VLALIGHT: LIGHTWEIGHT VISION-LANGUAGE-ACTION MODELS FOR EMERGENCY-AWARE TRAFFIC SIGNAL CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Traffic signal control (TSC) is essential for mitigating urban congestion. Recent advances in vision-language models (VLMs) enable richer interpretation of intersection scenes, opening new opportunities for visual-context-aware TSC. However, existing VLM-based controllers follow a cascaded perceive–reason–control pipeline whose vision-to-text conversion loses fine-grained spatial detail and whose sequential multi-agent inference adds latency that is problematic for real-time control. We propose VLALight, a lightweight vision-language-action (VLA) framework that maps a stitched multi-directional observation of an intersection, together with a textual description of its phase configuration, directly to discrete signal-phase actions in a single forward pass, avoiding intermediate image-to-text descriptions and handcrafted traffic-state representations. A simulation-grounded pipeline anchored to real intersection topologies provides expert demonstrations, and a compact 0.5B-parameter multimodal backbone, adapted with LoRA and a small set of learnable phase queries, answers each decision in 0.22 s on a local GPU. In closed-loop simulation on six replicas of real intersections, VLALight attains the strongest emergency-vehicle service among the compared controllers, reducing pooled emergency waiting time by 21.1% over the cascaded cloud-scale VLMLight, and generalizes to unseen intersection topologies and traffic-flow.

1 INTRODUCTION

Urban traffic congestion remains a major challenge in modern cities, and the signal strategies at intersections—the bottlenecks of road networks—largely determine network-wide efficiency and safety. Traffic signal control (TSC) methods have evolved from fixed-time plans to adaptive rule-based strategies such as SOTL (Gershenson, 2005) and Max Pressure (Varaiya, 2013) and, more recently, deep reinforcement learning (Wei et al., 2021). These controllers represent the traffic state with compact hand-crafted statistics such as queue lengths and lane occupancy, which discard fine-grained spatial information and can therefore misjudge abnormal scenarios such as accidents or emergency-vehicle priority requests, where scene understanding matters most. LLMs and VLMs have been introduced to inject semantic understanding into TSC (Lai et al., 2025; Wang et al., 2025a; Zou et al., 2026), yet they adopt a cascaded pipeline in which traffic images are first converted into textual descriptions, over which an LLM then reasons to produce control suggestions; the vision-to-text conversion again compresses spatial detail, and multi-stage inference adds latency. Both lines of work share a common bottleneck: *the visual scene is compressed before decisions are made*.

Vision-language-action (VLA) models offer a direct way to remove this bottleneck. Following RT-2 (Zitkovich et al., 2023), we use the term in its operational sense: a single vision-language backbone consumes raw visual observations and textual task descriptions, and produces actions through the same generative token-prediction interface as language, so that perception, instruction understanding, and action selection share one representation space. This paradigm transfers web-scale priors to control and naturally accommodates discrete action vocabularies (Octo Model Team et al., 2024; Kim et al., 2024)—exactly what emergency-aware traffic control demands: the controller must see the whole scene, understand topology and intent expressed in language, and act

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.

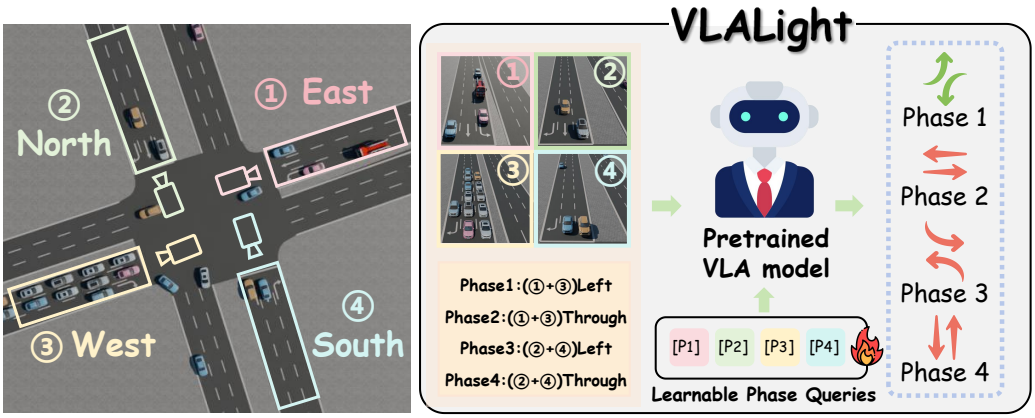


Figure 1: Overview of VLALight: a stitched multi-directional observation of the intersection and a textual phase instruction are mapped by a compact 0.5B vision-language-action backbone directly to the next signal phase in a single forward pass.

directly. Yet VLA research has so far focused on robot manipulation (Yu et al., 2025; Ye et al., 2025; Zhang et al., 2026); in this work, we bring the VLA paradigm to traffic signal control, where the action space is a small set of signal phases (Appendix E analyzes how intersection control differs from manipulation, driving, and UAV navigation).

In TSC scenarios, however, VLA models face several new challenges. *First, the observation structure differs*: a traffic controller must process multiple directional views at once, and each phase decision affects only the traffic movements visible in some of the views. *Second, intersection topologies vary* in shape and lane count, yet a single policy must serve all of them, and ideally unseen ones as well. *Third, the control objective is event-driven*: the controller must spot special vehicles such as ambulances and fire engines in dense ordinary traffic and immediately shift phase priority in their favor; such long-tail events are precisely where compact state statistics are weakest. Real-time operation imposes a further constraint on model size and adaptation cost.

We address these challenges with VLALight, a lightweight end-to-end VLA framework for intersection signal control. Because real surveillance footage paired with control labels is difficult to obtain at scale, we first build a simulation-grounded data-generation pipeline that replicates the lane-level topology of real intersections, designs diverse traffic-flow patterns with injected emergency vehicles, and renders high-fidelity camera images with expert labels derived from the simulator's ground-truth state. On the modeling side, VLALight stitches the directional camera views of an intersection into a single panoramic observation and pairs it with a synthesized textual instruction that establishes the correspondence between camera views, traffic movements, and signal phases. Conditioned on these inputs, a compact 0.5B multimodal backbone, adapted with LoRA through the VLA-Adapter paradigm (Wang et al., 2026) and augmented with learnable phase queries, directly decides the next phase in a single forward pass (Figure 1). Our contributions are:

- A simulation-grounded visual data-generation pipeline for TSC that replicates real-world intersection topologies, synthesizes traffic flows with injected emergency vehicles, derives expert labels from the simulator's ground-truth state, and renders high-fidelity multi-view images.
- A unified multi-directional visual representation and a TSC-native, instruction-grounded phase-classification action space supporting direct action prediction across intersections with different phase counts.
- VLALight, a lightweight end-to-end VLA framework that maps intersection observations and a textual phase configuration directly to discrete signal actions in a single forward pass on a local GPU.

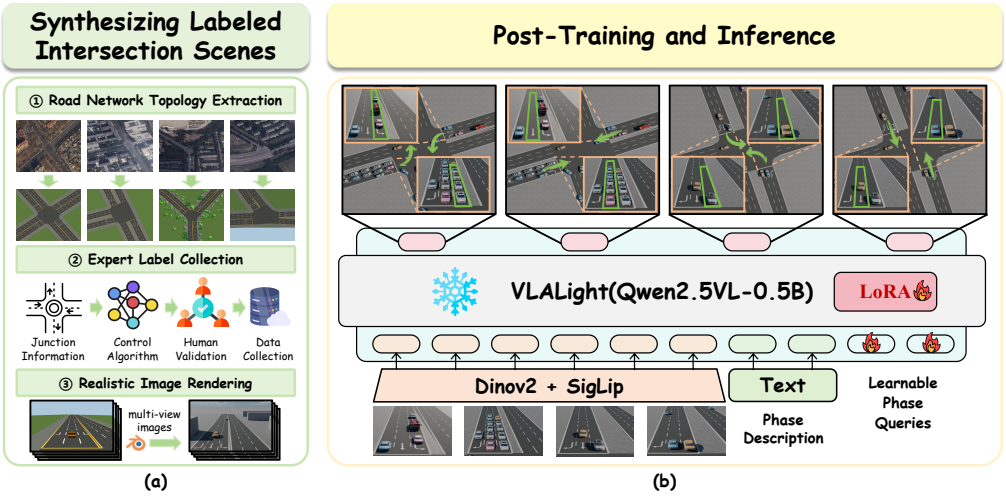


Figure 2: Architecture of VLALight. (a) The simulation-grounded traffic data-generation pipeline: road network topology extraction, traffic flow design and expert label collection, and realistic intersection image rendering. (b) Post-training and inference: the stitched multi-directional observation and the text instruction are encoded by the frozen DINOv2 and SigLIP visual encoders and the compact Qwen2.5-based VLM backbone, which directly outputs the next signal phase.

2 Related Work

Traffic signal control. Control strategies have progressed from fixed-time plans derived from historical statistics (Webster, 1958) to adaptive rule-based methods that react to real-time measurements (Gershenson, 2005; Varaiya, 2013) and deep reinforcement learning (Clifton & Laber, 2020; van Hasselt et al., 2016; Huang et al., 2018; Schulman et al., 2017; Wei et al., 2019b; Chen et al., 2020; Wei et al., 2018; 2019a; Oroojlooy et al., 2020; Wang et al., 2024b). All of these controllers consume compact numerical state statistics and none of them perceives or prioritizes emergency vehicles; we return to the implications for fair comparison in Section 4.1.

LLMs and VLMs for traffic control. To bring semantic reasoning into TSC, recent studies have turned to LLMs and VLMs (Nie et al., 2025). Early systems encode structured detector states as text and prompt an LLM for control decisions (Lai et al., 2025; Masri et al., 2025; Wang et al., 2024a); PromptGAT instead uses an LLM to produce traffic-dynamics descriptions that ground a learned state encoder for sim-to-real transfer, rather than letting the LLM select phases directly (Da et al., 2024). More recent designs couple visual encoders with language reasoning for signal control (Wang et al., 2025a; Zou et al., 2026). These systems still compress the scene into text before acting; VLALight removes this intermediate conversion.

Vision-language-action models. VLA models cast control as generative action prediction. RT-1 (Brohan et al., 2023), RT-2 (Zitkovich et al., 2023), Octo (Octo Model Team et al., 2024), and OpenVLA (Kim et al., 2024) demonstrated this paradigm at scale in robotic manipulation, and subsequent work has improved efficiency (Pertsch et al., 2025; Yu et al., 2025), reasoning (Zhao et al., 2025; Ye et al., 2025; Zhang et al., 2026), action representations (Black et al., 2025; Chi et al., 2025; Zhao et al., 2023), and safety (Zhang et al., 2025), while compact architectures such as RoboFlamingo (Li et al., 2024), UniVLA (Bu et al., 2025), and OpenVLA-OFT (Kim et al., 2025) reduce model size or latency. VLALight instantiates the same paradigm for a new domain, with signal phases as actions and roadside multi-directional views as observations (Appendix E).

3 Method

This section formulates emergency-aware traffic signal control as a vision-language-action problem, describes the simulation-grounded data-generation pipeline, and presents the VLALight policy.

3.1 Problem Formulation

An intersection connects incoming and outgoing approaches, each carrying one or more lanes; a *movement* carries vehicles from an incoming approach to an outgoing one, and movements are grouped into *phases*, each a set of non-conflicting movements that receive the green signal simultaneously. We consider intersection control with a phase set of size K, which varies across intersections, and time is discretized into decision steps of length Δt. At each step t, the agent observes a stitched surveillance image $I_t \in \mathbb{R}^{H\times W\times 3}$ covering all approaching directions, together with a natural-language instruction L describing the intersection topology and the meaning of each signal phase, and directly outputs a discrete phase $a_t \in \{0, \ldots, K-1\}$, equivalently a one-hot vector $\mathbf{a}_t \in \{0,1\}^K$. These are the only inputs: the policy does not receive the currently active phase or the elapsed green time, and phase-holding behavior is enforced externally by the minimum-green dwell constraint described in Section 3.3. The goal is to learn a policy $\pi_\theta(a_t \mid I_t, L)$ that minimizes the average travel and waiting times of all vehicles while remaining responsive to emergency-vehicle priority requests.

3.2 Synthesizing Labeled Intersection Scenes from Real Topology

Real intersection footage paired with control labels is difficult to obtain at scale: raw surveillance video raises privacy concerns, and the executed signal decisions are rarely accessible, so usable supervision labels do not exist in the real world. We therefore generate training data in simulation while anchoring every stage to the real world, through three steps: replicating real road-network topology, designing traffic flows and collecting expert labels, and rendering realistic camera images (details in Appendix A).

3.2.1 Road Network Topology Extraction

We replicate six real-world intersections in the TransimHub simulation platform (Wang et al., 2025b) with their lane-level topology exactly preserved. For each site, we obtain the raw road-network data from OpenStreetMap and refine it against Google satellite imagery and street-view photos until the simulated network matches reality. The selected intersections cover both four-way cross and T-junction layouts and differ in lane count and phase design.

3.2.2 Expert Label Collection

For each intersection we design a *regular flow* that reproduces a typical day with moderate, smoothly varying volumes, and collect the expert demonstrations under it; additional demand patterns used only for evaluation are introduced in Section 4. We inject emergency vehicles (police cars, ambulances, and fire engines) with predefined types, entry times, and paths, and attach a control label to every decision step. Under routine demand, each step is labeled by a maximum-pressure expert (Varaiya, 2013), which selects the phase with the highest pressure—the aggregated difference between incoming and outgoing vehicle counts over the movements it serves. Max pressure is blind to emergency vehicles, so whenever an emergency vehicle is present, the label is instead read directly from the simulator's ground-truth state: the phase serving the emergency vehicle's current movement is selected, granting it priority passage. This labeling is possible because simulation exposes emergency-vehicle identity directly, whereas deployed detectors provide no such information to conventional TSC controllers. The dataset contains 78 emergency-event trajectories yielding 1,462 keyframe samples, alongside the routine MaxPressure-labeled frames collected under 20 regular flows per intersection (Appendix A.3).

3.2.3 Realistic Image Rendering

Raw simulator frames suffer from low resolution, blurry textures, and missing detail, and policies trained on them transfer poorly to real scenes. We therefore render every scene with Blender's high-fidelity rendering engine, producing realistic lighting, materials, and vehicle appearance that narrow the sim-to-real visual gap. Each intersection is instrumented with up to four cameras, one per approach, at roadside surveillance viewpoints, matching how intersections are actually monitored.

3.3 POST-TRAINING AND INFERENCE

VLALight is built by post-training a frozen pretrained VLM rather than training from scratch. We adopt the VLA-Adapter paradigm (Wang et al., 2026) and inherit its Prism-Qwen2.5 backbone—frozen DINOv2 and SigLIP visual encoders feeding a Qwen2.5-0.5B language model—together with its LoRA adaptation recipe (Hu et al., 2022) and its query-conditioned action head with gated bridge attention. The remainder of this section describes how this general recipe is adapted to traffic signal control: a stitched multi-directional visual representation (Section 3.3.1), an instruction-grounded phase-classification action space with learnable phase queries (Section 3.3.2), and the policy-learning and inference procedure that produces the next phase in a single forward pass (Section 3.3.3).

3.3.1 MULTI-DIRECTIONAL VISUAL REPRESENTATION

A traffic controller sees the world very differently from a robot: intersections are monitored by roadside cameras, one per approach, and choosing a phase is a trade-off among competing directions, since granting green time to the movements in one view withholds it from all others. We therefore give the model the joint view that human operators watch on a surveillance wall, composing a single panoramic observation by stitching the directional views into a 2×2 grid:

$$I_t = \mathrm{Stack}_{2\times 2}\left(\{C_t^{(i)}\}_{i=0}^{3}\right) \in \mathbb{R}^{H\times W\times 3}, \tag{1}$$

where $C_t^{(i)} \in \mathbb{R}^{H_c\times W_c\times 3}$ is the i-th camera view and missing views (e.g., at T-junctions) are zero-padded; a scalar count of valid views accompanies the input so that the tokens of a padded black view are masked out (Appendix A.2). The stitched image is resized and normalized as a whole and encoded in a single pass by two frozen encoders, with DINOv2 contributing self-supervised geometric features and SigLIP language-aligned semantic features (Oquab et al., 2024; Zhai et al., 2023; Radford et al., 2021). At the native input resolution of the ViT/14 backbones, the stitched image yields 256 patch tokens in a 16×16 grid whose 8×8 blocks inherit the 2×2 layout of the stitched image, so the instruction can ground every token block to a direction, its movements, and its phase. All 256 tokens enter the language model as one sequence within a single forward pass of the 0.5B backbone, where self-attention contextualizes every visual token by all views at once:

$$\mathbf{v} = \left[\, \mathbf{v}^{\text{dino}} \,\|\, \mathbf{v}^{\text{siglip}} \,\right], \quad \mathbf{Z} = \mathrm{MLP}_{\text{proj}}(\mathbf{v}), \tag{2}$$

where $\mathbf{v}^{\text{dino}}$ and $\mathbf{v}^{\text{siglip}}$ are the patch features of the stitched image from the two encoders, $\|$ denotes concatenation along the feature dimension, and $\mathbf{Z}$ holds the N_p visual tokens projected to the LLM hidden size. The instruction L concisely describes the intersection, covering the approaching direction monitored by each camera and the traffic movement enabled by each phase index, inserted into a fixed prompt template (Appendix A.2); exposing topology through text lets a single policy serve intersections with different layouts, lane counts, and phase counts by only changing the instruction.

3.3.2 TSC-NATIVE ACTION SPACE

In signal control, the decision itself is the choice of a phase: at each step the controller selects the next phase from a small discrete set rather than emitting continuous signals. We therefore cast the decision as phase classification, formulated as token prediction in the same representation space as vision and language (Zitkovich et al., 2023; Kim et al., 2024). Concretely, each action dimension is tied to a reserved token from the end of the LLM vocabulary (Yang et al., 2024), occupied during training by a shared set of N_a learnable phase queries, decoupling the action representation from the tokenizer vocabulary. The number of queries is fixed ($N_a = 64$) and independent of the per-intersection phase count K: the action head maps the phase-query hidden states to the K phase logits $\hat{\mathbf{a}} \in \mathbb{R}^K$ defined by the instruction, and the predicted phase is $\hat{p}_t = \mathrm{argmax}_j\ \hat{a}_{t,j}$ over the K valid phases only, so invalid or padded directions can never be selected. At deployment, the prediction passes through a minimum-green dwell constraint: once activated, a phase is held for a minimum duration before any switch is permitted, mirroring real signal-timing rules and preventing oscillation between locally attractive phases.

3.3.3 POLICY LEARNING

A frozen pretrained VLM, comprising dual vision encoders and a compact language model, is augmented with four lightweight trainable modules: a multilayer projector, LoRA adapters, the phase queries $\mathbf{Q}$, and a residual-MLP action head; Algorithm 1 in Appendix B.2 gives the complete procedure and all hyperparameters. At each decision step, the projected visual tokens $\mathbf{Z}$ are placed right after the beginning-of-sequence embedding so that every later position can attend to the full visual context under the causal mask, followed by the instruction tokens and the phase queries:

$$\mathbf{E} = [\mathbf{e}_{\text{BOS}}; \mathbf{Z}_1; \ldots; \mathbf{Z}_{N_p}; \mathbf{e}_1; \ldots; \mathbf{e}_{N_L}; \mathbf{q}_1; \ldots; \mathbf{q}_{N_a}; \mathbf{e}_{\text{EOS}}], \quad (3)$$

where $\mathbf{e}_{\text{BOS}}$ and $\mathbf{e}_{\text{EOS}}$ are the beginning- and end-of-sequence embeddings, $\mathbf{e}_1, \ldots, \mathbf{e}_{N_L}$ are the embeddings of the N_L instruction tokens, and only the phase-query positions are supervised. A zero-initialized action state $\mathbf{x}^{(0)}$ is refined through residual blocks paired one-to-one with the LLM layers; each block performs a tri-branch attention that fuses, under a single softmax, keys and values from three sources (the action state itself, the phase-query hidden states, and the visual hidden states):

$$\text{Attn}(\mathbf{x}) = \text{softmax}\left(\frac{\mathbf{q}\,[\mathbf{k}_x;\, \mathbf{k}_a;\, \tanh(g)\,\mathbf{k}_v]^\top}{\sqrt{d_h}}\right) \cdot [\mathbf{v}_x;\, \mathbf{v}_a;\, \tanh(g)\,\mathbf{v}_v], \quad (4)$$

where $\mathbf{q}$ is a linear projection of the action state $\mathbf{x}$; $(\mathbf{k}_x, \mathbf{v}_x)$, $(\mathbf{k}_a, \mathbf{v}_a)$, and $(\mathbf{k}_v, \mathbf{v}_v)$ are the key-value pairs computed from the action state, the phase-query hidden states, and the visual hidden states; and d_h is the head dimension. The visual branch is gated by $\tanh(g)$ in both keys and values, with a learnable scalar g initialized to zero: at $g = 0$ the visual keys contribute zero logits and the visual values zero content, so the gate fully suppresses the visual branch regardless of the softmax weights, and visual context is incorporated gradually as g moves away from zero. After a residual connection and a feed-forward layer, a final linear projection maps the refined action state to the phase logits, $\hat{\mathbf{a}} = \mathbf{W}_o\,\mathbf{x}^{(L)} + \mathbf{b}_o$. Training is behavioral cloning of the expert labels: given the one-hot ground-truth action $\mathbf{a}^*$, we minimize the cross-entropy over the K phases,

$$\mathcal{L}_{\text{CE}} = -\sum_{j=0}^{K-1} a_j^* \log \frac{\exp(\hat{a}_j)}{\sum_{k=0}^{K-1} \exp(\hat{a}_k)}, \quad (5)$$

updating only the lightweight modules $\theta = \{\theta_{\text{proj}}, \theta_{\text{LoRA}}, \mathbf{Q}, \theta_{\text{head}}\}$ while the vision encoders and the base LLM remain frozen, preserving the visual-linguistic priors of large-scale pretraining (Liu et al., 2023; Li et al., 2023; Radford et al., 2021).

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

4.1.1 DATASET AND SIMULATION

We evaluate VLALight on the six real-world intersections introduced in Section 3, rendered in TransimHub (Wang et al., 2025b); Figure 3 pairs each real intersection with its simulated counterpart. A single model is trained and evaluated across all sites in closed-loop simulation, on the *regular flow* used to collect the expert demonstrations and three *test flows* (*off-peak*, *rush-hour*, and *fluctuating*). Each episode lasts 600 s; the controller issues a decision every 5 s; every green phase is followed by a fixed 3 s yellow/all-red clearance inserted by the simulator identically for all controllers; and the minimum-green dwell constraint holds each phase for at least 15 s. Every city–flow pair is evaluated under four emergency-vehicle injection sets (set1–set4) shared identically across all compared methods; we report their mean and standard deviation. The variation therefore reflects event-set variability.

4.1.2 EVALUATION METRICS

We report the mean and standard deviation of four metrics: Average Travel Time (ATT) and Average Waiting Time (AWT) for routine efficiency, and Emergency Travel Time (ETT) and Emergency Waiting Time (EWT) for emergency-vehicle service, all in seconds (lower is better).



Figure 3: The six real-world intersections in our dataset. For each site, the satellite view of the real intersection (left) is paired with its rendered simulation (right).

Table 1: Pooled comparison on the regular flow (mean±std over the six intersections and four emergency-injection sets, in seconds; lower is better). The top-3 values per metric are boldfaced and ranked */†/‡. VLALight attains the best emergency metrics (ETT, EWT) while VLMLight (32B) attains the best routine metrics (ATT, AWT), making the routine-efficiency trade-off explicit; per-intersection and per-flow breakdowns of all four metrics are given in Appendices C.1 and C.2.

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	47.9±5.8	13.7±4.4	51.9±14.9	16.3±12.5
	Webster	48.1±5.9	13.8±4.4	52.2±15.0	16.6±12.6
	SOTL	47.6±3.2	13.7±3.3	50.9±11.6	15.5±10.6
	MaxPressure	**47.1±4.7**‡	**12.9±3.2**‡	55.9±18.0	20.1±15.4
RL	IntelliLight	51.3±9.4	14.7±6.3	60.8±32.4	22.6±25.3
	PressLight	68.5±17.8	27.9±13.9	76.1±27.8	32.8±19.6
	AttendLight	54.0±8.7	17.7±8.0	62.3±21.7	24.6±18.6
	UniTSA	**45.8±3.7**†	**10.91±3.2***	**47.7±10.8**†	**12.5±8.7**‡
VLM	VLMLight (32B)	**45.2±3.6***	**11.2±3.1**†	**48.1±10.8**‡	**12.5±8.6**†
	VLMLight (3B)	77.2±15.5	32.1±10.5	93.0±46.4	42.0±32.3
Ours	**VLALight** (0.5B)	53.0±5.6	19.2±5.4	**44.7±11.6***	**9.84±9.3***

4.1.3 Compared Methods

We compare against nine controllers from three families: four rule-based (FixTime, Webster (Webster, 1958), SOTL (Gershenson, 2005), MaxPressure (Varaiya, 2013)); four RL-based (IntelliLight (Wei et al., 2018), PressLight (Wei et al., 2019a), AttendLight (Oroojlooy et al., 2020), UniTSA (Wang et al., 2024b)); and the VLM-based VLMLight (Wang et al., 2025a) with two backbone sizes (cloud 32B and a local 3B variant). VLALight uses a 0.5B backbone and runs entirely locally. Baseline configurations are detailed in Appendix B.3.

4.2 Main Results

Table 1 reports the pooled comparison under the regular flow with all four metrics; per-city, per-intersection, and per-flow breakdowns appear in Appendices C.1 and C.2. VLALight provides the strongest emergency-vehicle service among the compared controllers on the regular flow: it attains the lowest ETT and EWT in four of the six cities and, pooled over all six cities, improves on the cloud-scale VLMLight (32B) by 7.0% in ETT and 21.1% in EWT. The two exceptions are Beijing Changjianglu and Tianjin Zhijingdao, where rule-based controllers or VLMLight (32B) rank ahead. Appendix D (Figure 8) shows a concrete example: VLALight switches to the phase serving an approaching emergency vehicle at the next decision step.

The rule- and RL-based baselines consume compact state statistics (counts, queues, pressure) that carry no emergency identity, so they cannot respond to emergency vehicles, although they attain strong routine performance. VLMLight is the only compared controller that serves both routine traffic and emergency vehicles well, but its cascaded design relies on a large backbone and incurs high decision latency. Our emergency comparison therefore focuses on VLMLight, which shares

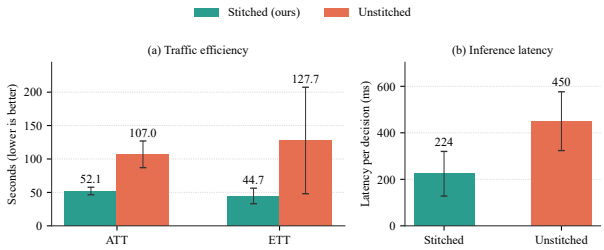


Figure 4: Ablation of the visual-input design: stitched vs. unstitched visual input. (a) Pooled ATT and ETT under the regular flow; (b) inference latency per decision.

the same emergency-aware objective, while the rule- and RL-based controllers serve as references for routine efficiency.

Routine-efficiency trade-off. The emergency gains are not free. Pooled over the six intersections, VLALight's ATT and AWT are 53.0 s and 19.2 s, versus 45.2 s and 11.2 s for VLMLight (32B)—17.1% and 71.9% higher—and VLMLight (32B) also attains the best routine metrics overall (Table 1). This trade-off is the deliberate consequence of prioritizing emergency vehicles under a compact single-frame model, and it concentrates in low-demand regimes (Section 4.5).

Statistical interpretation. The pooled 21.1% EWT reduction corresponds to an improvement from 12.5 s to 9.8 s per emergency vehicle, and several city-level margins are smaller than the corresponding standard deviations. The improvement is nevertheless consistent in sign across cities and across the heavier flows (Section 4.5), and it is obtained with a 64× smaller backbone at a fraction of the latency; we interpret it as a consistent, practically meaningful but modest effect rather than uniform dominance.

4.3 Inference Latency and Deployment Analysis

Real-time deployment hinges on inference latency. Emergency-aware service is exactly where VLMLight's slow branch fires: the cascaded design chains several agent calls for a single decision, costing 10.9 s with the cloud 32B backbone and 4.2 s on average with a locally deployed 3B backbone, both close to or beyond the 5 s decision interval. VLALight instead outputs the action in a single local forward pass, averaging 224 ms per decision on one RTX 4090 GPU, supporting fully on-premises deployment; it is the only compared controller that stays comfortably within the decision budget with emergency awareness always enabled. Measurement protocols and per-agent latency breakdowns are given in Appendix C.3.

4.4 Ablation Study

As Figure 4 shows, we ablate the visual-input design by comparing our single stitched image against feeding the four camera views as four separately encoded images, which quadruples the visual-token count from 256 to 1,024. The single stitched input wins in every city and every injection set under the regular flow, reducing pooled ATT from 107.0 s to 52.1 s and ETT from 127.7 s to 44.7 s, and it is twice as fast at inference (224 ms versus 450 ms per decision). Two qualifications delimit this result: the two arms differ simultaneously in layout, token count, and per-view spatial sampling, so the result validates the stitched input as the better *operating point* for a compact backbone, not joint encoding as an isolated causal mechanism; and since both arms retain vision, the value of direct visual input over textual scene conversion is instead supported by the matched-objective comparison against the caption-based VLMLight cascade in Table 1. The minimum-green dwell constraint is ablated in Appendix C.4.

4.5 Generalization Analysis

We assess generalization along two axes: unseen traffic-flow patterns and unseen intersection topologies.

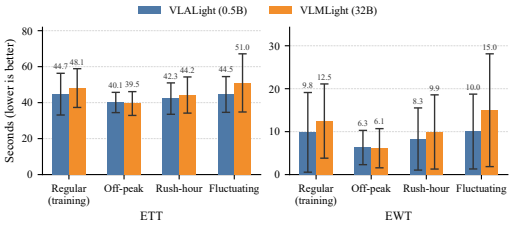


Figure 5: Unseen-flow generalization.

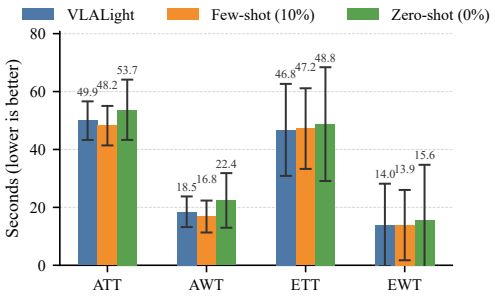


Figure 6: Unseen-topology transfer.

Generalization to unseen traffic flows. Without retraining, we evaluate the same model on the three test flows (*off-peak*, *rush-hour*, and *fluctuating*; per-flow results in Appendix C.2). As Figure 5 shows, pooled over the three test flows, VLALight retains the strongest emergency service, improving on VLMLight (32B) by 5.8% in ETT and 20.8% in EWT, and the advantage grows as conditions depart from the training distribution: on par under the *off-peak flow*, 16.5% lower EWT under the *rush-hour flow*, and 33.1% under the *fluctuating flow*. The off-peak flow is the one regime in which VLALight does not lead: there VLMLight (32B) retains a slight edge (1.5% lower ETT and 2.6% lower EWT), because demand is low enough that cloud round-trips do not miss the decision window.

Generalization to unseen topologies. Figure 6 compares three variants on two unseen test sites, Beijing Beishahe and Tianjin Zhijingdao: the fully trained model, a few-shot variant fine-tuned with only 10% of the target intersections' data, and a zero-shot variant given no target data. The few-shot variant stays within 1–3% of the fully trained model on ETT and EWT, and even slightly improves ATT and AWT. Without any target data, the zero-shot variant remains competitive: its ETT and EWT rise by only 4.3% and 11.4%, and ATT by 7.6%. VLALight thus learns transferable visual-language priors rather than memorizing the training intersections.

5 Discussion and Conclusion

We presented VLALight, a lightweight vision-language-action framework for traffic signal control: a compact 0.5B model maps a stitched multi-directional observation and a textual phase instruction directly to discrete signal actions in a single forward pass. In closed-loop simulation on six replicas of real-world intersections, VLALight attains the strongest emergency-vehicle service among the compared controllers, answers each decision in 224 ms on a local GPU, and generalizes to unseen topologies and flows. A discussion of limitations and open directions is provided in Appendix F.

Ethics Statement

Traffic signal control is safety-critical infrastructure; emergency-priority policies must be validated in the real world before deployment, and our simulation results do not imply deployment readiness. All data are synthetic; deployment would require privacy safeguards for live camera feeds.

Reproducibility Statement

Appendices document the evaluation protocol (B.1), training configuration (B.2), baselines (B.3), data generation (A.3), and full per-intersection and per-flow results (C.1, C.2). Code and configurations will be released.

AI Use Statement

AI assistants were used only for copyediting. No ideas, results, or figures were AI-generated; we take responsibility for the final content.

REFERENCES

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2025.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael Ryoo, Grecia Salazar, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale. In *Proceedings of Robotics: Science and Systems*, 2023.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2025.

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.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10–11):1684–1704, 2025. doi: 10.1177/02783649241273668.

Jesse Clifton and Eric B. Laber. Q-learning: Theory and applications. *Annual Review of Statistics and Its Application*, 7(1):279–301, 2020.

Longchao Da, Minquan Gao, Hao Mei, and Hua Wei. Prompt to transfer: Sim-to-real transfer for traffic signal control with prompt learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 82–90, 2024.

Carlos Gershenson. Self-organizing traffic lights. *Complex Systems*, 16(1):29–53, 2005.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nzeVKeeFYf9.

Ying Huang, Guoliang Wei, and Yongxiong Wang. V-d d3qn: The variant of double deep q-learning network with dueling architecture. In *2018 37th Chinese Control Conference (CCC)*, pp. 9130–9135, 2018.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P. Foster, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. In *Proceedings of The 8th Conference on Robot Learning*, volume 270, pp. 2679–2713. PMLR, 2024.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2025.

Siqi Lai, Zhao Xu, Weijia Zhang, Hao Liu, and Hui Xiong. Llmlight: Large language models as traffic signal control agents. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2335–2346. ACM, 2025. doi: 10.1145/3690624.3709379.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 19730–19742, 2023.

Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations*, 2024.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023.

Sari Masri, Huthaifa I. Ashqar, and Mohammed Elhenawy. Large language models (llms) as traffic control systems at urban intersections: A new paradigm. *Vehicles*, 7(1):11, 2025. doi: 10.3390/vehicles7010011.

Tong Nie, Jian Sun, and Wei Ma. Exploring the roles of large language models in reshaping transportation systems: A survey, framework, and roadmap. *Artificial Intelligence for Transportation*, 1:100003, 2025.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Afshin Oroojlooy, Mohammadreza Nazari, Davood Hajinezhad, and Jorge Silva. Attendlight: Universal attention-based reinforcement learning model for traffic signal control. In *Advances in Neural Information Processing Systems*, volume 33, pp. 4079–4090, 2020.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763, 2021.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), 2016.

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.

Maonan Wang, Aoyu Pang, Yuheng Kan, Man-On Pun, Chung Shue Chen, and Bo Huang. Llm-assisted light: Leveraging large language model capabilities for human-mimetic traffic signal control in complex urban environments, 2024a.

Maonan Wang, Xi Xiong, Yuheng Kan, Chengcheng Xu, and Man-On Pun. Unitsa: A universal reinforcement learning framework for v2x traffic signal control. *IEEE Transactions on Vehicular Technology*, 73(10):14354–14369, 2024b.

Maonan Wang, Yirong Chen, Aoyu Pang, Yuxin Cai, Chung Shue Chen, Yuheng Kan, and Man-On Pun. Vlmlight: Safety-critical traffic signal control via vision-language meta-control and dual-branch reasoning architecture. In *Advances in Neural Information Processing Systems*, volume 38, 2025a.

Maonan Wang, Aoyu Pang, Yuejiao Xie, Man-On Pun, Yirong Chen, Ding Wang, Yuxin Cai, Zian Ma, Chengcheng Xu, Yuheng Kan, Kemou Jiang, Zhiyong Cui, Laurent Roullet, Chung Shue Chen, and Michael Lepech. Transimhub: A unified air–ground simulation platform for multi-modal perception and decision-making, 2025b.

Yihao Wang, Pengxiang Ding, Lingxiao Li, Can Cui, Zirui Ge, Xinyang Tong, Wenxuan Song, Han Zhao, Wei Zhao, Pengxu Hou, Siteng Huang, Yifan Tang, Wenhui Wang, Ru Zhang, Jianyi Liu, and Donglin Wang. Vla-adapter: An effective paradigm for tiny-scale vision-language-action model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18638–18646, 2026.

F. V. Webster. Traffic signal settings. Road Research Technical Paper 39, Road Research Laboratory, London, 1958.

Hua Wei, Guanjie Zheng, Huaxiu Yao, and Zhenhui Li. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2496–2505. ACM, 2018.

Hua Wei, Chacha Chen, Guanjie Zheng, Kan Wu, Vikash V. 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. ACM, 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 *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1913–1922, 2019b.

Hua Wei, Guanjie Zheng, Vikash V. Gayah, and Zhenhui Li. Recent advances in reinforcement learning for traffic signal control: A survey of models and evaluation. *ACM SIGKDD Explorations Newsletter*, 22(2):12–18, 2021.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report, 2024.

Angen Ye, Zeyu Zhang, Boyuan Wang, Xiaofeng Wang, Dapeng Zhang, and Zheng Zhu. Vla-r1: Enhancing reasoning in vision-language-action models, 2025.

Zhaoshu Yu, Bo Wang, Pengpeng Zeng, Haonan Zhang, Ji Zhang, Lianli Gao, Jingkuan Song, Nicu Sebe, and Heng Tao Shen. A survey on efficient vision-language-action models, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11975–11986, 2023.

Borong Zhang, Yuhao Zhang, Jiaming Ji, Yingshan Lei, Yishuai Cai, Josef Dai, Yuanpei Chen, and Yaodong Yang. Safevla: Towards safety alignment of vision-language-action model via constrained learning. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Jianke Zhang, Xiaoyu Chen, Qiuyue Wang, Mingsheng Li, Yanjiang Guo, Yucheng Hu, Jiajun Zhang, Shuai Bai, Junyang Lin, and Jianyu Chen. Vlm4vla: Revisiting vision-language-models in vision-language-action models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, Ankur Handa, Tsung-Yi Lin, Gordon Wetzstein, Ming-Yu Liu, and Donglai Xiang. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1702–1713. IEEE, 2025.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proceedings of Robotics: Science and Systems*, 2023.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Wenhao Huang, Stefan Welker, Karol Hausman, Jack Orbay, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Proceedings of The 7th Conference on Robot Learning*, pp. 2165–2183. PMLR, 2023.

Xingchen Zou, Yuhao Yang, Zheng Chen, Xixuan Hao, Yiqi Chen, Chao Huang, and Yuxuan Liang. Traffic-r1: Reinforced LLMs bring human-like reasoning to traffic signal control systems. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 21823–21838, San Diego, California, United States, 2026. Association for Computational Linguistics.

A Dataset Details

A.1 Intersection Topologies, Cameras, and Signal Phases

Table 2 summarizes the six simulated intersections, and Figure 3 pairs each real intersection with its simulated counterpart. All of them are rebuilt from real-world road networks in Beijing and Tianjin: the raw topology is extracted from OpenStreetMap and then corrected against Google satellite and street-view imagery, so that the lane geometry and the lane-function assignment of every approach match reality. The set deliberately mixes crossroads and T-junctions with different lane counts.

Each incoming approach of a four-phase crossroad is monitored by a camera placed about 15 m above ground with a 90° field of view, yielding the four directional views $\{I_N, I_E, I_S, I_W\}$. The three-phase T-junctions use the same per-camera height and FOV but only three active views; the missing direction is kept as an all-black image so that the input layout remains identical across intersections. Cameras render at 720×480; frames are center-cropped and resized to 256×256, and the available directional views are stitched into a single 512×512 composite that is fed to the model.

Every intersection runs a fixed ring of through and left-turn green phases; each green phase is followed by a 3 s yellow/all-red clearance phase. The model only selects among the green phases, and the clearance phases are inserted by the simulator automatically and identically for every compared controller.

Table 2: Topology, camera, and signal-program summary of the six intersections. Ap.: number of incoming approaches; Ln.: total incoming lanes; Ph.: number of selectable green phases; Cam.: number of approach cameras.

Intersection	City	Type	Ap.	Ln.	Ph.	Cam.
Beihuan	Beijing	T-junction	3	6	3	3
Beishahe	Beijing	T-junction	3	9	3	3
Changjianglu	Beijing	T-junction	3	9	3	3
Gaojiaoyuan	Beijing	crossroad	4	12	4	4
Pinganli	Beijing	crossroad	4	14	4	4
Zhijingdao	Tianjin	crossroad	4	12	4	4

A.2 Instruction Templates

Each decision sample carries a natural-language instruction assembled from three fields: the composite image layout, the control objective, and the phase-to-image mapping of the intersection.

The instruction grounds the visual input for the policy and is inserted into a fixed prompt wrapper before tokenization:

> "What action should the robot take to {instruction}?"

The full instruction text of the four-phase intersection Gaojiaoyuan reads:

> "The input is a single composite image. Top-Left shows Image 1, Top-Right shows Image 2, Bottom-Left shows Image 3, Bottom-Right shows Image 4. Keep the intersection clear and minimize average waiting time, prioritizing special vehicles. Intersection Beijing_Gaojiaoyuan phase configuration: Phase 1: Through & left lanes in Image 4; Phase 2: Through & left lanes in Image 3; Phase 3: Through & left lanes in Image 2; Phase 4: Through & left lanes in Image 1."

For the three-phase T-junctions the same template is used with only three phase mappings, as in Beishahe:

> "The input is a single composite image. Top-Left shows Image 1, Top-Right shows Image 2, Bottom-Left shows Image 3, Bottom-Right shows Image 4. Keep the intersection clear and minimize average waiting time, prioritizing special vehicles. Intersection Beijing_Beishahe phase configuration: Phase 1: Through & left lanes in Image 3; Phase 2: Through & left lanes in Image 1; Phase 3: Through & left lanes in Image 2."

Image 4 is then an all-black frame that corresponds to no valid phase; a scalar `num_valid_images` in the state vector records the number of active views, so the tokens of the black view are explicitly invalidated and the model never selects the missing direction.

The same template is used at training and inference, so transferring the policy to a new intersection only requires rewriting the instruction. Instructions are deterministic functions of the intersection topology and are generated once per site; no instruction text is seen by the vision encoders, and the policy must ground the text in the stitched image through the shared token sequence. The instruction text is tokenized together with the special tokens `<|im_start|>`, `<|im_end|>`, `<|vision_start|>`, and `<|vision_end|>`, and the projected visual tokens of the stitched image (Section 3.3.1) are spliced between the vision markers, so the directional views are interleaved with the textual instruction as a single sequence.

A.3 Expert Demonstration Generation

For each scenario we design four distinct event injection sets and collect expert demonstrations under 20 randomly generated regular flows. From the emergency scenarios we extract key decision frames around each special-vehicle event, which yields 78 extracted trajectories and 1,462 keyframe decision samples after compression.

In the regular scenarios, actions are labeled by MaxPressure (Varaiya, 2013). In the event scenarios, the label at each emergency-relevant keyframe is read directly from the simulator's ground-truth state: the phase serving the emergency vehicle's current movement is selected, so the vehicle is granted priority passage; the remaining frames keep the MaxPressure labels. Because emergency-vehicle identity is available directly from the simulation state, these labels require no perception or recognition step—a capability that conventional TSC controllers, which receive only counts and queues from deployed detectors, do not have.

B Experimental Setup Details

B.1 Simulation and Evaluation Protocol

All experiments run in closed-loop simulation in the SUMO-based TransimHub environment. Each episode lasts 600 s of simulated time; the policy issues a decision every $t = 5$ s, and a minimum-green dwell constraint holds each phase for at least 15 s (three decision intervals) before a switch is permitted. Every green phase is followed by a 3 s yellow/all-red clearance phase inserted by the simulator, identically for all compared controllers. Every city–flow pair is evaluated with four evaluation seeds (set1–set4), which correspond to four distinct emergency-vehicle injection sets that

are shared identically across all compared methods; we report the mean and standard deviation over all intersections and seeds. Because all runs evaluate the same trained checkpoint, the reported standard deviations reflect event-set variability rather than training-run variance.

Table 3: Per-edge average inflow rates of the four traffic-flow patterns for two representative intersections. The total intersection demand is the sum of the listed per-edge rates over all incoming approaches. The rush-hour row corresponds to the increasing-demand flow. Rates are computed from the generated route files as vehicles per minute averaged over a 10-minute episode.

Intersection	Flow	Edge inflow (veh/min)	Total (veh/min)
Beishahe	regular	[15.0, 15.2, 16.6]	46.8
	off-peak	[8.0, 6.0, 7.0]	21.0
	rush-hour	[17.6, 15.0, 15.0]	47.6
	fluctuating	[15.2, 11.6, 14.8]	41.6
Gaojiaoyuan	regular	[14.4, 13.6, 14.6, 12.6]	55.2
	off-peak	[6.2, 7.4, 7.4, 7.4]	28.4
	rush-hour	[15.0, 13.0, 13.2, 13.4]	54.6
	fluctuating	[13.6, 11.6, 9.0, 11.6]	45.8

Traffic-flow patterns. Flows are parameterized as vehicles per minute on each incoming edge. Table 3 lists the parameters of the four patterns for two representative intersections; the off-peak pattern lowers the regular rates, the rush-hour pattern raises them, and the fluctuating pattern assigns different rates to different movements to mimic directional imbalances observed on weekends. The complete configurations of all six intersections will be released with the code.

Special-vehicle injection. Each evaluation flow is paired with four injection sets that differ in the type, number, and departure time of the special vehicles. For example, in Beishahe Set 1 a police car departs at 45 s on one through movement and a second one at 87 s on another, while Set 4 combines three special vehicles departing at 55 s, 133 s, and 240 s with an accident that closes a westbound lane from 120 s to 420 s. Table 4 reports the total number of injected special vehicles per intersection across all sets and flows.

Table 4: Total number of injected special vehicles per intersection, summed over the four injection sets and the evaluation flows.

Intersection	Police	Ambulance	Fire eng.
Beihuan	8	8	8
Beishahe	8	12	12
Changjianglu	12	12	12
Gaojiaoyuan	12	16	16
Pinganli	12	12	12
Zhijingdao	12	12	12

B.2 Training Details and Infrastructure

Table 5 lists the key training hyperparameters. Training runs for 5,000 steps with an effective batch size of 128 and takes approximately 48 hours on four NVIDIA RTX 4090 GPUs; the vision encoders and the base LLM remain frozen throughout, and only the projector, LoRA adapters, phase queries, and action head are updated. Checkpoints are saved every 1,000 steps, and all reported results use the final checkpoint.

Visual encoders and projector. The fused vision backbone pairs a DINOv2 ViT-L/14 (1,024-dimensional patch features) with a SigLIP ViT-SO/14 (1,152-dimensional patch features), both frozen and both read out at the penultimate block. Patch features are concatenated along the feature dimension, giving 2,176-dimensional visual tokens, and the projector is a three-layer MLP ($2{,}176 \to 8{,}704 \to 896 \to 896$ with GELU activations) that maps them to the LLM hidden size of 896.

Algorithm 1 VLALight Training and Inference

Input: Expert demonstrations $\mathcal{D} = \{(I, L, \mathbf{a}^*)\}$ (stitched image, instruction, one-hot expert action)
Parameters: Vision encoder E, projector g_{proj}, LLM f_{LLM}, action head f_{head}, phase queries $\mathbf{Q}$, adapter parameters θ
Output: Policy π_θ

Initialize $\theta \leftarrow \{\theta_{\text{proj}}, \theta_{\text{head}}, \mathbf{Q}, \theta_{\text{LoRA}}\}$
Freeze E and f_{LLM}
while not converged **do**
  Sample a batch $\{(I^i, L^i, \mathbf{a}^{*i})\}_{i=1}^{B}$ of size B from $\mathcal{D}$
  $\mathbf{v}^i = E(I^i)$
  $\mathbf{Z}^i = g_{\text{proj}}(\mathbf{v}^i)$
  $\mathbf{E}^i = [\mathbf{e}_{\text{BOS}}; \mathbf{Z}^i; \text{Tokenizer}(L^i); \mathbf{Q}; \mathbf{e}_{\text{EOS}}]$
  $\hat{\mathbf{a}}^i = f_{\text{head}}\big(f_{\text{LLM}}(\mathbf{E}^i; \theta)\big)$
  Update θ by minimizing $\mathcal{L}_{\text{CE}}(\hat{\mathbf{a}}^i, \mathbf{a}^{*i})$
end while
for each evaluation step t **do**
  Observe (I_t, L_t); $\mathbf{Z}_t = g_{\text{proj}}\big(E(I_t)\big)$
  $\mathbf{E}_t = [\mathbf{e}_{\text{BOS}}; \mathbf{Z}_t; \text{Tokenizer}(L_t); \mathbf{Q}; \mathbf{e}_{\text{EOS}}]$
  $\hat{\mathbf{a}}_t = f_{\text{head}}\big(f_{\text{LLM}}(\mathbf{E}_t; \theta)\big)$
  Execute $\hat{p}_t = \text{argmax}_j\ \hat{a}_{t,j}$ over the K valid phases, subject to the minimum-green dwell constraint
end for
return π_θ

Phase queries and action head. The 64 phase queries are 896-dimensional learnable embeddings initialized to zero. Each query position replaces a reserved token at the end of the Qwen2.5 vocabulary (unused token ids starting at 151,386), and the cross-entropy loss is computed only at the query positions. The action head is a 24-block residual MLP: a zero-initialized action state is refined block by block, where each block runs a tri-branch attention with 8 heads (head dimension 112) over the action state itself, the phase-query hidden states, and the visual hidden states, the visual branch gated in both keys and values by $\tanh(g)$ with g initialized to zero, followed by a feed-forward layer; a final linear projection produces the K phase logits of the intersection at hand.

Optimization. We train in bfloat16 with AdamW (learning rate 5×10^{-5}), gradient clipping at max norm 1.0, and a multi-step schedule that decays the learning rate by 0.1 after the warm-up phase. LoRA adapters use Gaussian initialization and are inserted into all linear layers of the LLM. Checkpoints save the LoRA adapter, the action head, and the dataset statistics.

Table 5: Key training hyperparameters.

Parameter	Value	Parameter	Value
Base VLM	Prism-Qwen2.5	LoRA rank r	64
LLM	Qwen2.5-0.5B	LoRA α	128
Vis. enc.	DINOv2+SigLIP	LoRA target	all-linear
Projector	3-layer MLP	Action head	MLPResNet-Pro
Phase queries	64	Batch size	128
Learning rate	5×10^{-5}	Optimizer	AdamW
Grad. clip	1.0	Weight decay	0.0
Warmup steps	500	Max steps	5,000

Infrastructure. Table 6 reports the hardware and software environment used for training and evaluation.

B.3 Compared Methods

We compare VLALight against nine controllers spanning three families. Each baseline is briefly described below.

Table 6: Hardware and software environment.

Component	Specification
OS	Ubuntu Linux (kernel 5.15.0-126)
Python	3.10.16
CUDA	12.1 (driver 550.135)
PyTorch	2.2.0+cu121
Transformers	4.40.1
PEFT	0.11.1
GPU	4 × NVIDIA RTX 4090 (24 GB)

FixTime cycles through the phases with a fixed, precomputed green split.

Webster (Webster, 1958) derives the cycle length and green splits from historical flow ratios using Webster's delay-minimizing formula.

SOTL (Gershenson, 2005) is a self-organizing controller that switches phases according to real-time vehicle counts approaching the stop line.

MaxPressure (Varaiya, 2013) greedily activates the phase with the highest pressure, the aggregated difference between incoming and outgoing vehicle counts, and is a strong throughput-oriented baseline.

IntelliLight (Wei et al., 2018) learns a deep Q-network over image-like state representations with a phase-gated reward.

PressLight (Wei et al., 2019a) optimizes the pressure objective with reinforcement learning, linking RL rewards to the max-pressure theory.

AttendLight (Oroojlooy et al., 2020) replaces hand-crafted lane aggregation with an attention mechanism that weighs approaching vehicles.

UniTSA (Wang et al., 2024b) learns a universal, topology-agnostic policy that transfers across intersection layouts. All RL controllers are trained per scenario with the same flows and evaluation protocol as VLALight.

VLMLight (Wang et al., 2025a) couples a fast RL branch (a pretrained UniTSA/PPO controller) with a slow branch of cascaded VLM/LLM agents for emergency-aware reasoning; an image-captioning agent first converts the scene to text, over which language agents then analyze, select the control mode, and emit the phase. We evaluate it with two backbones: the cloud configuration (∼32B) and a locally deployed 3B variant, both under the same decision interval and emergency-vehicle injection as VLALight.

C Additional Results

C.1 Comparison Across Intersections

Averaged over the six intersections, VLALight consistently leads on emergency metrics while trading away some routine efficiency. Relative to the cloud-deployed VLMLight (32B), VLALight's ATT is 15.8% higher and AWT is 73% higher, but it cuts ETT by 6.1% and EWT by 20.9%. The gap widens against the locally deployed VLMLight (3B): VLALight lowers ATT by 19.0%, AWT by 21.3%, ETT by 39.5%, and EWT by 69.0%. Even against the strongest rule/RL baseline, MaxPressure, VLALight raises ATT by 12.8% yet reduces ETT by 13.3% and EWT by 40.7%. These overall figures confirm the intended design trade-off: a compact, single-frame model sacrifices part of its routine throughput to obtain substantially faster and more reliable emergency clearance.

The emergency gains are geographically consistent across the heterogeneous topology mix. On the four-lane crossroads Gaojiaoyuan and Pinganli, where queue spillback is most severe, VLALight

Table 7: Per-intersection breakdown of all four metrics (mean±std over 4 flows × 4 seeds, seconds). Values are grouped by method family; the top-3 values per block are boldfaced with ⋆, †, ‡.

Beijing Beihuan

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	46±6.2	10±4.3	55±15	17±10
	Webster	47±6.4	10±4.5	56±15	17±11
	SOTL	**45±1.6**‡	9.8±0.9	57±11	20±10
	MaxPressure	46±6.2	**9.4±4.3**‡	59±18	19±15
RL	IntelliLight	55±11	16±7.2	81±33	36±26
	PressLight	66±16	24±9.5	76±31	30±21
	AttendLight	47±4.3	10±3.2	57±16	18±14
	UniTSA	**44±4.2**†	**7.6±3.2**†	**49±9.6**⋆	**12±8.3**⋆
VLM	VLMLight (32B)	**43±4.2**⋆	**7.5±3.2**⋆	**53±12**‡	**14±9.6**‡
	VLMLight (3B)	71±25	23±15	98±44	41±27
Ours	**VLALight** (0.5B)	48±5.3	12±3.9	**50±8.1**†	**12±6.3**†

Beijing Gaojiaoyuan

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	48±4.1	16±3.3	62±11	27±9.5
	Webster	48±3.9	16±3.2	63±11	27±9.8
	SOTL	50±4.6	17±3.4	61±15	25±12
	MaxPressure	**47±3.3**†	**15±2.5**‡	61±10	25±8.6
RL	IntelliLight	51±7.3	16±5.8	**53±8.2**‡	**18±6.7**†
	PressLight	74±14	36±11	101±38	57±31
	AttendLight	70±24	36±25	90±37	52±36
	UniTSA	**47±5.6**‡	**14±4.6**†	58±12	22±9.9
VLM	VLMLight (32B)	**46±4.6**⋆	**13±3.9**⋆	**53±11**†	**18±9.4**‡
	VLMLight (3B)	67±18	28±13	91±32	45±23
Ours	**VLALight** (0.5B)	53±3.1	21±2.5	**49±7.7**⋆	**15±6.6**⋆

Beijing Beishahe

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	41±2.0	10±1.3	45±4.2	13±3.7
	Webster	40±1.9	9.9±1.1	45±3.3	12±3.2
	SOTL	46±2.8	15±3.4	47±6.7	15±6.7
	MaxPressure	**39±2.3**‡	8.9±1.6	42±4.9	10.0±4.3
RL	IntelliLight	**39±4.1**†	**7.3±2.8**†	41±6.5	8.1±4.9
	PressLight	50±19	17±18	55±18	21±17
	AttendLight	48±15	16±14	41±6.4	8.9±5.5
	UniTSA	39±2.8	**7.7±1.8**‡	**40±3.8**‡	**8.0±3.0**‡
VLM	VLMLight (32B)	**38±3.1**⋆	**7.1±2.3**⋆	**40±5.5**†	**7.5±4.0**†
	VLMLight (3B)	53±12	17±7.5	56±13	17±8.8
Ours	**VLALight** (0.5B)	45±1.6	14±1.3	**38±4.0**⋆	**6.5±3.0**⋆

Beijing Pinganli

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	48±1.9	15±1.3	47±8.6	12±8.2
	Webster	48±1.8	15±1.2	47±8.2	12±8.0
	SOTL	48±0.8	14±0.7	47±10	12±9.2
	MaxPressure	**47±1.3**†	**13±1.0**‡	**43±8.3**‡	**8.8±7.4**‡
RL	IntelliLight	55±7.5	18±6.1	43±8.2	**8.3±7.2**†
	PressLight	64±6.6	26±4.6	59±20	22±16
	AttendLight	60±5.1	24±3.9	55±17	19±14
	UniTSA	**48±3.3**‡	**12±2.5**†	44±9.7	9.0±7.6
VLM	VLMLight (32B)	**46±2.6**⋆	**12±2.3**⋆	**43±6.5**†	9.0±5.5
	VLMLight (3B)	57±11	20±8.1	51±15	16±12
Ours	**VLALight** (0.5B)	56±3.2	23±3.2	**37±4.3**⋆	**3.1±2.8**⋆

Beijing Changjianglu

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	43±1.1	8.0±0.6	**37±4.2**†	**2.0±2.8**⋆
	Webster	43±1.1	7.9±0.7	**37±3.9**⋆	**2.0±3.0**†
	SOTL	48±2.2	13±2.2	43±6.9	7.6±6.0
	MaxPressure	**43±1.6**‡	7.7±1.2	40±6.7	5.4±5.5
RL	IntelliLight	**42±2.0**†	**5.6±1.2**⋆	38±2.9	**2.5±2.4**‡
	PressLight	44±3.0	7.6±2.2	39±5.6	3.3±4.2
	AttendLight	44±3.2	7.5±2.2	38±4.0	3.6±3.8
	UniTSA	43±2.2	**6.3±1.5**‡	**38±5.2**‡	2.7±3.5
VLM	VLMLight (32B)	**42±2.0**⋆	**6.3±1.5**†	38±4.3	3.0±3.0
	VLMLight (3B)	54±9.9	14±6.4	44±10	7.0±6.5
Ours	**VLALight** (0.5B)	47±1.1	12±1.0	38±4.4	3.5±3.8

Tianjin Zhijingdao

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	51±5.7	18±4.3	55±18	21±15
	Webster	51±5.7	18±4.3	55±18	21±15
	SOTL	55±2.8	23±2.2	67±20	32±20
	MaxPressure	**48±5.2**⋆	**16±4.0**†	52±20	19±18
RL	IntelliLight	57±11	21±8.7	62±24	27±20
	PressLight	92±27	51±24	91±60	51±49
	AttendLight	81±30	45±30	78±49	43±48
	UniTSA	**51±7.2**‡	**16±6.0**‡	**50±14**‡	**16±12**‡
VLM	VLMLight (32B)	**48±7.4**†	**15±6.0**⋆	**48±18**†	**14±14**†
	VLMLight (3B)	74±23	33±17	85±54	41±39
Ours	**VLALight** (0.5B)	55±4.3	23±3.6	**45±13**⋆	**12±11**⋆

Table 8: Per-city comparison on the regular flow (mean±std over 4 emergency-injection sets, seconds). The top three per column are *boldfaced* and ranked ⋆/†/‡. BH: Beijing Beihuan; BSH: Beijing Beishahe; CJL: Beijing Changjianglu; GJY: Beijing Gaojiaoyuan; PAL: Beijing Pinganli; ZJD: Tianjin Zhijingdao.

Type	Method	BH		BSH		CJL		GJY		PAL		ZJD	
		ETT↓	EWT↓	ETT↓	EWT↓	ETT↓	EWT↓	ETT↓	EWT↓	ETT↓	EWT↓	ETT↓	EWT↓
Rule	FixTime	**59±19**‡	**19±12**†	44±2.9	12±2.6	**37±4.1**†	**1.6±2.8**†	57±6.7	22±7.2	51±7.6	16±7.8	63±24	28±20
	Webster	62±17	21±12	43±2.6	11±2.6	**37±4.0**⋆	**1.4±2.6**⋆	57±7.0	22±7.1	51±7.8	16±7.8	63±24	28±20
	SOTL	**59±9.5**†	**21±9.5**‡	44±3.8	11±3.5	41±4.3	5.0±4.4	54±4.6	19±3.3	46±15	11±12	61±15	26±14
	MaxPressure	77±21	35±17	42±5.9	9.3±5.1	45±10	9.4±8.6	59±10	25±10	45±8.3	9.7±8.0	67±20	33±17
RL	IntelliLight	124±25	69±24	41±5.1	8.8±3.9	40±3.2	3.6±2.5	**52±6.9**‡	**16±5.9**‡	**43±5.3**‡	**8.0±4.3**‡	65±18	29±16
	PressLight	112±21	52±11	54±5.6	19±4.2	41±4.6	4.6±4.1	91±13	45±7.8	69±9.1	29±7.8	89±26	47±20
	AttendLight	72±15	30±13	**40±6.3**†	**6.9±5.4**†	41±2.2	5.0±2.7	83±10	42±7.8	58±16	22±13	80±24	42±21
	UniTSA	62±7.0	22±5.6	**41±4.0**‡	**7.8±3.7**‡	40±8.2	4.8±6.4	53±10	18±8.7	**41±8.5**†	**6.1±6.8**†	**50±8.0**†	**16±5.5**†
VLM	VLMLight (32B)	65±13	25±9.9	42±2.9	8.7±3.4	40±3.9	3.5±2.6	**48±4.0**†	**14±3.9**†	44±9.8	9.2±8.7	**49±6.2**⋆	**14±3.9**⋆
	VLMLight (3B)	154±12	70±14	65±11	24±7.8	53±13	12±7.6	113±10	62±6.5	61±22	22±18	112±70	62±55
Ours	**VLALight** (0.5B)	**57±13**⋆	**17±9.8**⋆	**38±1.7**⋆	**6.6±2.2**⋆	**38±1.4**‡	**2.4±1.7**‡	**44±4.2**⋆	**9.6±3.3**⋆	**36±4.2**⋆	**3.0±2.7**⋆	**55±16**‡	**20±13**‡

reduces EWT by 17.0% and 65.6% over VLMLight (32B) and by 67.2% and 80.5% over VLMLight (3B). Tianjin Zhijingdao shows a similar pattern, with EWT falling by 18.6% relative to the 32B model and by 71.6% relative to the 3B model. The T-junctions Beihuan and Beishahe record smaller but still clear EWT improvements (11.7% and 13.6% over 32B; 70.1% and 62.9% over 3B). The only exception is Changjianglu, a low-volume T-junction where emergency vehicles rarely queue:

Table 9: Per-flow breakdown of all four metrics. Values are grouped by method family; the top-3 values per block are boldfaced with ⋆, †, ‡.

Regular flow

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	48±5.8	14±4.4	52±15	16±13
	Webster	48±5.9	14±4.4	52±15	17±13
	SOTL	48±3.2	14±3.2	51±12	15±11
	MaxPressure	**47±4.7**‡	**13±3.2**‡	56±18	20±15
RL	IntelliLight	51±9.4	15±6.3	61±32	23±25
	PressLight	68±18	28±14	76±28	33±20
	AttendLight	54±8.7	18±8.0	62±22	25±19
	UniTSA	**46±3.7**†	**11±3.2**⋆	**48±11**†	**12±8.7**‡
VLM	VLMLight (32B)	**45±3.6**⋆	**11±3.1**†	**48±11**‡	**12±8.6**†
	VLMLight (3B)	77±15	32±11	93±46	42±32
Ours	**VLALight** (0.5B)	53±5.6	19±5.4	**45±12**⋆	**9.8±9.3**⋆

Rush-hour flow

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	46±3.6	13±4.1	49±11	14±9.8
	Webster	**46±3.5**‡	13±4.0	49±12	15±11
	SOTL	48±3.8	15±4.2	55±14	20±13
	MaxPressure	**45±2.8**†	**12±3.2**†	49±9.9	14±9.0
RL	IntelliLight	52±7.5	16±6.7	49±14	13±12
	PressLight	69±19	31±19	71±33	32±29
	AttendLight	67±23	31±25	64±36	28±35
	UniTSA	47±3.9	**12±4.2**‡	**46±11**‡	**11±9.9**‡
VLM	VLMLight (32B)	**45±3.0**⋆	**11±3.5**⋆	**44±10**†	**9.9±8.6**†
	VLMLight (3B)	59±4.2	20±4.5	60±18	21±14
Ours	**VLALight** (0.5B)	50±5.8	18±6.5	**42±8.7**⋆	**8.3±7.2**⋆

Off-peak flow

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	42±3.0	9.7±3.3	45±10	12±9.4
	Webster	42±3.0	9.6±3.2	45±9.4	12±9.2
	SOTL	49±4.3	17±4.9	53±18	19±17
	MaxPressure	41±3.2	8.6±3.0	**41±8.7**‡	**7.3±7.7**‡
RL	IntelliLight	41±4.4	7.6±3.4	43±11	9.2±9.3
	PressLight	49±8.1	15±7.4	47±14	13±13
	AttendLight	48±9.3	14±9.0	47±15	13±14
	UniTSA	**40±2.2**†	**6.4±1.9**†	42±8.8	8.2±7.4
VLM	VLMLight (32B)	**39±2.8**⋆	**6.0±2.2**⋆	**39±6.6**⋆	**6.1±4.6**⋆
	VLMLight (3B)	**41±4.1**‡	**7.3±3.4**‡	44±9.5	9.2±7.1
Ours	**VLALight** (0.5B)	48±3.7	15±4.0	**40±5.6**†	**6.3±4.0**†

Fluctuating flow

Type	Method	ATT↓	AWT↓	ETT↓	EWT↓
Rule	FixTime	50±5.3	15±5.1	54±17	19±15
	Webster	49±5.3	15±5.1	54±17	18±15
	SOTL	50±5.7	16±5.4	55±16	20±14
	MaxPressure	**48±5.3**†	**13±5.1**‡	53±17	17±14
RL	IntelliLight	55±13	18±10	59±24	21±19
	PressLight	74±30	34±27	87±58	44±49
	AttendLight	65±30	29±31	68±45	31±44
	UniTSA	**49±7.4**‡	**13±6.9**†	**50±15**†	**15±13**†
VLM	VLMLight (32B)	**47±7.1**⋆	**13±6.6**⋆	**51±16**‡	**15±13**‡
	VLMLight (3B)	74±18	30±13	87±42	39±30
Ours	**VLALight** (0.5B)	52±5.8	19±5.9	**45±9.9**⋆	**10±8.7**⋆

VLALight is on par with VLMLight (32B) on ETT and slightly worse on EWT, because the cascaded 32B pipeline has enough time to reason when demand is light. The fact that a single 0.5B model preserves its emergency advantage across three T-junctions, three crossroads, and 6 to 14 incoming lanes demonstrates strong cross-topology generalization.

The breakdown by method family further clarifies the trade-off. Rule-based methods such as MaxPressure, FixTime, and Webster remain competitive on routine ATT/AWT—MaxPressure even achieves the lowest overall ATT—but they treat every vehicle equally and therefore produce much higher emergency waiting times. Among RL methods, IntelliLight and UniTSA are robust on routine metrics, whereas PressLight and AttendLight deteriorate markedly under event scenarios, with ATT and ETT rising by 20% to 50% on several intersections. VLMLight (32B) achieves the best routine metrics overall, thanks to its large backbone and explicit reasoning, but its multi-agent cascade is slow and brittle. VLMLight (3B) is faster yet unstable, often matching or underperforming the rule baselines. VLALight occupies a distinct niche: it is the only model that simultaneously keeps emergency metrics at or near the best while running in a single forward pass.

C.2 Comparison Across Traffic Flows

Averaged across the four evaluation flows, the same routine-for-emergency trade-off holds. On the regular training flow, rush-hour, and fluctuating flow, VLALight reduces ETT by between 4.5% and 12.6% and EWT by between 16.5% and 33.1% over VLMLight (32B). Against VLMLight (3B), the emergency improvements are larger (ETT reductions from 8.5% to 51.9%, EWT reductions from 31.2% to 76.6%) and extend to every flow. As with the per-city comparison, the compact model pays a routine-efficiency penalty: its ATT is from 11.2% to 23.6% higher than VLMLight (32B) and its AWT is from 46.8% to 158.7% higher.

Flow characteristics explain most of the variation. Under the rush-hour pattern, the heaviest and most sustained demand, VLALight's single-pass reaction matters most: it cuts ETT by 4.5% and EWT by 16.5% versus the 32B cloud model, and by 29.9% and 61.3% versus the 3B local model. The fluctuating pattern, which mimics directional imbalances, produces the largest ETT reduction over VLMLight (32B) at 12.6%, because rapid changes in dominant movements penalize the cascaded pipeline's decision latency. The regular flow also favors VLALight, where ETT drops by 7.0% and EWT by 21.1% relative to the 32B model. The only exception is the low-density off-peak flow: here VLMLight (32B) retains a slight edge (1.5% lower ETT and 2.6% lower EWT), because demand is low enough that cloud round-trips and multi-step reasoning do not miss the decision window. Even

in this regime VLALight still outperforms VLMLight (3B) on emergency metrics, with ETT 8.5% lower and EWT 31.2% lower. As Figure 5 in Section 4.5 shows, these flow-level results support the central claim of the paper: a lightweight end-to-end VLA is most beneficial when traffic is heavy or rapidly changing, exactly the conditions under which emergency vehicles are most likely to be delayed. The routine penalty is concentrated in low-demand settings where overall congestion is already modest, so the social cost of the trade-off is limited.

C.3 Inference Latency Comparison

Table 10 summarizes per-decision latencies. VLALight completes a pure forward pass in 0.224 s (mean over 36 closed-loop cases; median 0.182 s), whereas the VLMLight (32B, cloud) end-to-end estimate averages 4.684 s over 211 cases—this figure also covers SUMO stepping, serial agent calls, and network round-trips, and is therefore an upper-bound estimate of its per-step cost—and the locally deployed VLMLight (3B) agent totals 3.696 s over 77 decisions at Beijing Beihuan under the off-peak flow, without network round-trips. In other words, VLALight is more than an order of magnitude faster than either VLMLight deployment in normal decisions. For emergency decisions, where VLMLight's slow branch must fire, the latency advantage becomes even larger: the 32B cloud branch takes about 10.9 s and the 3B local branch takes about 4.2 s on average (peaking at 6.9 s), while VLALight still finishes in well under one second. Table 11 explains why the local 3B deployment remains slow: the emergency-decision LLM agent alone accounts for 2.8 s on average, and the scene-understanding VLM adds another 0.4 s before any control choice is made.

These latency differences are operationally significant. The controller issues a decision every 5 s, so an emergency branch that lasts between 4 s and 11 s cannot react to fast-approaching special vehicles within a single decision interval; it must either delay the response or pre-empt the phase earlier than necessary, both of which hurt throughput. VLALight leaves more than 4 s of headroom within the 5 s control cycle, making it suitable for edge deployment without cloud dependence. The advantage also explains why the per-flow results improve most under fluctuating and rush-hour demand: when queues evolve quickly, a low-latency model can adapt before congestion propagates.

Table 10: Per-decision inference latency in seconds. †End-to-end estimate including SUMO stepping and network round-trips; the other rows measure model/agent inference only.

Scheme	Mean	Median	Range
VLALight (0.5B, local)	0.224	0.182	0.143–1.675
VLMLight (32B, cloud)†	4.684	4.568	2.624–13.024
VLMLight (3B, local)	3.696	3.958	1.076–6.915

Table 11: Per-agent latency (seconds) of the locally deployed VLMLight (3B) over 77 decisions at Beijing Beihuan under the off-peak flow.

Agent	Calls	Mean	Med.	Range
Scene understand. (VLM)	231	0.411	0.399	0.303–5.458
Scene analysis (LLM)	77	0.081	0.074	0.061–0.590
Mode selection (LLM)	77	0.079	0.079	0.068–0.092
Emergency decision (LLM)	64	2.770	2.765	1.789–4.195

C.4 Visual-Input and Dwell Ablations

Minimum-green dwell constraint. We ablate the minimum-green dwell constraint (Figure 7) by letting the policy switch phases at every 5 s decision step (w/o MinGreen). Removing the constraint degrades every metric, confirming that a minimum phase duration is essential for continuous, non-myopic closed-loop control.

The minimum-green rule forces the policy to hold a selected phase for at least 15 s (three decision intervals) before switching. Without it, the model becomes myopic: it can chase short-term queue reductions by flipping phases every step, but each switch triggers a 3 s yellow/all-red clearance interval during which no vehicle clears the intersection. The cumulative lost time outweighs the apparent gain, and the resulting phase flicker also prevents vehicles from building enough headway

to pass efficiently. The ablation therefore raises both routine and emergency waiting times, because emergency vehicles are more likely to arrive during a clearance interval or be trapped behind a just-started queue. The minimum-green dwell acts as a simple but effective inductive bias: it stabilizes the policy, discourages trivial oscillations, and aligns the learned behavior with real-world signal-timing practice.

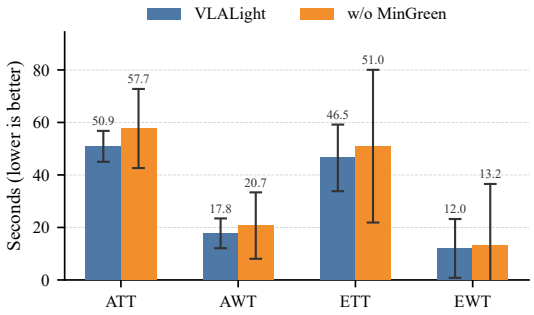


Figure 7: Ablation: no minimum-green dwell constraint.

D Extended Case Studies

We use representative snapshots to illustrate how VLALight translates the visual observation into a phase decision. Figure 8 shows an emergency-priority example at a four-phase crossroad: at time T an emergency vehicle approaches the intersection in one camera view, VLALight switches to the phase serving its movement at the next decision step, and at time $T{+}1$ the vehicle has crossed the intersection. The three case studies below further compare the input composite image at two consecutive decision steps; the green overlay marks the phase selected at step $T + 1$. Orange boxes highlight special vehicles for visualization only and are not part of the VLA input.

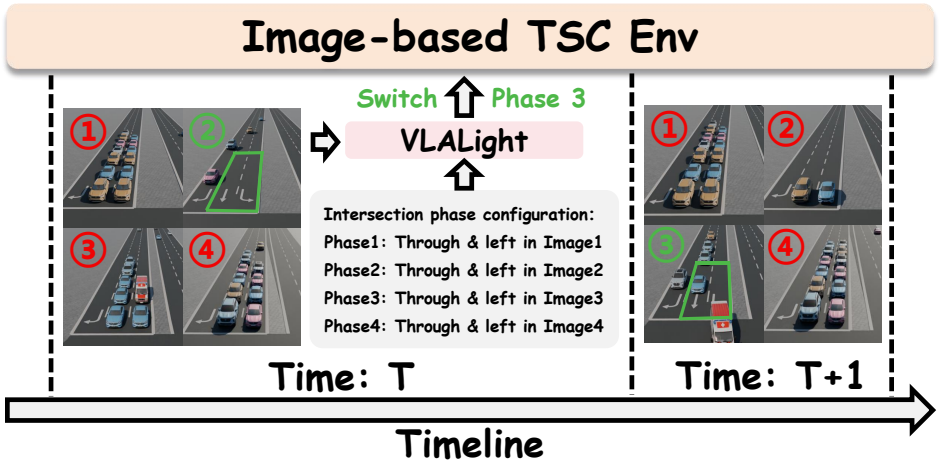


Figure 8: An emergency-priority example in TransimHub. At time T an emergency vehicle approaches the intersection in one camera view; VLALight switches to the phase 3, and at time $T{+}1$ the vehicle has crossed the intersection.

D.1 Routine Control: Serving the Heaviest Queue

Figure 9 shows a routine decision under regular traffic. At step T the bottom-right approach carries the longest queue, while the other approaches are comparatively light. VLALight switches to the phase serving that approach; at step $T + 1$ the bottom-right view receives the green overlay and the

queued vehicles are discharged. This example confirms that, in the absence of special vehicles, the model allocates green time to the approach with the highest immediate demand.

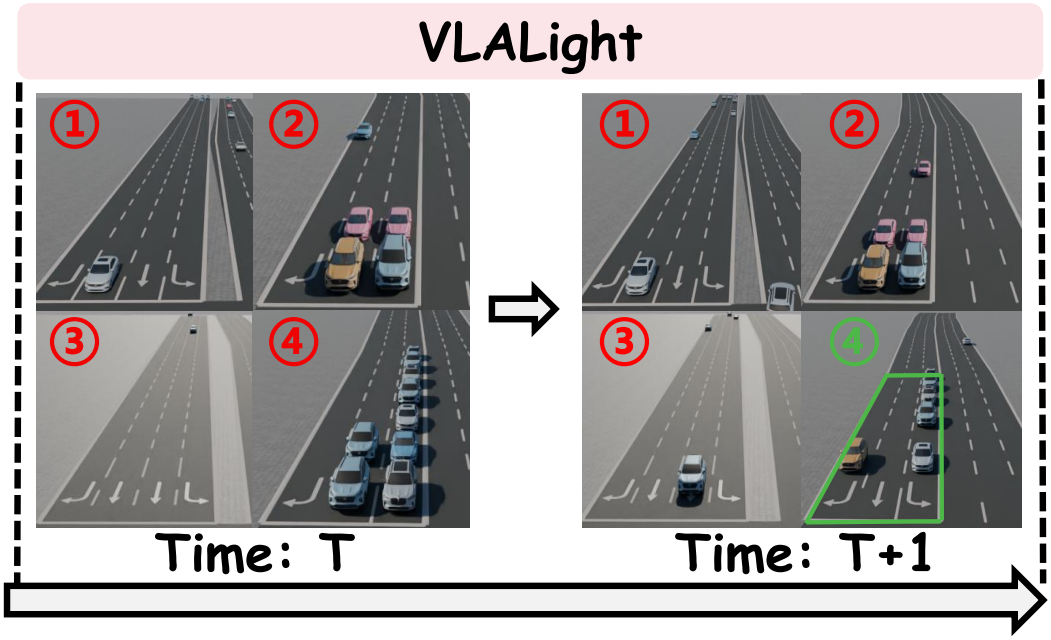


Figure 9: D1: routine control. VLALight serves the longest queue. Orange boxes, if present, mark special vehicles for visualization only.

D.2 Emergency Priority: Special-Vehicle Preemption

Figure 10 illustrates emergency-priority behavior at a three-phase T-junction. A special vehicle is approaching from the top-left approach (orange box). Although the top-right approach also carries traffic, VLALight switches to the phase serving the top-left approach so the special vehicle can pass without stopping; at step $T + 1$ the top-left view is highlighted green. The decision shows that the model has learned to detect emergency vehicles visually and give them priority regardless of the intersection geometry.

D.3 Failure Case: Two Simultaneous Special Vehicles

Figure 11 shows a harder situation in which special vehicles appear on two approaches at step T (orange boxes). The vehicle on the bottom-right approach has been waiting in a longer queue, so serving that approach would clear more vehicles and reduce overall waiting time. VLALight instead switches to the phase for the top-left approach at step $T + 1$. This mistake is likely due to two limitations. First, the policy decides from a single observation and lacks long-range temporal context about how long each queue has been waiting. Second, the visual features of the police-car model are not sufficiently distinct from regular vehicles, so the model may fail to recognize that the bottom-right vehicle is also a high-priority emergency vehicle.

D.4 Structured Failure Observations

Beyond the single snapshot above, we qualitatively categorize the failure modes observed in the evaluation logs along the five dimensions that a deployment analysis would need to track. (i) *Conflicting simultaneous emergencies*: when two special vehicles approach on conflicting movements, the policy commits to one without weighing queue seniority, as in Figure 11; this is the most frequent failure pattern we observe. (ii) *Vehicle-appearance ambiguity*: police cars are visually closer to ordinary sedans than ambulances or fire engines, and missed or late priority correlates with police-car events. (iii) *Single-frame temporal blindness*: because the policy sees one frame per

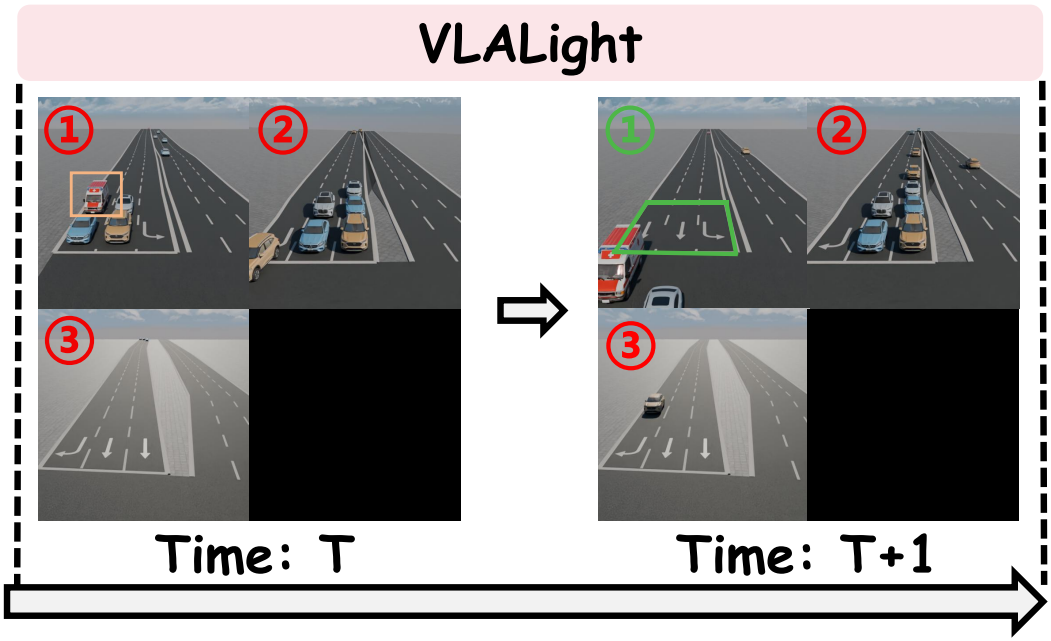


Figure 10: D2: emergency priority at a T-junction. VLALight grants green to the phase serving the special vehicle. Orange boxes are for visualization only.

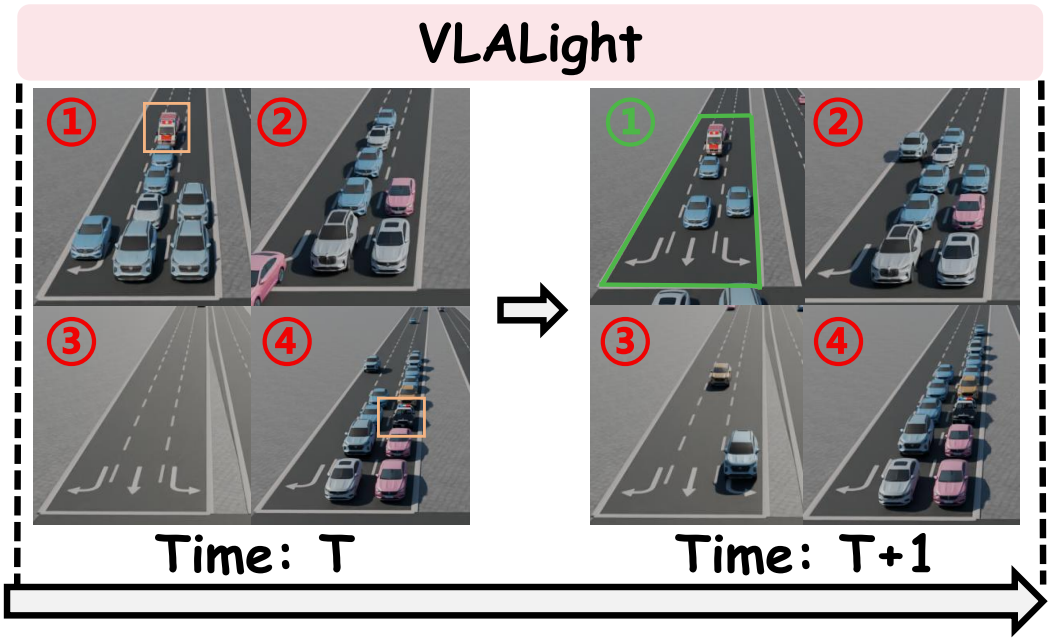


Figure 11: D3: failure case. VLALight prioritizes the top-left special vehicle while the longer bottom-right queue remains unserved. Orange boxes are for visualization only.

decision, it cannot tell how long a vehicle has already waited, which biases it toward whichever emergency is most salient in the current view. (iv) *Dwell-constraint blocking*: an emergency vehicle that arrives immediately after a phase switch must wait out the remaining minimum-green interval; such delays are the price of the stabilization analyzed in Appendix C.4. (v) *Distance and occlusion*: vehicles far from the stop line occupy few pixels in the roadside views, and priority is occasionally

granted only once the vehicle is close enough to be visually distinct. We report these as qualitative observations rather than measured statistics; a quantitative failure census is left to future work.

E VLALight in the Broader VLA Landscape

VLA research has so far been dominated by robot manipulation, where policies consume one or a few ego-centric views of a workspace and emit continuous motor commands (Brohan et al., 2023; Zitkovich et al., 2023; Octo Model Team et al., 2024; Kim et al., 2024). Recent work has begun to carry the paradigm to other embodied domains, notably autonomous driving, where surround vehicle-mounted views are mapped to planned trajectories, and UAV navigation, where first-person aerial views are mapped to waypoint or velocity commands. Intersection traffic signal control, the target of VLALight, differs from these domains along four aspects: visual input, decision context, action output, and decision budget.

Three differences are structural. First, the observations are multiple roadside directional views rather than ego-centric views of one workspace, which motivates the stitched multi-directional visual representation. Second, decisions hinge on cross-view spatial relations, since granting green time to one direction withholds it from the others, rather than on the temporal interaction dynamics that dominate manipulation. Third, the output is a discrete phase choice rather than continuous motor commands, which motivates the phase-classification action space with learnable phase queries. A fourth difference is operational: TSC decisions are required every few seconds, a budget that is far more forgiving than driving or flight control yet still excludes multi-agent cascaded pipelines, which is why VLALight answers each decision in a single forward pass of a compact backbone.

F Limitations and Future Directions

We conclude by situating the limitations of VLALight and identifying promising directions for future work.

Design trade-offs. VLALight treats traffic signal control as a visual-language-action problem by casting the intersection state into a multi-directional image and predicting the next phase directly. This design removes hand-crafted state features, but it also discards explicit temporal memory and structured lane-level information that conventional RL methods often encode. The result is a compact policy that generalizes across visual layouts, yet it can lag when an episode requires reasoning about how queue lengths have evolved over many seconds.

Attribution of the emergency gains. Our comparison isolates the architectural difference from the emergency-aware VLMLight cascade, but it does not include rule- or RL-based controllers with emergency-aware objectives. We argue that such variants are artificial: these methods consume compact state statistics (counts, queues, pressure), and neither deployed detectors nor their simulator state interfaces carry emergency-vehicle identity, so an emergency-aware reward would have nothing to ground on. Their inability to perceive emergency vehicles is an inherent limitation of the paradigm, and our emergency claims are accordingly scoped to controllers that can perceive emergency vehicles.

Sim-to-real gap. All experiments were conducted in simulation, and transferring the model to a real intersection would introduce several practical challenges, including camera calibration errors, occlusions, object-detection failures, and communication latency between perception and the signal controller. Fine-tuning on a small real-world dataset or combining the visual backbone with robust tracking modules is a natural next step.

Routine versus emergency optimization. VLALight optimizes both routine and emergency metrics simultaneously, and prioritizing emergency vehicles necessarily withholds green time from routine traffic, so the emergency gains come with a measurable routine-efficiency cost (Section 4.2). VLALight is also not the best controller in every setting: under low-density off-peak demand and at low-volume intersections such as Beijing Changjianglu, the cascaded VLMLight (32B) retains an edge, because light demand leaves enough slack for its multi-step reasoning to stay

within the decision window, whereas VLALight's advantage comes from single-pass responsiveness precisely when demand is heavy or rapidly changing. Under extreme congestion, rare emergency delays can still occur, especially under injection set 4, which combines several special vehicles with a lane-blocking accident. Future work could add an explicit safety wrapper or a constrained optimization objective that reserves headroom for emergency vehicles under worst-case demand.

Temporal reasoning. Because the policy acts on a single observation, it cannot explicitly track how long individual vehicles have been waiting. Incorporating a lightweight recurrent module or a short history buffer into the visual encoder could improve performance during prolonged congestion without substantially increasing inference cost.

Broader future directions. Beyond single intersections, network-level coordination, multi-modal perception, and continual learning under changing traffic patterns remain open problems. We believe the VLA perspective provides a scalable starting point for these extensions.