Data Engineering and Applications

Research Paper

Diffusion Policy Learning for Cooperative Ramp Merge Control from Vehicle–Infrastructure Collaborative Perception Data

Nasser Al-Qasimi[1, *], Humaid Al-Mulla[1] and Ibrahim Al-Balushi[1]

[1] College of Engineering, Information Technology Division, University of Sharjah, Sharjah, PO Box 27272, United Arab Emirates
*Corresponding author: nasser.qas@sharjah.ac.ae

Abstract. Cooperative ramp merging is still difficult due to the delayed, partially occluded and heterogeneous observations of connected vehicles. Develop a safety-shielded diffusion policy in this paper that converts vehicle-infrastructure collaborative perception into coordinated longitudinal control. Roadside and onboard tracks are aligned in time and associated to form a confidence-aware interaction graph. A conditional denoising process then samples joint acceleration sequences, and risk guidance biases the reverse trajectory towards acceptable gaps, while a lightweight safety projection repairs residual constraint violations. Optimize auxiliary demonstrations for training strategies and test various traffic demand, penetration rates, sensor noise, occlusion, and communication delay scenarios in a closed-loop micro-simulator. Compared with a rule-based cooperative controller, model predictive control, behavior cloning and an unshielded diffusion policy, the proposed method has reduced the average ramp delay by 24.7%, decreased speed variance by 31.2%, and increased successful merges to 98.1%. The 5th-percentile time-to-collision is 1.54 seconds for the strongest non-diffusion baseline and 2.06 seconds; only 3.8% of the executed actions modify the safety projection. Under 300ms communication delay, the success rate is 94.6%, and confidence conditioning and receding-horizon resampling avoid a sharp drop. Therefore, the above results show that diffusion-based multimodal action generation can effectively and stably support merge coordination when combined with explicit uncertainty representation and a minimally intrusive safety layer.

Keywords: *Vehicle–Infrastructure Cooperation, Diffusion Policy, Ramp Merging, Risk Guidance*

Received on 07 May 2023, Accepted on 10 September 2023, Published on 18 September 2023

Introduction

Freeway on-ramp merging concentrates the conflict in a small Area and propagates the acceleration choice of one vehicle to the mainline. Traditional ramp control only regulates the supply of aggregate and does not solve the vehicle-level negotiation problem caused by a short gap and heterogeneous driving response [1]. Connected-vehicle message communication adds intent and kinematic data, but the on-board view is often obscured by trucks, barriers or ramp geometry [2]. Roadside sensing extends the observable area and offers vehicles preparing to merge a common spatial reference [3]. Together, these sources can be used to conduct vehicle-infrastructure collaborative perception to detect conflicts earlier and support coordinated motion rather than a series of independent yielding decisions [4]. However, the useful output is not an ideal scene; rather, it is a time-varying belief that has been constructed from measurements with different latencies, visibility and confidence levels.

A cooperative merging controller with rules, game models, optimization, and reinforcement learning has already been proposed. Rule-based methods are transparent, but they are cumbersome when there are multiple valid merge orders [5]. Model Predictive Control (MPC) can handle dynamics and explicit constraints, but the online search increases rapidly with the number of interacting vehicles and uncertain hypotheses [6]. The learned strategy reduces computation after training, but deterministic regression tends to average over different actions

https://doi.org/10.64972/dea.2023.v2i3.3748d:96-107

and may issue uncertain control commands near the merging sequence boundary [7]. Multi-agent reinforcement learning can find coordination, but reward design, distribution shift and safety certification for sparse hazardous events are still problems [8]. More significantly, many controllers collapse collaborative detections into a single deterministic state and thus fail to distinguish between a well-observed empty gap and a gap that appears empty only because of stale roadside packets.

This paper studies merge control as conditional generation over a short joint action horizon. A diffusion policy is selected to keep several feasible solutions and add guidance at inference time through iterative sampling. The four components of the proposed model are: confidence-aware fusion of onboard and roadside tracks, an interaction encoder that represents candidate conflicts, risk-guided denoising of joint accelerations, and a control-barrier projection applied only when a sampled action is close to a hard safety boundary. The optimization-assisted demonstration provides smooth and diverse supervision without the need for online optimization during deployment. Determine whether the combination of the two is more efficient without losing robustness, and if so, whether this advantage still exists in the case of sensor and communication degradation. Closed-loop testing is organized around traffic performance, safety tail, robustness surface, action distribution, and ablation, with all claimed mechanisms traceable to observable outcomes.

Related Work

Cooperative Perception and Merge Coordination

Collaborative Perception aligns observations from vehicle-mounted sensors and fixed infrastructure before object-level or feature-level fusion. Object-level fusion is attractive for control because tracks have physical meaning and require a relatively narrow bandwidth; however, it is dependent on association quality [9]. Feature-level collaboration has more abundant evidence, but it requires more communication and is harder to synchronize in time [10]. Infrastructure sensing has been used to expand the look-ahead region and distribute recommended gaps or target speeds for merge coordination [11]. These studies have shown the value of a shared scene, but confidence is often used only to filter detections. The controller received a normal status and did not explicitly report that a key neighbor is supported by only one of the two most recent sources, which is an old source. Track age, covariance and source diversity are thus control-relevant factors. Omission is particularly detrimental at a ramp; a small longitudinal error can cause the inferred order of the two vehicles to be reversed. A good collaborative representation should maintain physical consistency, handle unknown cross-correlation, and be small enough for a 10 Hz control loop.

Generative Policies and Safety-Constrained Learning

The optimization-based merging planner directly encodes collision, comfort, and throughput objectives, while the backward view execution limits the impact of prediction errors [12]. Imitation learning replaces repeated optimization with supervised reasoning, but unimodal action regression may cause demonstrations expressing different valid merge sequences to collapse [13]. Diffusion models offer an alternative by learning a conditional score of action sequences and have shown an ability to represent multimodal trajectories [14]. Safety layers based on barrier functions or constrained projection can then correct an unsafe nominal action while keeping most of the learned policy output [15]. Existing generative-control research has seldom integrated collaborative perception confidence, interaction-specific risk guidance and joint merge actions in a single deployable loop. Currently, the formula only addresses this connection and determines how often an explicit shield is required. The above separation is necessary for engineering validation: credit will be given to the generative policy for producing useful modes, guidance on ranking modes under soft risk, and a shield for enforcing hard local feasibility. If there are no component-level measurements, good safety outcomes may be due to excessive fallback rather than learning behavior.

https://doi.org/10.64972/dea.2023.v2i3.3748d:96-107

Collaborative Perception and Problem Formulation

Confidence-Aware Collaborative State

The controlled region has a ramp lane, two mainline lanes and a 240-m coordination zone upstream of the gore. Every connected car sends its pose, velocity and destination path at 10Hz. Two roadside units output object tracks at 20 Hz. All measurements are converted to curvilinear coordinates along the lane centerline and propagated to the current control time. For track i from source s, the propagated longitudinal position is given by the formula above.

$$p_i(t) = p_i(t_s) + v_i(t_s)\Delta t + \frac{1}{2}a_i(t_s)\Delta t^2 \quad \text{Eq.(1)}$$

Propagation separates transport delay from measurement time, and thus a delayed packet is not mistaken for a stationary object. The corresponding covariance increases with latency according to the above formula, and older packets are still valid but have a reduced impact.

$$\boldsymbol{\Sigma}_i(t) = \boldsymbol{F}(\Delta t)\boldsymbol{\Sigma}_i(t_s)\boldsymbol{F}^T(\Delta t) + \boldsymbol{Q}(\Delta t) \quad \text{Eq.(2)}$$

Track association uses a gated Mahalanobis distance in position-velocity space. Because the cross-correlations between roadside and onboard estimators are unknown, associated tracks are fused by covariance intersection as written below. A learned scalar gate adjusts the conservative mixing coefficient from visibility, packet age, and source count. The confidence retained for downstream control is then obtained from the fused covariance and packet age.

$$\boldsymbol{\Sigma}_f^{-1} = \omega\boldsymbol{\Sigma}_v^{-1} + (1-\omega)\boldsymbol{\Sigma}_r^{-1}, \boldsymbol{\mu}_f = \boldsymbol{\Sigma}_f[\omega\boldsymbol{\Sigma}_v^{-1}\boldsymbol{\mu}_v + (1-\omega)\boldsymbol{\Sigma}_r^{-1}\boldsymbol{\mu}_r] \quad \text{Eq.(3)}$$

The fused estimate still requires a separate reliability descriptor before it is passed to the interaction encoder.

$$c_i = exp\left[-\frac{tr(\boldsymbol{\Sigma}_i)}{\sigma_c^2}\right] exp\left(-\frac{\Delta t_i}{\tau_c}\right) \quad \text{Eq.(4)}$$

The processing chain is shown in Figure 1: time-aligned vehicle and roadside observations enter track association, conservative fusion, confidence calibration, interaction-graph construction, and the receding-horizon control interface. The two feedback paths in the diagram are: executed-action feedback to the ego estimator and communication-quality feedback to confidence calibration.

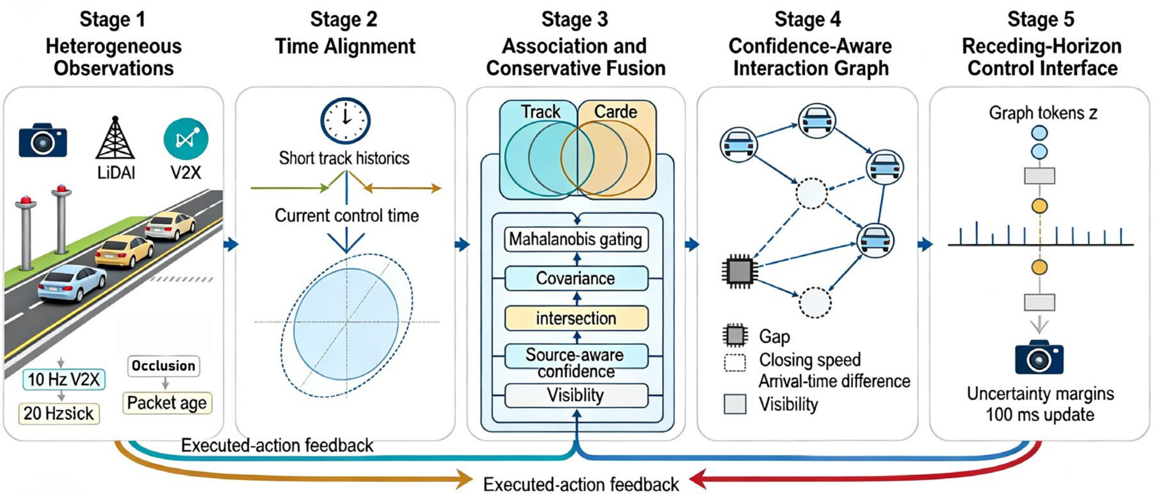

Figure 1. Vehicle-infrastructure collaborative perception and control-state construction pipeline.

Interaction Graph and Merge Objective

The graph is cropped by interaction relevance rather than Euclidean distance alone. A mainline vehicle far upstream can be omitted when its arrival interval cannot overlap the ramp vehicle, while a vehicle beyond the visible taper is retained if its closing speed makes it a plausible gap boundary. The missing neighbors are displayed as masks, and the order of the nodes remains unchanged. A random subset of low-confidence edges is removed and the remaining confidence values are perturbed during training. Augmentation can help the encoder differentiate between a sparse but plausible scene and a dense scene with weak associations. Lane topology enters via learned embeddings for upstream, acceleration, merge and downstream segments, and the same encoder weights are used to process vehicles before and after the lane boundary.

A Directed Graph is built at each control stage. Nodes are ramp vehicles and nearby mainline vehicles, and edges connect vehicles with overlapping predicted lane-aligned intervals within 5 seconds. Node features are position of merge, speed, acceleration, lane, vehicle length, connectivity and confidence. Edge features are relative distance, closing speed, estimated arrival-time difference and a visibility flag. The attention update is as follows: attenuate the uncertain sender features without eliminating them.

$$\boldsymbol{h}_i' = LN\left[\boldsymbol{h}_i + \sum_{j\in\mathcal{N}(i)} softmax_j\left(\frac{\boldsymbol{q}_i^T\boldsymbol{k}_j}{\sqrt{d}} + log\ c_j\right)\boldsymbol{W}_v\boldsymbol{h}_j\right] \quad \text{Eq.(5)}$$

The control output is a joint sequence of longitudinal accelerations for the ramp vehicle and up to four cooperative mainline vehicles over a 3s horizon. Non-cooperative vehicles are predicted by an ensemble motion model. The demonstration generator balances progress, speed harmony, comfort, gap acceptance and control effort based on the weighted objective below. Merge order is not explicitly marked; rather, it is derived from the optimised paths, and some valid orders in ambiguous scenarios remain ambiguous.

$$J = \lambda_p J_{progress} + \lambda_v J_{speed} + \lambda_g J_{gap} + \lambda_j J_{jerk} + \lambda_u J_{control} \quad \text{Eq.(6)}$$

Safety refers to the predicted longitudinal clearance under relative motion and state uncertainty. Therefore, the required gap is larger for a faster-closing and more uncertain approach, so a noisy closing vehicle will be given more leeway than a well-observed receding vehicle. The above state and purpose of uncertainty are actionable in terms of representation and constraints.

$$g_{ij} = (p_j - p_i - L_j) - \left[g_0 + T_h max(v_i - v_j, 0) + \kappa\sqrt{\sigma_i^2 + \sigma_j^2}\right] \quad \text{Eq.(7)}$$

Risk-Guided Diffusion Merge Policy

Conditional Denoising of Joint Actions

The four resolution levels and causal one-dimensional convolutions for each action channel in the Temporal Network. At the two coarsest levels, cross-attention is used and the receptive field covers the whole control horizon. The condition vector contains graph tokens, ramp vehicle identity, current actuator state, communication-quality statistics and the remaining distance to the gore. Sinusoidal Embeddings describe the diffusion step. Joint generation is superior to sequential vehicle decision-making because it can directly display coordinated behavior; for example, ramp acceleration can be combined with small-scale mainline release, rather than predicting separately and then coordinating. Padding masks ensure that scenes with fewer cooperative vehicles do not generate artificial zero-acceleration agents.

Demonstration action sequences are standardized and perturbed through a cosine variance schedule. At diffusion step k, the forward process gradually removes the structure of the joint action horizon by adding independent Gaussian noise.

$$\boldsymbol{x}_k = \sqrt{\bar{\alpha}_k}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_k}\boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \quad \text{Eq.(8)}$$

A temporal U-Net with graph-conditioned cross-attention predicts the injected noise. Learning the reverse process requires the network to recover the perturbation applied at each diffusion step.

$$\mathcal{L}_{diff} = \mathbb{E}_{\boldsymbol{x}_0,k,\boldsymbol{\epsilon}}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_k, k, \boldsymbol{z})||_2^2] \quad \text{Eq.(9)}$$

To prevent overdependence on perfect connectivity, source masks, packets, and confidence channels are randomly dropped during training. At deployment, the reverse process begins from Gaussian noise and runs for 12 deterministic sampling steps using the update below.

$$\boldsymbol{x}_{k-1} = \sqrt{\bar{\alpha}_{k-1}}\hat{\boldsymbol{x}}_0 + \sqrt{1-\bar{\alpha}_{k-1}}\hat{\boldsymbol{\epsilon}}_\theta \quad \text{Eq.(10)}$$

Then, according to the circumstances and risks, a nominal reverse update is carried out. Classifier-free guidance integrates conditional and unconditional noise estimates. A separate differentiable risk term calculates soft gap violations, acceleration discontinuity, and terminal merge infeasibility, and its gradient alters the predicted noise as shown below.

$$\tilde{\boldsymbol{\epsilon}}_\theta = \boldsymbol{\epsilon}_u + w_c(\boldsymbol{\epsilon}_c - \boldsymbol{\epsilon}_u) + w_r\sqrt{1-\bar{\alpha}_k}\nabla_{x_k}R(\boldsymbol{x}_k, \boldsymbol{z}) \quad \text{Eq.(11)}$$

This shifts the samples without retraining the denoiser. The policy generates eight candidate joint sequences in parallel. Candidates are ranked according to predicted safety margin, progress, comfort and disagreement with the fused belief. Maintain multimodality before the final selection step, avoiding averaging early merged sequences with priority output sequences.

Safety Projection and Receding-Horizon Execution

Barrier constraints are only generated for pairs that can interact in the next 1.5 seconds to keep the projection small and avoid conservative corrections from distant vehicles. Linearization is carried out at the first denoised action and the current fused state. For cooperative mainline vehicles, adjustments can be distributed between the two agents based on comfort weight; for non-cooperative neighboring vehicles, only the controlled vehicle will be changed. The ranking score also has a margin for model mismatch, and therefore the projection is not expected to correct systematically risky samples. Therefore, the frequency of intervention is an engineering diagnosis; a rising rate may indicate sensing degradation, out-of-distribution traffic patterns or poorly tuned guidance before a serious violation occurs.

Only the first action of the selected sequence is executed. Before actuation, a one-step quadratic program finds the smallest correction that satisfies control bounds and the pairwise barrier condition below.

$$\dot{h}_{ij}(\boldsymbol{x}, \boldsymbol{u}) + \gamma h_{ij}(\boldsymbol{x}) \geq 0 \quad \text{Eq.(12)}$$

These barrier inequalities are enforced through a minimum-correction projection of the nominal command.

$$\begin{aligned} \boldsymbol{u}^* = \underset{\boldsymbol{u}}{arg\,min} \quad & \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{u}_{nom}\|_{\boldsymbol{W}}^2 + \rho\|\boldsymbol{\xi}\|_1 \\ s.t. \quad & \boldsymbol{Au} \leq \boldsymbol{b} + \boldsymbol{\xi}, \qquad \boldsymbol{\xi} \geq \boldsymbol{0}. \end{aligned} \quad \text{Eq.(13)}$$

Slack is permitted only for numerical feasibility and carries a high penalty. The operational architecture is summarized in Figure 2. The upper branch encodes the collaborative graph; the center branch performs conditional denoising, risk guidance, and candidate ranking; the lower branch projects the selected first action and feeds the observed transition back to the next 100 ms cycle. Distinct icons for sensors, graph attention, a denoising trajectory stack, a shield, and cooperating vehicles should make the inference path visually separable from the safety path.

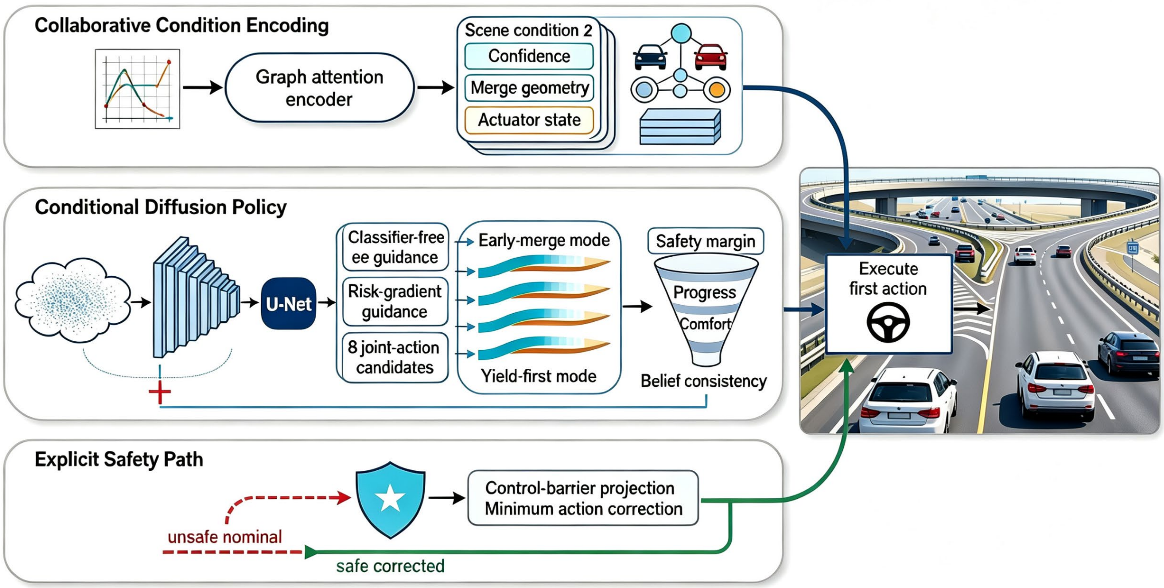

Figure 2. Risk-guided diffusion policy, candidate selection, safety projection, and receding-horizon execution.

The final executed action is rate-limited by the clipping operation below, suppressing command chatter when successive samples select neighboring modes.

$$\boldsymbol{u}_t = clip[\boldsymbol{u}_t^*, \boldsymbol{u}_{t-1} - j_{max}\Delta t, \boldsymbol{u}_{t-1} + j_{max}\Delta t] \quad \text{Eq.(14)}$$

If the collaborative data packet flow is lost, confidence will decrease, and the condition mask will shift the strategy to only allow boarding behavior. If no candidate meets the ranking requirements, a conservative yielding sequence for that projection layer will be returned.

Experimental Evaluation and Analysis

Experimental Setup and Baselines

The simulator will update vehicle motion at 50Hz and the control update rate is 10Hz. Human-driven vehicles use an intelligent-driver longitudinal model with heterogeneous time gaps and a lane-change model that reproduces forced and cooperative merges. Cooperative commands are subject to a 0.35 s first-order actuator lag and bounded between -4.0 and $2.5\ \mathrm{m/s^2}$. A merge is successful when the ramp vehicle crosses the lane boundary without collision, without an emergency stop below 2 m/s in the taper, and with a post-merge gap that stays positive for 2 s. Delay is measured against an unconstrained free-flow traversal, preventing slow initial arrivals from being counted as controller delay. Hard braking denotes acceleration below $-3.0\ \mathrm{m/s^2}$ for at least 0.2 s.

A closed-loop microscopic simulator models a 1.4 km freeway segment with a 320 m acceleration lane. Traffic demand ranges from 1200 to 2200 veh/h/lane, ramp inflow from 250 to 700 veh/h, and connected-vehicle penetration from 20% to 100%. Passenger-car desired speeds follow a truncated distribution between 22 and $33\ \mathrm{m/s}$, with vehicle lengths from 4.2 to 5.3 m. Roadside range noise is $0.25 - 1.50\ \mathrm{m}$, packet delay is $0 - 500\ \mathrm{ms}$, and random block occlusions last $0.5 - 3.0\ \mathrm{s}$. Scenario generation follows the principle that demand, sensing, and communication factors should vary jointly rather than in isolated nominal cases [16]. Each configuration uses 30 random seeds and at least 600 merge events.

Compared five controllers: a gap rule-based cooperative controller, constrained model predictive control, graph conditional behavior cloning, an unshielded conditional diffusion policy, and the proposed risk-guided shielded diffusion policy. The predictive controller has the same dynamics and nominal safety gap as the demonstration optimizer, thus avoiding model advantage. Behavior cloning shares the graph encoder and action horizon with the method proposed above. Evaluation reports include success, ramp delay, speed variance, jerk, minimum time-to-collision, hard-braking rate and 95th-percentile computation time. Tail safety indicators are still used because averages may conceal infrequent close encounters [17]. Bootstrap confidence intervals are computed over seeds, and paired tests use matched scenario instances [18].

The training used 420,000 action windows, of which 70% were generated through constraint optimization and 30% were collected through random expert rules to expand the maneuvering patterns. The train, validation and test maps use different demand seeds and occlusion positions. AdamW is run for 220 epochs with a peak learning rate of 2×10^{-4}, batch size 256, exponential moving average 0.995 and gradient clipping at 1.0. The diffusion horizon has 30 control points, with a candidate count of eight, and the deployed sampler uses 12 steps. Hyperparameters are chosen based on the validation success and within a fixed safety limit; the test data remains unchanged during selection [19].

Traffic Efficiency, Safety, and Control Behavior

Under nominal demand, the gap rule controller repeatedly waits for a single deterministic acceptance threshold and exhibits stop-and-go behavior when the observed gap fluctuates around that value. Model predictive control anticipates the gap evolution and is smoother, but its limited branching horizon often commits to yielding before a later cooperative opening becomes beneficial. Behavior cloning reacts quickly but sets an excessively low probability for acceleration in ambiguous scenes. The two diffusion controllers are in different states. Risk guidance changes their rank the most when the predicted arrival-time differences are within 0.4s, that is to say, when the merge order is genuinely uncertain. Outside of that area, the guided and unguided candidates are almost the same; therefore, the safety mechanism does not apply a large delay penalty.

As shown in Figure 3(a), the proposed method has achieved a relatively high merge success rate of 98.1%, which is higher than that of unshielded diffusion (96.2%), model predictive control (94.8%), Behavior Cloning (91.5%), and the gap rule (88.7%). Combine proximity and response-severity indicators in the safety report instead of using a single average [20]. As shown in Figure 3(b) of the efficiency-stability plane, the method has an average delay of 3.42s and a standard deviation of 0.83m/s for speed; the corresponding model-predictive values are 4.31s and 1.07m/s. The safety tails shown in Figure 3(c) also show a 2.06 s 5th-percentile time-to-collision and a 0.42% hard-braking rate, improving the strongest non-diffusion baseline values of 1.54 s and 0.91%. Confidence Intervals for Success and Delay Remain Separate.

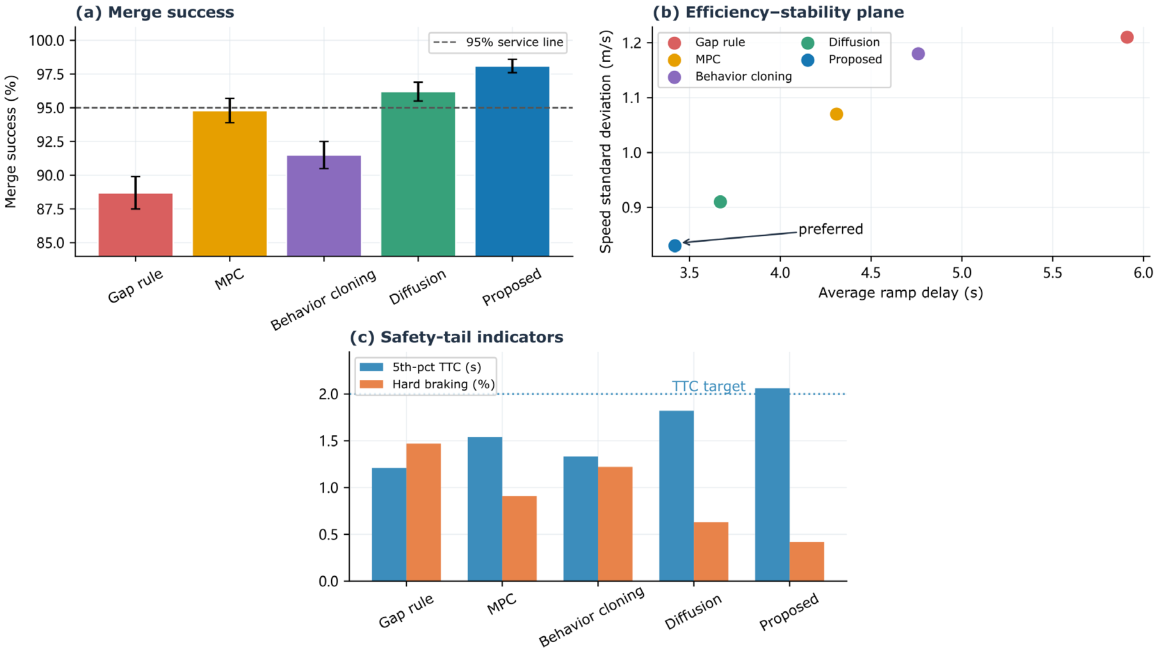

Figure 3. Overall closed-loop performance: (a) merge success with bootstrap intervals; (b) efficiency-stability Pareto scatter; (c) safety-tail indicators with threshold annotations.

Across the five demand levels, Figure 4(a) shows a controlled success decline from 99.0% at 1200 veh/h/lane to 94.2% at 2200 veh/h/lane, whereas model predictive control declines from 97.3% to 88.1%. Because congestion episodes produce asymmetric delay tails, distributional evidence is needed alongside mean performance [21]. The violin-box distributions in Figure 4(b) give a proposed median of 3.18 s, an interquartile range of 2.31 – 4.06 s, and a 95th percentile of 6.72 s, while the rule-based tail reaches 11.84 s.

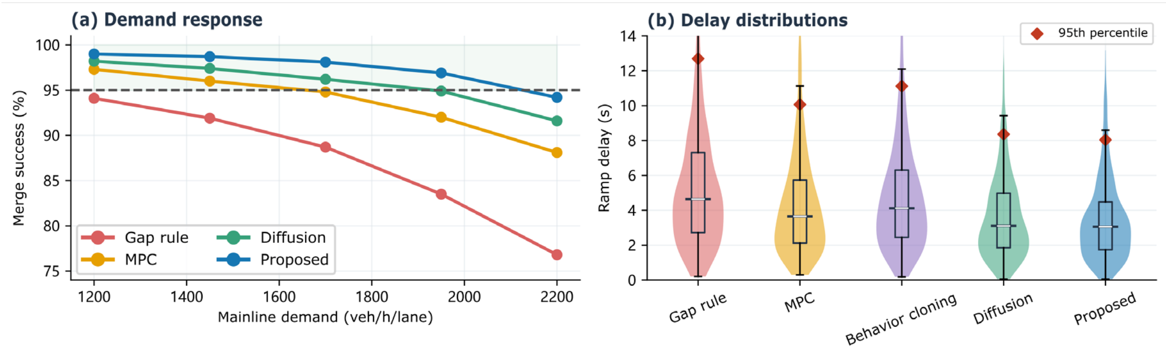

Figure 4. Demand response and delay distribution: (a) success curves across five demand levels with a 95% service line; (b) violin-box distributions with medians, quartiles, and 95th-percentile markers.

The gains do not arise from aggressive acceleration. Across all test runs, root-mean-square jerk is $1.18\ \mathrm{m/s^3}$ for the proposed controller, $1.12\ \mathrm{m/s^3}$ for model predictive control, $1.36\ \mathrm{m/s^3}$ for unshielded diffusion, and $1.74\ \mathrm{m/s^3}$ for behavior cloning. Mean ramp-vehicle energy proxy, calculated as positive tractive power per merge, is 0.6% higher than that of model predictive control but 8.4% lower than the gap rule. Therefore, this policy will be used to provide a small amount of smooth control effort to stabilize interactions without forcing the generation of brief gaps. Paired seed analysis reduces the delay by 24.7% and the speed-variance by 31.2% compared to the gap controller.

Stratified check can be free-running, synchronous or congested. In free flow, all learned controllers have exceeded 97% success and their average-delay difference is less than 0.6s. In a synchronous flow, the policy has the largest practical advantage; that is, small complementary actions can create a usable slot without requiring either stream to stop. Under congestion, the absolute delay of each controller increases, but the diversity of diffusion candidates reduces the number of aborted merge attempts. Therefore, the improvement will be in interactive modes rather than simple isolated modes. Seed-level confidence intervals are narrower after more merge operations per run due to a larger number of merge events, but scenario-level variability increases; both aggregation levels have been examined to avoid overstating the precision.

The early-merge space-time corridor in Figure 5(a) has a minimum predicted clearance of 2.34s and shows how the ramp vehicle passes between the mainline leader and follower. For the alternative yield-first response, Figure 5(b) shows the accepted-gap evolution and marks the yield command, gap-opening instant, and merge commitment; the minimum clearance is still 2.27 s before the accepted gap expands to 12.7 m. When multiple merged sequences achieve the same goal, multimodal output is appropriate [22], and these two representations confirm that the denoiser did not simplify the alternatives into an averaging operation. Before candidate ranking, the first-action density in Figure 5(c) contains peaks at -0.62 and $0.38\ \mathrm{m/s^2}$. The safety-intervention waterfall in Figure 5(d) attributes the increase in predicted minimum time-to-collision from 1.63 to 2.11 s primarily to risk guidance and records a peak safety-projection correction of only $0.21\ \mathrm{m/s^2}$.

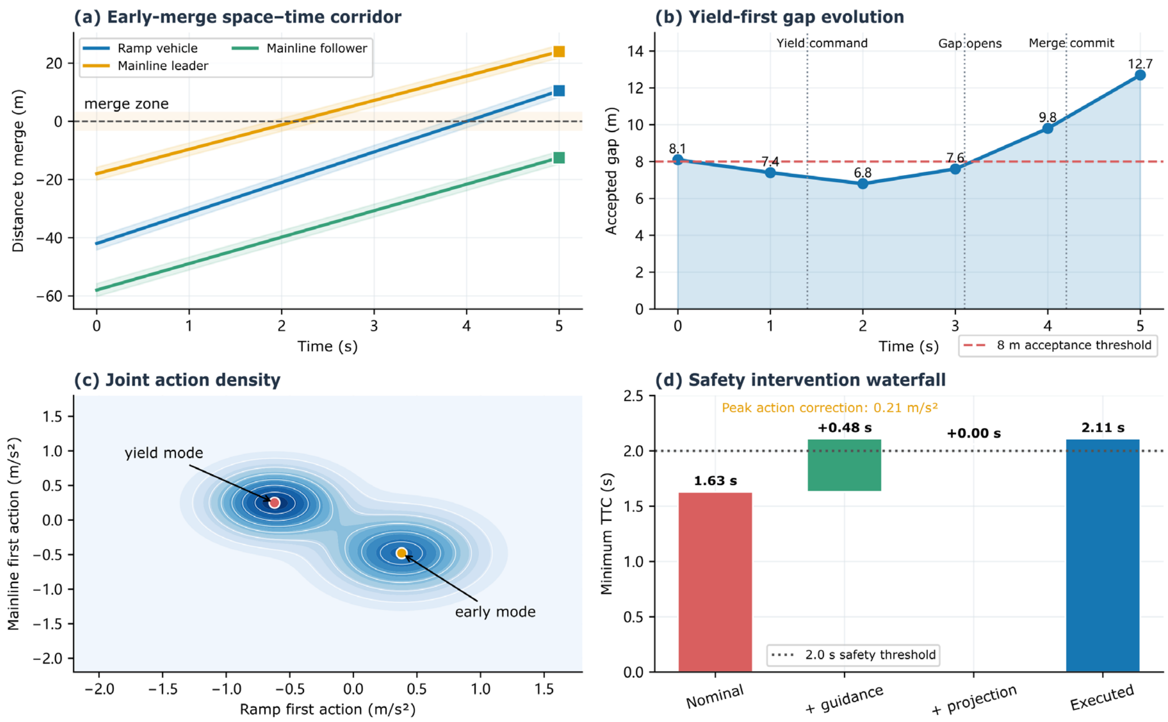

Figure 5. Policy behavior and multimodality: (a) early-merge space–time corridor; (b) yield-first accepted-gap evolution with maneuver events; (c) joint first-action density with modal annotations; (d) safety-intervention waterfall with threshold and correction annotations.

Robustness and Ablation Analysis

Sensitivity tests also vary the diffusion sampler. Reducing the reverse process from 12 to 6 steps cuts median denoising time by 43%, but merge success decreases by 1.4 points and jerk rises by $0.16\ \mathrm{m/s^3}$. Increasing the steps to 20 only improves the accuracy by 0.2 and increases the time by 13.8 ms; therefore, the chosen configuration is close to the knee of the accuracy-latency curve. Stochastic samplers produce slightly more diverse candidates but lack consistency in consecutive control cycles; deterministic sampling with a new initial noise provides a good balance. Warm-start the first half of the candidates from the previous shifted solution to

https://doi.org/10.64972/dea.2023.v2i3.3748d:96-107

reduce acceleration discontinuity without suppressing other merge orders, as the remaining candidates still start from independent noise.

The robust grid consists of independent Gaussian localization errors and burst communication losses generated by a bistable channel. Confidence calibration reduces the weight of propagated roadside tracks during a burst, and the graph mask preserves the last reliable topology for no more than 0.5 s. This way avoids the two opposite failures: treating a stale object as exact and deleting it immediately. At a 300 ms delay, the calibrated policy expands the predicted gap margin by 0.28 m and moves 12% of the selected candidates from the early-merge mode to the yield-first mode. It will change slowly and not suddenly. An uncalibrated variant maintains the nominal mode distribution until its success collapses at a large delay, indicating that confidence functions are operating as control inputs rather than simply as detectors of quality.

The success surface in Figure 6(a) remains above 94.6% through 1.0 m position noise and 300 ms delay, then falls to 89.7% at 1.5 m and 500 ms. Delay-aware testing reflects the bursty nature of cooperative communication more accurately than a fixed-noise assumption [23]. Over the same latency range, Figure 6(b) shows average delay rising from 3.42 to 4.18 s for the proposed method, compared with 4.76 to 6.41 s for behavior cloning. Shield activity in Figure 6(c) remains limited overall at 3.8%, increasing from 1.1% under clear perception to 8.9% in the most degraded setting.

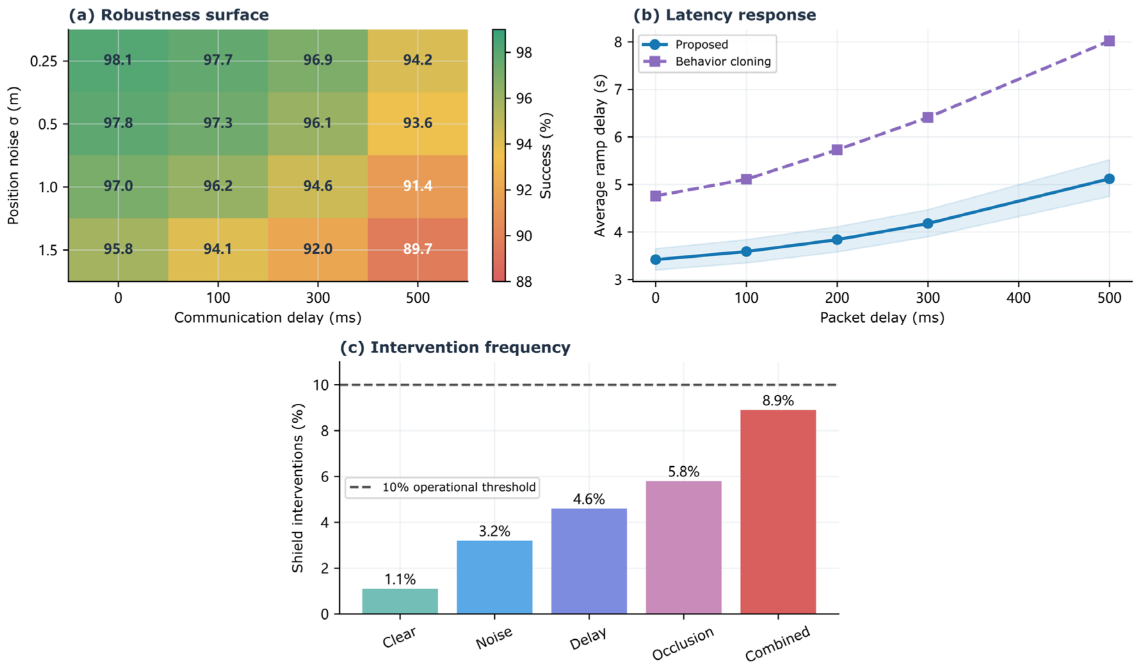

Figure 6. Robustness to perception and communication degradation: (a) success heat map over noise and delay; (b) delay curves under packet latency; (c) shield-intervention bars by degradation regime with a 10% operational threshold.

Success is $91.2\%, 94.0\%, 96.3\%, 97.4\%$, and 98.1% at $20\%, 40\%, 60\%, 80\%$, and 100% connected penetration, respectively. The benefit saturates beyond 80% because roadside tracks still cover unconnected vehicles. Therefore, the cooperative system assessment should also report on communication coverage and sensing coverage [24]. Removing roadside observations at a 40% penetration reduces the success rate to 87.6% and adds 1.31s of delay; thus, infrastructure sensing contributes beyond message coverage when a non-connected mainline vehicle occupies the target gap.

Occlusion duration produces a smaller penalty than equal-duration packet loss when the roadside unit remains available. With a 2 s onboard occlusion, fused-track continuity keeps success at 96.8%; simultaneous roadside occlusion reduces it to 92.7%. Correlated exchanges are rare, but more destructive than missed detections: even with an average delay change of less than 0.1 seconds, injecting a 1% probability of identity exchange can reduce tail collision time by 0.17 seconds. The confidence gate detects many swaps through kinematic inconsistency, yet it cannot resolve two adjacent vehicles with nearly identical velocities. This failure mode motivates retaining the explicit safety projection and indicates that future fusion should incorporate appearance or secure identity cues.

In Figure 7(a), removing confidence conditioning lowers success by 2.9 points, removing risk guidance lowers it by 1.8 points, removing the shield lowers it by 1.1 points, and replacing diffusion with deterministic regression lowers it by 5.4 points. Component-wise ablation is necessary here to distinguish representation, generation, guidance, and constraint effects [25]. The safety-efficiency frontier in Figure 7(b) gives guidance weight 1.2 at a 3.42s delay and a 2.06s tail time-to-collision; increasing the weight to 2.0 raises clearance to 2.24s but also increases delay to 4.05s. As shown in Figure 7(c), the proportions of graph encoding, denoising, ranking and projection are 4.8ms, 21.6ms, 3.7ms and 1.9ms respectively; the total reaches 34.7ms in the 95th percentile of the 100ms control budget.

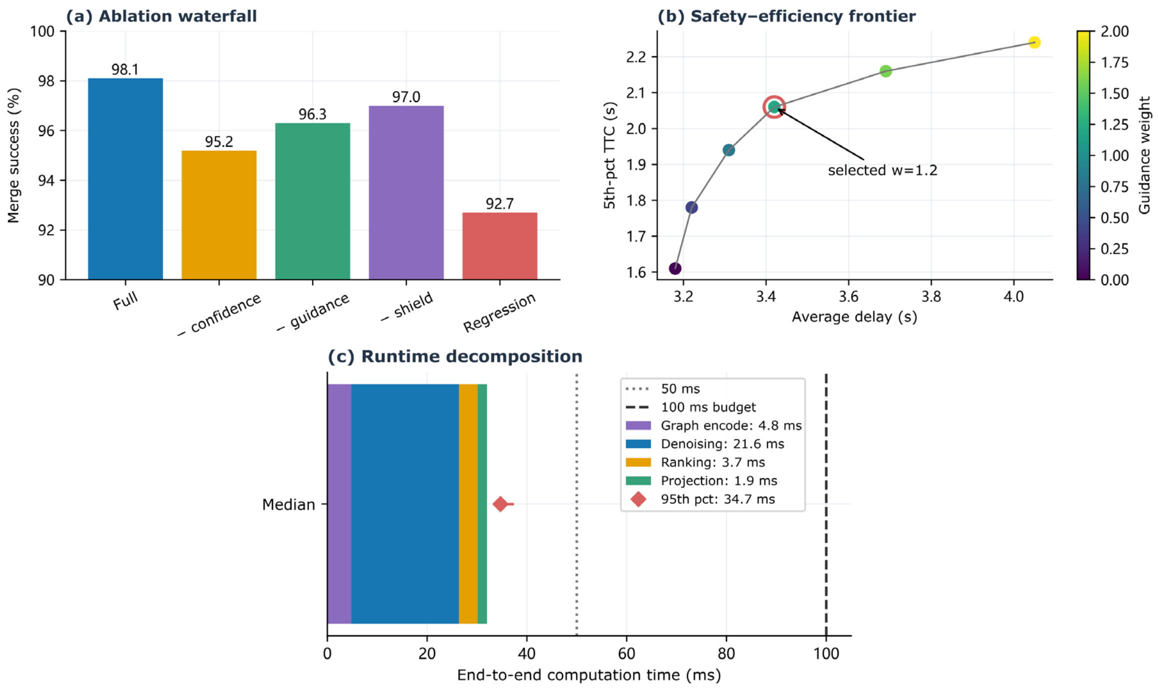

Figure 7. Mechanism and deployment analysis: (a) ablation waterfall for merge success; (b) guidance-weight safetyefficiency frontier; (c) computation-time decomposition with 50 and 100 ms budget markers.

The trained policy has 18.7 million parameters and is 76 MB in 32-bit format. INT8 post-training quantization reduces the storage to 21MB and lowers the median inference time from 29.4ms to 22.8ms; the success loss is 0.3 points. To avoid a too-small time window, runtime measurements include preprocessing and device transmission [26]. The candidate count will be reduced from 8 to 4, saving an additional 7.1ms; however, the tail time-to-collision will decrease by 0.09s because fewer maneuver modes reach the top ranking. The observed margins support 10Hz execution of the automotive edge GPU, and an explicit fallback is still required for hardware faults and out-of-distribution road geometry.

Conclusion

This paper presents a safety-shielded diffusion policy for freeway merge control with vehicle-infrastructure cooperative perception. The method keeps the source confidence in track fusion, encodes the interaction structure as a graph, generates multimodal joint action horizons via conditional denoising, and applies risk guidance before a minimally invasive barrier projection. Optimizing auxiliary demonstrations and recursive view sampling connects offline learning with control loops capable of responding to new observations. Therefore, the resulting formula considers uncertainty as part of the decision conditions and does not hide it in a nominal fused scene.

The present study is limited by simulation-based sensing and vehicle dynamics, a single merge geometry family, longitudinal control only, and an assumed trusted infrastructure clock. The demonstration generator may also underestimate unusual human negotiation behaviors, while the projection layer only guaranties the modeling of short-term constraints. Therefore, when there are positioning biases, actuator delays, adversarial packets, or culturally different concession behaviors, performance will vary.

In the future, hardware-in-the-loop testing will be combined with roadside field data, the action space will be expanded to coupled lane and longitudinal decisions, and online calibration for sensor and communication reliability will be added. Formal reachability analysis can be combined with sampling strategies to provide extended time horizon guaranties, while federated adaptation can update local behavior without centralizing the original trajectories. A third direction is cooperative explanation: transmit a concise intention and confidence summary for the generated merge order to help human-driven and autonomous vehicles anticipate the merge order before control actions are displayed. Cross-site evaluation should test whether the shared generative backbone can be adapted to different ramp lengths, traffic conventions and infrastructure layouts with limited local demonstrations. Add a secure message authentication and fault diagnosis function to enhance the robustness of the policy in the operation of a deployed cooperative transportation system.

Author Contributions

Nasser Al-Qasimi contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision, project administration, and funding acquisition. Humaid Al-Mulla contributes to analysis, investigation, data collection, draft preparation. Ibrahim Al-Balushi contributes to validation, analysis, investigation. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Yurtsever, E., Lambert, J., Carballo, A., & Takeda, K. (2020). A survey of autonomous driving: Common practices and emerging technologies. IEEE Access, 8, 58443–58469. https://doi.org/10.1109/ACCESS.2020.2983149

[2] Grigorescu, S., Trasnea, B., Cocias, T., & Macesanu, G. (2020). A survey of deep learning techniques for autonomous driving. Journal of Field Robotics, 37(3), 362–386. https://doi.org/10.1002/rob.21918

[3] Badue, C., Guidolini, R., Carneiro, R. V., Azevedo, P., Cardoso, V. B., Forechi, A., Jesus, L., Berriel, R., Paixão, T. M., Mutz, F., de Paula Veronese, L., Oliveira-Santos, T., & De Souza, A. F. (2021). Self-driving cars: A survey. Expert Systems with Applications, 165, Article 113816. https://doi.org/10.1016/j.eswa.2020.113816

[4] Muhammad, K., Ullah, A., Lloret, J., Del Ser, J., & De Albuquerque, V. H. C. (2020). Deep learning for safe autonomous driving: Current challenges and future directions. IEEE Transactions on Intelligent Transportation Systems, 22(7), 4316-4336. https://doi.org/10.1109/TITS.2020.3032227

[5] Kiran, B. R., Sobh, I., Talpaert, V., Mannion, P., Al Sallab, A. A., Yogamani, S. K., & Pérez, P. (2021). Deep reinforcement learning for autonomous driving: A survey. IEEE Transactions on Intelligent Transportation Systems, 23(6), 4909–4926. https://doi.org/10.1109/TITS.2021.3054625

[6] Aradi, S. (2020). Survey of deep reinforcement learning for motion planning of autonomous vehicles. IEEE Transactions on Intelligent Transportation Systems, 23(2), 740-759. https://doi.org/10.1109/TITS.2020.3024655

[7] Makantasis, K., Kontorinaki, M., & Nikolos, I. (2020). Deep reinforcement-learning-based driving policy for autonomous road vehicles. IET Intelligent Transport Systems, 14(1), 13–24. https://doi.org/10.1049/iet-its.2019.0249

[8] Pérez-Gil, Ó., Barea, R., López-Guillén, E., Bergasa, L. M., Gómez-Huélamo, C., Gutiérrez, R., & Díaz-Díaz, A. (2022). Deep reinforcement learning based control for autonomous vehicles in CARLA. Multimedia Tools and Applications, 81(3), 3553–3576. https://doi.org/10.1007/s11042-021-11437-3

[9] Wang, H., Yuan, S., Guo, M., Li, X., & Lan, W. (2021). A deep reinforcement learning-based approach for autonomous driving in highway on-ramp merge. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 235(10–11), 2726–2739. https://doi.org/10.1177/0954407021999480

[10] Wang, Y., Zheng, K., Tian, D., Duan, X., & Zhou, J. (2021). Pre-training with asynchronous supervised learning for reinforcement learning based autonomous driving. Frontiers of Information Technology & Electronic Engineering, 22(5), 673–686. https://doi.org/10.1631/FITEE.1900637

[11] Shan, M., Narula, K., Wong, Y. F., Worrall, S., Khan, M., Alexander, P., & Nebot, E. (2021). Demonstrations of cooperative perception: Safety and robustness in connected and automated vehicle operations. Sensors, 21(1), Article 200. https://doi.org/10.3390/s21010200

[12] Thandavarayan, G., Sepulcre, M., & Gozalvez, J. (2020). Cooperative perception for connected and automated vehicles: Evaluation and impact of congestion control. IEEE Access, 8, 197665–197683. https://doi.org/10.1109/ACCESS.2020.3035119

[13] Cui, G., Zhang, W., Xiao, Y., Yao, L., & Fang, Z. (2022). Cooperative perception technology of autonomous driving in the Internet of Vehicles environment: A review. Sensors, 22(15), Article 5535. https://doi.org/10.3390/s22155535

[14] Godoy, J., Jiménez, V., Artuñedo, A., & Villagra, J. (2021). A grid-based framework for collective perception in autonomous vehicles. Sensors, 21(3), Article 744. https://doi.org/10.3390/s21030744

[15] Li, T., Wu, J., Chan, C.-Y., Liu, M., Zhu, C., Lu, W., & Hu, K. (2020). A cooperative lane change model for connected and automated vehicles. IEEE Access, 8, 54940–54951. https://doi.org/10.1109/ACCESS.2020.2981169

[16] Ding, J., Peng, H., Zhang, Y., & Li, L. (2020). Penetration effect of connected and automated vehicles on cooperative on-ramp merging. IET Intelligent Transport Systems, 14(1), 56–64. https://doi.org/10.1049/iet-its.2019.0488

[17] Min, H., Fang, Y., Wang, R., Li, X., Xu, Z., & Zhao, X. (2020). A novel on-ramp merging strategy for connected and automated vehicles based on game theory. Journal of Advanced Transportation, 2020, Article 2529856. https://doi.org/10.1155/2020/2529856

[18] Chen, N., van Arem, B., Alkim, T., & Wang, M. (2021). A hierarchical model-based optimization control approach for cooperative merging by connected automated vehicles. IEEE Transactions on Intelligent Transportation Systems, 22(12), 7712–7725. https://doi.org/10.1109/TITS.2020.3007647

[19] Shi, Y., Yuan, Z., Yu, H., Guo, Y., & Qi, Y. (2021). A graph-based optimal on-ramp merging of connected vehicles on the highway. Machines, 9(11), Article 290. https://doi.org/10.3390/machines9110290

[20] Chen, N., van Arem, B., & Wang, M. (2022). Hierarchical optimal maneuver planning and trajectory control at on-ramps with multiple mainstream lanes. IEEE Transactions on Intelligent Transportation Systems, 23(10), 18889–18902. https://doi.org/10.1109/TITS.2022.3167727

[21] Fang, Y., Min, H., Wu, X., Wang, W., Zhao, X., & Mao, G. (2022). On-ramp merging strategies of connected and automated vehicles considering communication delay. IEEE Transactions on Intelligent Transportation Systems, 23(9), 15298–15312. https://doi.org/10.1109/TITS.2022.3140219

[22] Liu, B., Cen, Y., Yao, Z., Song, X., Liu, H., & Gao, H. (2022). Combined safety and coordination of connected automated vehicles in merging area with featuring optimal merging positions. Journal of Advanced Transportation, 2022, Article 2087510. https://doi.org/10.1155/2022/2087510

[23] Pang, M., & Huang, J. (2022). Cooperative control of highway on-ramp with connected and automated vehicles as platoons based on improved variable time headway. Journal of Transportation Engineering, Part A: Systems, 148(7), Article 04022034. https://doi.org/10.1061/JTEPBS.0000683

[24] Kavas-Torris, O., Gelbal, S. Y., Cantas, M. R., Aksun-Guvenc, B., & Guvenc, L. (2022). V2X communication between connected and automated vehicles (CAVs) and unmanned aerial vehicles (UAVs). Sensors, 22(22), Article 8941. https://doi.org/10.3390/s22228941

[25] Lv, P., Liu, H., Xu, J., & Li, T. (2022). Trajectory prediction with correction mechanism for connected and autonomous vehicles. Electronics, 11(14), Article 2149. https://doi.org/10.3390/electronics11142149

[26] Lyu, Y., Luo, W., & Dolan, J. M. (2022). Adaptive safe behavior generation for heterogeneous autonomous vehicles using parametric-control barrier functions. Proceedings of the AAAI Conference on Artificial Intelligence, 36(11), 13009–13010. https://doi.org/10.1609/aaai.v36i11.21641