	–	11,350
Jinan	6,295	10,703	12,590
Hangzhou	5,966	8,949	11,932
Manhattan	6,831	–	11,058
Manhattan-48	2,824	–	5,648
Hangzhou-style $2 \times 2/3 \times 3/4 \times 4$	2,911/4,414/5,966	–	–

demand configurations, a single Advanced-MPLight, G2P-MPLight, or Advanced-QMIX experiment typically takes 0.5 to 2 hours, whereas CoordLight requires 3 to 12 hours. Across four-map profiling, TCO adds 2.6% mean training wall-time for non-CoordLight backbones and 3.3% for CoordLight, with 7%–20% peak-memory overhead. Execution is unchanged; this overhead arises only from constructing training labels.

A.6 Assets and Sources

Table 11 identifies the existing software and data sources used for simulation, baseline implementations, benchmarks, and geographic illustrations. License information refers to the distributed repositories; the G2P repository does not specify a software license.

Table 11: Software and data sources used in this work.

Asset	Use	License or terms	Source
CityFlow	Simulator	Apache-2.0	Repository; Zhang et al. (2019)
Advanced-XLight	Baselines	GPL-3.0	Repository; Zhang et al. (2022b)
CoordLight	Baseline	MIT	Repository; Zhang et al. (2025)
G2P / G2P-MPLight	Baselines	Unspecified	Repository; Liao et al. (2025)
OpenStreetMap	Map illustrations	ODbL	OpenStreetMap contributors
NYC TLC trip records	Traffic demand source	Provider terms	NYC TLC data portal

B Additional Performance Results

B.1 Lightweight Configuration

Table 12: ATT for TCO variants with the lightweight configuration ($H = 10$, $K = 1$, and $\gamma_\tau = 0.8$), reported as mean $\pm$ SD over five runs.

Algorithm	Jinan			Hangzhou			Manhattan	
	Mod.	Heavy	Ext.	Mod.	Heavy	Ext.	Mod.	Ext.
Advanced-MPLight-TCO	289.77±2.05	449.21±6.84	**554.18**±10.5	**362.53**±2.94	555.73±11.21	**714.43**±4.53	407.16±5.07	1066.13±19.1
G2P-MPLight-TCO	288.53±0.77	531.05±16.6	601.50±10.7	395.60±9.55	618.78±14.7	760.03±2.59	410.84±11.8	970.31±27.6
CoordLight-TCO	**287.94**±0.88	**426.58**±38.0	644.74±50.3	366.86±2.27	**551.75**±14.7	731.59±19.7	**379.32**±21.9	**892.81**±70.1

B.2 Comparison with COMA

We additionally evaluate counterfactual multi-agent policy gradients (COMA) (Foerster et al., 2017) using the observations of Advanced-MPLight on Syn-3, a synthetic network with three intersections. Across five seeds, COMA obtains ATT of 664.15 ± 71.94 under Moderate demand and 728.90 ± 9.34 under Extreme demand. The corresponding reported TCO values are 123.27 and 562.50 seconds. Advanced-QMIX in Table 1 provides a further comparison with learning from a shared global reward; it estimates a joint action-value function through value decomposition.

C Reward Ablations

This section supplements the reward ablations in Section 5.5 with the full horizon sweep, the Extreme-demand discount results, and the definitions of the reward variants.

C.1 Trajectory Horizon

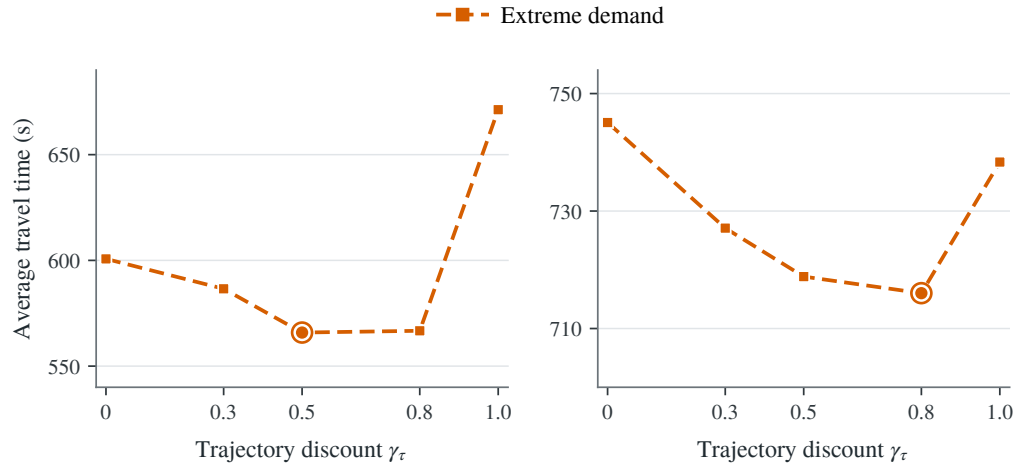
Table 13 gives the complete trajectory-horizon sweep for Advanced-MPLight-TCO. It includes Syn-3, a synthetic network with three intersections, alongside the three urban benchmarks. The lowest ATT in each scenario is shown in bold.

Table 13: Ablation analysis of the trajectory horizon H, with $K = 8$ and $\gamma_\tau = 0.8$ fixed.

Scenario	$H = 5$	$H = 8$	$H = 12$	$H = 16$	$H = 24$	$H = 32$	$H = 48$	$H = 64$
Syn-3 (Mod.)	149.65	139.68	140.44	148.27	143.16	**134.52**	137.32	139.15
Syn-3 (Heavy)	**362.63**	396.76	377.01	403.10	431.57	382.06	415.59	388.91
Syn-3 (Ext.)	584.58	591.29	558.08	575.62	**556.88**	578.79	576.52	580.44
Jinan (Mod.)	287.89	288.32	**287.10**	288.99	288.35	288.42	288.98	288.45
Jinan (Heavy)	445.08	449.22	441.89	443.60	**436.79**	447.20	450.68	459.06
Jinan (Ext.)	558.36	558.42	549.54	**544.17**	551.79	555.84	554.22	566.71
Hangzhou (Mod.)	365.01	365.60	360.41	356.56	361.38	364.42	362.40	**356.46**
Hangzhou (Heavy)	555.59	551.78	552.48	555.33	563.14	559.30	**547.99**	559.22
Hangzhou (Ext.)	723.38	719.39	708.16	713.86	**707.40**	720.45	717.75	716.02
Manhattan (Mod.)	446.56	464.17	410.68	410.06	**404.01**	418.19	419.90	453.46
Manhattan (Ext.)	1019.34	989.41	1017.26	1026.38	994.74	1008.95	**932.51**	988.97

C.2 Trajectory Discount Under Extreme Demand

Figure 4 plots the two Extreme-demand columns of Table 7. Jinan and Hangzhou attain their lowest reported ATT at $\gamma_\tau = 0.5$ and 0.8, respectively, among the tested values.



(a) Jinan (b) Hangzhou

Figure 4: Effect of the trajectory discount under Extreme demand, with $H = 64$ and $K = 8$. Circled points indicate the lowest reported ATT among the tested discounts. The two panels use different vertical scales.

C.3 Cross-Trajectory Feedback

The variants in Table 5 add a penalty for other vehicles' downstream queueing costs. At intersection j and control step s, let $\mathcal{Q}_j^s$ be the queued vehicles on non-right-turn movements. Let $\mathcal{A}_j^s$ contain vehicles on currently permitted non-right-turn movements that have a recorded upstream visit. For each $w \in \mathcal{A}_j^s$, write (i_w, t_w) for its last recorded intersection–time pair before entering j. The fraction of vehicle v's cost assigned to source (i, t) is

$$\omega_{i,t}^{v,j,s} = \frac{\sum_{w \in \mathcal{A}_j^s \setminus \{v\}} \mathbb{I}[(i_w, t_w) = (i, t)]\, \gamma_\tau^{s-t_w}}{\sum_{w \in \mathcal{A}_j^s \setminus \{v\}} \gamma_\tau^{s-t_w}}, \tag{8}$$

with zero weight when the denominator is zero. The exponent counts elapsed control steps between the upstream record and the downstream cost. The augmented reward is

$$r_{i,+}^t = \tilde{r}_i^t + \lambda \sum_{j,s} g_j^s \sum_{v \in \mathcal{Q}_j^s} \omega_{i,t}^{v,j,s} u_v^s. \tag{9}$$

We use $\lambda = 1$. For +Cross, $g_j^s = 1$; for +Gated, $g_j^s = \mathbb{I}[\rho_j^s \geq 0.5]$, where ρ_j^s is the queued fraction of all vehicles observed at (j, s). The base TCO reward is retained in both variants.

C.4 Reward Aggregation Variants

Let $\mathcal{H}_v^t$ contain the retained observation offsets for vehicle v within the trajectory window starting at t, subject to the horizon and hop limits. Temporal normalization divides each vehicle's trajectory reward by the sum of the discount weights within that same window, before aggregating across vehicles:

$$r_{i,\mathrm{TimeNorm}}^t = \sum_{v \in \mathcal{V}_i^t} \frac{U_v^t}{\sum_{h \in \mathcal{H}_v^t} \gamma_\tau^h}. \tag{10}$$

Vehicle averaging retains each vehicle's trajectory reward and averages over the vehicles observed at the source intersection:

$$r_{i,\mathrm{VehicleMean}}^t = \begin{cases} \dfrac{1}{|\mathcal{V}_i^t|} \displaystyle\sum_{v \in \mathcal{V}_i^t} U_v^t, & |\mathcal{V}_i^t| > 0, \\ 0, & |\mathcal{V}_i^t| = 0. \end{cases} \tag{11}$$

Table 8 reports results for these two aggregation variants.

C.5 Additional Reward Comparisons

Scaled Local applies the fixed multiplier $r_{i,\mathrm{Scaled}}^t = 4.97 r_i^t$ to the local queue reward. The multiplier approximates the nominal discount weight of the main TCO window, $\sum_{h=0}^{23} 0.8^h \approx 4.9764$. For γ-Reward, we follow the original recursive reward construction in Section 2.2, Eq. (2) of Liu et al. (2021), using Advanced-MPLight as the learner:

$$R_i(t) = r_i(t) \left[1 + g \tanh \left(\sum_{j \in \mathcal{N}_i} \left(\frac{R_j(t+d)}{r_j(t)} - c \right) \right) \right]. \tag{12}$$

Here $r_i(t)$ is the current local waiting penalty and $\mathcal{N}_i$ contains directly adjacent intersections, whose contributions are summed. We use correction strength $g = 0.5$, relative-change threshold $c = 0.8$, and delay $d = 10$ seconds (one decision step). Rewards are computed backward over the full episode, with the final step retaining its local reward. This mode uses neither the trajectory discount γ_τ nor the H, K cutoffs. The learner uses reward scale 0.075, RL discount $\gamma = 0.99$, and learning rate 10^{-3}. Table 3 reports these reward comparisons.

When $r_j(t) = 0$, we set the ratio to 1 if $R_j(t+d) = 0$ and to $+\infty$ if $R_j(t+d) < 0$, without adding an epsilon. The latter case saturates $\tanh$ at 1, giving the maximum correction multiplier $1 + g = 1.5$. If the source reward $r_i(t) = 0$, the corrected reward remains zero.

D Partial Trajectory Coverage

D.1 Coverage Results

The coverage experiment in Section 5.6 uses a vehicle-level availability indicator Z_v, with the same indicator applied throughout a vehicle's recorded trajectory. For coverage fraction p, the reward is

$$\tilde{r}_{i,p}^t = r_i^t + \frac{1}{p} \sum_{v \in \mathcal{V}_i^t} Z_v \big(U_v^t - u_v^t\big). \tag{13}$$

Here r_i^t is the immediate Local reward for all observed vehicles, and $U_v^t - u_v^t$ contains the subsequent costs in the selected vehicle's trajectory window. Multiplication by $1/p$ compensates for the reduced number of trajectories contributing delayed feedback. With equal inclusion probability p, this scaling preserves the expected sum of delayed costs on a fixed recorded episode.

Table 14 provides the ATT values underlying Figure 3.

Table 14: ATT under partial trajectory coverage: five-seed means, with SDs below. Trajectory variants use the lightweight configuration; Local is independently trained with the same seeds.

Scenario	Local	5%	10%	25%	50%	100%
Jinan (Heavy)	487.58 ±7.34	501.66 ±1.73	476.95 ±14.95	465.40 ±9.03	443.74 ±4.18	**442.52** ±5.77
Jinan (Ext.)	625.02 ±19.64	604.96 ±9.90	592.13 ±5.75	558.16 ±8.38	560.40 ±9.96	**557.30** ±9.35
Hangzhou (Heavy)	597.96 ±6.00	589.63 ±10.12	576.71 ±12.40	562.24 ±2.05	554.10 ±9.50	**550.17** ±9.07
Hangzhou (Ext.)	751.74 ±7.89	758.12 ±9.87	731.81 ±10.46	721.35 ±4.29	**714.44** ±9.58	714.67 ±8.61

D.2 Continuous Reward Labeling

During interaction, the current implementation records vehicle identities, positions, and instantaneous rewards alongside the baseline transitions. At episode end, it computes $\tilde{r}_i^t$ for the completed episode, followed by replay insertion or advantage computation and PPO updates as described in Section 4.3. For a continuous stream, a rolling buffer could finalize a reward when its trajectory window closes through vehicle exit or an H, K limit. Continual online adaptation remains to be evaluated.

E Fixed-Policy Diagnostics

These analyses examine the constructed rewards on logged trajectories under fixed policies. They supplement the controller comparisons in Tables 1 and 4.

E.1 Trajectory Reward Variability

We first examine the effect of the trajectory horizon under fixed policies across four networks and two demands. Increasing the trajectory horizon H from 1 to 64 records raises the raw variance of vehicle trajectory rewards by approximately 10 to 17 times. The normalized variance divides this quantity by the squared mean absolute trajectory reward:

$$\frac{\mathrm{Var}(U_v^t)}{(\mathbb{E}|U_v^t|)^2}.$$

This normalized measure decreases in all eight settings.

We then vary network size. The 2×2, 3×3, and 4×4 networks share road/lane parameters, phases, and Hangzhou-Moderate flow distributions scaled by one boundary-length demand-density rule. Table 15 uses $H = 24, K = 3, \gamma_\tau = 0.8$. For each seed, we average three episodes and then report mean±SD over three seeds. Raw and normalized variation in trajectory rewards show no systematic increase with network size. Both measures combine stochastic noise and variation in traffic states.

Table 15: Variation in vehicle trajectory rewards under controlled network scaling.

Size	$\mathrm{Var}(U_v^t)$	$\mathrm{Var}(U_v^t)/(\mathbb{E}\lvert U_v^t\rvert)^2$
2×2	0.7134 ± 0.0060	0.4000 ± 0.0084
3×3	0.6793 ± 0.0037	0.3589 ± 0.0053
4×4	0.6807 ± 0.0112	0.3520 ± 0.0071

E.2 Action–Reward Dependence

We use mutual information (MI) as a fixed-policy diagnostic of action–reward dependence:

$$I(A_i; R_i) = H(R_i) - H(R_i \mid A_i). \tag{14}$$

Here, R_i denotes the constructed intersection reward, and $H(\cdot)$ denotes entropy. MI characterizes statistical dependence, while ATT measures controller performance. We use the MI estimates for post hoc analysis; optimization and model selection follow the baseline training procedures.

We collect offline trajectories with fixed Max-Pressure policies on Syn-3 (1×3), Hangzhou (4×4), Manhattan-48 (3×16), and Manhattan (7×28). From the same logged transitions, we retrospectively construct Global, Region, Local, and Trajectory rewards. We estimate $I(A_i; R_i)/H(A_i)$ with the KSG estimator and report values relative to Local.

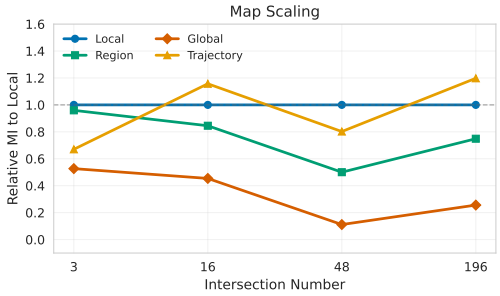


Figure 5: Relative action–reward dependence across map sizes.

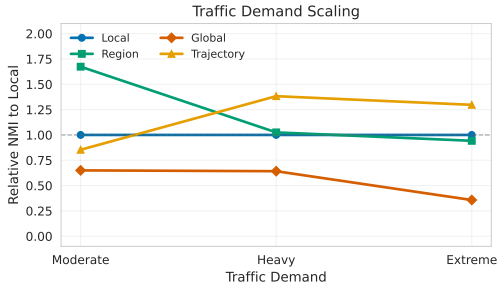


Figure 6: Relative action–reward dependence across traffic demands.

In Figure 5, Global and Region have lower measured dependence than Local at each network size. Trajectory exceeds Local on Hangzhou and Manhattan, and falls below it on Syn-3 and Manhattan-48. In the Jinan demand sweep (Figure 6), Region has the highest value under Moderate demand; Trajectory has the highest values under Heavy and Extreme demand.

F COVERAGE-WEIGHTED QUEUEING COST

For a recorded episode of length T, let $\mathcal{V}$ contain all recorded vehicle identities. Extend q_v^s by zero at unrecorded times. For v observed at time t, define

$$w_v^{t,s} = \sum_{h=0}^{H-1} \gamma_\tau^h \mathbb{I}\big[\tau_v(t,h) = s,\ \ell_v^{t,h} \le K\big]\,, \tag{15}$$

where unavailable trajectory entries contribute zero. Set $w_v^{t,s} = 0$ when v is unobserved at t. This definition retains the original control-step timestamps when successive vehicle records have gaps. Equation (3) becomes

$$U_v^t = -\sum_{s=t}^{T-1} w_v^{t,s} q_v^s. \tag{16}$$

Proof of Proposition 1. Each recorded vehicle belongs to one intersection at a given time. Substituting the expression for U_v^t and exchanging the finite sums gives

$$\sum_{t=0}^{T-1} \gamma^t \sum_i \tilde{r}_i^t = -\sum_{t=0}^{T-1} \gamma^t \sum_{v\in\mathcal{V}} \sum_{s=t}^{T-1} w_v^{t,s} q_v^s \tag{17}$$

$$= -\sum_{s=0}^{T-1} \sum_{v\in\mathcal{V}} q_v^s \sum_{t=0}^{s} \gamma^t w_v^{t,s} \tag{18}$$

$$= -\sum_{s=0}^{T-1} \gamma^s \sum_{v\in\mathcal{V}} \left(\sum_{t=0}^{s} \gamma^{t-s} w_v^{t,s} \right) q_v^s. \tag{19}$$

The last line is Eq. (6). The identity holds for each recorded trajectory and hence also after taking expectation under any trajectory-generating policy. □

Consecutive observations. When $\tau_v(t,h) = t+h$, define $\chi_{v,s,h} = 1$ if vehicle v is observed at $s-h$ and the record at s lies within the K-hop limit of that window, and zero otherwise. Then

$$m_v^s = \sum_{h=0}^{\min(H-1,s)} \left(\frac{\gamma_\tau}{\gamma}\right)^h \chi_{v,s,h}. \tag{20}$$

For $\gamma_\tau = \gamma$, the weight counts the historical windows covering a queue-cost observation. Several windows can originate from the same intersection at different times. Vehicle entry, exit, finite records, and the hop cutoff affect this count. If observations have gaps, Eq. (6) continues to use the control-step discount γ^{t-s} and the record-based trajectory discount in $w_v^{t,s}$.

Constant coverage. If m_v^s is the same positive constant for every cost observation and every trajectory under consideration, the aggregate discounted return under TCO is a constant multiple of the discounted sum of the local rewards. In general, the weights depend on the realized trajectories. This identity provides a trajectory-level interpretation of the aggregate return as a coverage-weighted queueing cost.

AI USE STATEMENT

We used generative AI tools to assist with running experimental scripts and improving the clarity and wording of the manuscript.