vehicles

MDPI

Article

Gap Generation by Two Upstream Anchor Vehicles for Asymmetric Unsignalized Intersections

Csaba Antonya and Silviu Butnariu *

Department of Automotive and Transport Engineering, Transilvania University of Brașov, 29 Eroilor Blvd, 500036 Brașov, Romania; antonya@unitbv.ro
* Correspondence: butnariu@unitbv.ro

Abstract

Unsignalized priority junctions often suffer severe rush-hour congestion because main-road platoons block side-road merging. Most cooperative merging strategies assume many Connected and Automated Vehicles (CAVs), but near-term deployment must work with few. This paper studies a gap-generation strategy where only two automated "anchor" vehicles in a 20-car priority platoon (10% penetration) decelerate slightly upstream of an unsignalized merge. In traffic simulation with a rule-based gap arbiter and held-out scenarios, the strategy raises side-road throughput from nearly zero (~0.03 merges) to 8–9 merges per 20-car episode, at a delay cost of about 20 s. All anchor placements and delay weights fall along a linear throughput–delay frontier (~0.45 merges per second of peak delay), with the default anchor positions on an efficiency plateau rather than an isolated optimum. A Deep Reinforcement Learning (RL) policy based on Twin Delayed Deep Deterministic Policy Gradient (TD3) achieves reward performance statistically equivalent (within ±5%) to that of black-box optimization methods, namely Covariance Matrix Adaptation Evolution Strategy (CMA-ES) and Random Search, as well as a simple offline-tuned fixed setting. The ~10% theoretical advantage of a per-scenario Oracle disappears under 5% execution noise. These results suggest that complex RL policies may be over-engineered for static gap-creation tasks on single-lane approaches, while simple, robust CAV deceleration profiles offer an effective, low-complexity alternative for mixed-traffic bottlenecks.

Keywords: priority junction; cooperative merging; connected automated vehicles; low penetration; gap creation; reinforcement learning; black-box optimization

Check for updates

Academic Editors: Lázaro V. Cremades Oliver and Edwin Samir Pinto

Received: 5 November 2026
Revised: 23 November 2026
Accepted: 24 November 2026
Published: 27 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Unsignalized roundabouts and priority junctions are asymmetric: a free-moving priority road passes vehicles in groups, while a condensed queue on the secondary road waits for gaps that these groups rarely leave [1–3]. In typical urban networks, this occurs where a minor secondary street with condensed, queued commuter traffic feeds into a major priority corridor that carries a relatively light, free-flowing stream that arrives in groups (platoons). Acceptable merging gaps in the priority stream are extremely rare, causing extensive delays and queue spillback on the yielding secondary approach. A real-world asymmetric junction in Brașov, Romania, illustrates this challenge and provides qualitative motivation for this study [4] (Figure 1). The priority road is not saturated, yet the queue grows because the spaces between vehicles in a platoon are shorter than the gap the side road needs.

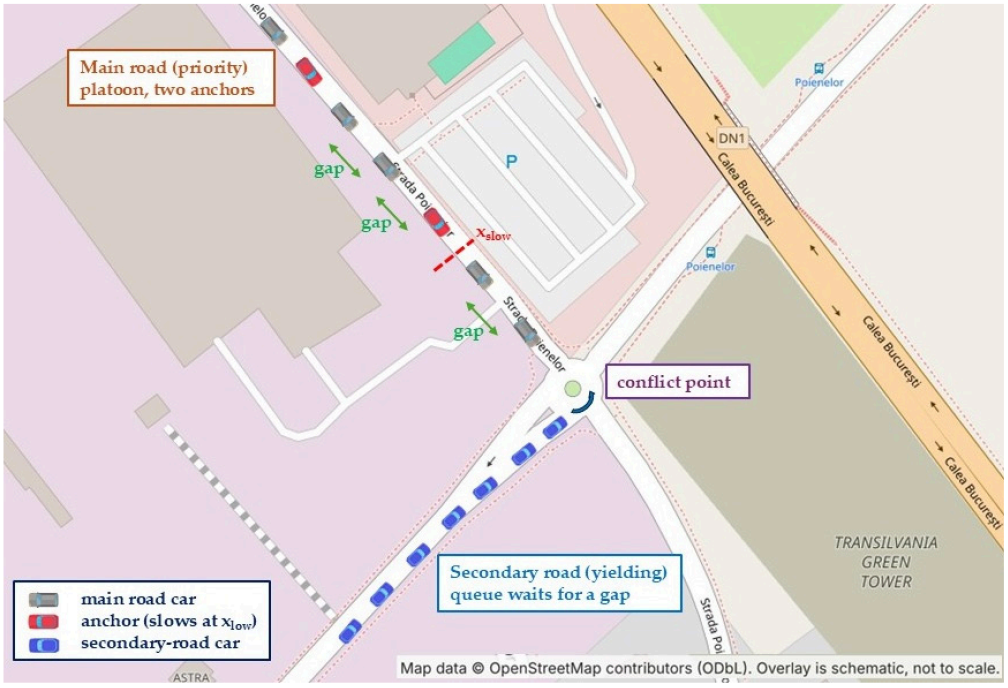


Figure 1. The junction: the priority platoon on the main road, the two anchors and the queue on the secondary road.

Connected and Automated Vehicles (CAVs) can, in principle, remove this bottleneck. Cooperative merging and signal-free junction management have been studied for highway on-ramps [5–7], intersections [8,9] and roundabouts [10–13].

Most of these methods coordinate all vehicles, or the merging vehicles, and their benefit grows with the CAV penetration rate [7,10,14,15]. Real traffic will contain only a small share of automated vehicles for years. A separate line of work shows that a few automated vehicles can change the behavior of an entire traffic stream, for example, by dissipating stop-and-go waves [16–18]. This raises the question that full coordination does not address: with only a few automated vehicles in the priority stream and no control over the other drivers, can these vehicles help the yielding queue merge?

In this paper, we study a targeted upstream anchor deceleration strategy for this asymmetric case: a lightly loaded main road with a single 20-car platoon and a condensed secondary queue of 20 vehicles. Two vehicles of the priority platoon, the anchors, slow down to a reduced speed when they pass a trigger position upstream of the junction; vehicles ahead of an anchor keep their speed and move away, opening a gap that yielding vehicles can use. Only the anchors are controlled, so 10% of the platoon is automated, and the scheme has four parameters (a trigger position and a reduced speed for each anchor). We evaluate how to select these parameters, comparing Deep Reinforcement Learning (RL) policy trained with the Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm with black-box evolutionary search (Covariance Matrix Adaptation Evolution Strategy, CMA-ES), uniform Random Search, an offline-tuned fixed setting, and a per-scenario Oracle. This comparison matters because reinforcement learning and optimization methods are frequently applied to merging problems [19–24] without verifying whether simpler alternatives would perform equally well [25–28].

The key scientific contributions are:

1. A strategy for cooperative merging with per-anchor parameters and a reward that weighs merges against per-car delay and jerk.
2. A reproducible merge simulator and evaluation protocol with fresh-seed scenarios.

3. An analysis of anchor placement showing a common throughput–delay curve and an efficiency plateau, with the pair used in the experiments as a representative operating point, not an optimum.
4. A comparison of Twin Delayed Deep Deterministic Policy Gradient (TD3) with Covariance Matrix Adaptation Evolution Strategy (CMA-ES), Random Search, a per-scenario Oracle, and a fixed setting, over several seeds, including tolerance to action noise.
5. A safety and comfort evaluation against the uncontrolled run, arguing that the gap rule, not the anchors, sets the safety floor.

Section 2 reviews related work and Section 3 describes the method. Section 4 analyzes the placement of the anchors and compares the parameter-selection methods, Section 5 studies the trade-off between merges and delay and the robustness of the scheme, Section 6 discusses the results and their limits, and Section 7 concludes.

2. Related Work

2.1. Signal-Free Junctions and Cooperative Merging

Coordinating Connected and Automated Vehicles (CAVs) at junctions and merges has a long history. Early work replaced signals with reservation protocols in which vehicles request space-time slots from an intersection manager [8], and later work added batch processing to avoid starving minor-road vehicles [9]. Surveys cover both intersection coordination and highway on-ramp merging and classify the approaches as centralized or decentralized [5]. For merging, a centralized optimal-control formulation assigns the vehicles a first-in, first-out merging order and computes closed-form optimal trajectories for the ramp and main-road vehicles [6], and hierarchical schemes combine sequence scheduling with motion planning [7]. Game-theoretic and MPC-based decision-making have been used in multi-lane merging zones [29], and platoon-specific controllers handle the insertion of a single vehicle into a platoon [30].

Roundabouts have received less attention than ramps and crossroads, but the same ideas have been applied. Optimal coordination of connected vehicles entering a roundabout reduced travel duration and fuel consumption in simulation, with the benefit depending strongly on the penetration rate [10]. Bi-level and hierarchical controllers choose entry sequences and merging times for vehicles approaching a roundabout [11,12], and a queueing analysis study compares signal-free roundabouts and intersections under pure CAV traffic [13]. Reviews of CAV roundabout simulation are available [3].

These methods control the merging vehicles, or all vehicles, and mostly assume full or high CAV penetration. In contrast, the proposed strategy controls strictly two “anchor” vehicles in a 20-car priority platoon (10% penetration). Secondary-road vehicles receive no control commands, and the merging maneuver itself remains entirely governed by a local, rule-based gap arbiter.

2.2. Gap Acceptance and Roundabout Capacity with Automated Vehicles

Roundabout entry capacity is conventionally modeled with gap-acceptance theory, in which a minor-road driver enters if the time gap in the priority stream is greater than a critical gap [3,31]; the same critical-gap and follow-up-time formulation also forms the basis of HCM capacity analysis at unsignalized intersections in general [1,2]. The critical and follow-up headways vary with intersection geometry and vehicle type and between countries, so locally calibrated values are recommended [31]. Simulation studies lower the critical and follow-up times as the share of automated vehicles grows and report capacity increases: roughly 10% and 20% higher leg capacity at 20% and 40% automated vehicles in one Vissim study [15], and HCM-style adjustment factors for CACC-equipped traffic in another [14]. In these studies, automated vehicles change the parameters of the gap rule

everywhere. In our setting, the gap rule is fixed ($G \geq G_{min}$ (16 m) and the other arbiter conditions), while the supply of special gaps is controlled by upstream anchor vehicles.

2.3. Sparse Automated Vehicles in Mixed Traffic

Previous studies show that a very small number of automated vehicles can change the behavior of a whole traffic stream. Field experiments with a single automated vehicle dissipated stop-and-go waves on a ring road [16]; control-theoretic analysis shows that mixed traffic on a ring is stabilizable by a few vehicles, with a speed increase above 6% at 5% penetration in the numerical example [17]; and model-based predictive control was validated experimentally with one or a few vehicles [18].

Reinforcement learning has been used to design such controllers for mixed autonomy; with the open-source Flow framework, learned control laws raised system-level velocity by up to 57% with only 4–7% automated vehicles [19]. Studies of merging in mixed traffic similarly find that moderate penetration (about 30% in one study) already mitigates shockwaves at on-ramps [7].

While prior mixed-autonomy control focuses on flow smoothing and wave dissipation on continuous ring roads or ramps, our framework controls CAVs for active gap creation at an unsignalized bottleneck, quantifying the exact trade-off between secondary merges gained and main-road delay.

2.4. Gap Creation by Main-Stream Vehicles

The closest studies let main-stream vehicles create gaps for merging traffic. Scarinci et al. [32] proposed a cooperative merging assistant for metered motorway on-ramps, in which one intelligent vehicle per signal cycle slows down, and the gap opening ahead of it is synchronized with the ramp signal; validated in microscopic simulation, it reduced congestion and late merges, assuming homogeneous drivers and full availability of cooperative vehicles. Chen et al. [33] let a main-line CAV slow down temporarily at freeway merging bottlenecks, so that the spacing behind it accumulates into gaps for batch insertions; this raises throughput even at low CAV penetration but may distribute delays unevenly. Closest to the present problem, Alanazi et al. [34] let major-road CAVs at an unsignalized T-intersection slow down to extend their gap when a minor-road vehicle was waiting; with 30% to 70% CAVs, minor-road delay fell by up to 72% at a small cost to the major road.

The present study shares this principle but differs in four respects.

i. Only two platoon vehicles (10%) are controlled, by a single open-loop decision of four parameters made before the platoon arrives, rather than by reactive per-CAV decisions [34] or per-cycle coordination with ramp metering [32].
ii. The setting is an urban unsignalized priority junction, not a motorway on-ramp [32] or freeway bottleneck [33].
iii. The parameters are optimized for a reward weighing merges against delay, jerk and braking energy, and five selection methods are compared on paired held-out scenarios, whereas refs. [32,33] derive the control analytically and ref. [34] uses a fixed rule.
iv. The cost of the gaps is quantified as a linear throughput–delay frontier, with a cap on worst-case main-road delay that addresses the equity concern raised in [33].

2.5. Learning-Based Merging

Reinforcement learning has been applied extensively to merging problems. Typical formulations control the acceleration of one merging vehicle with Deep Deterministic Policy Gradient (DDPG) or TD3 [20,23], coordinate several CAVs with multi-agent actor-critic methods [21,22], or learn behavior at a work-zone merge [24]. Jerk and comfort enter as reward terms in some of these [23]. Rather than learning a per-timestep trajectory policy

for merging cars, our formulation sets four static parameters once per episode, before the platoon arrives. The control task thus becomes a one-step decision problem, which allows a direct comparison of Deep RL (TD3), derivative-free black-box optimizers (CMA-ES, Random Search), a fixed setting and a per-scenario Oracle.

2.6. Baselines Used for Parameter Search

The vehicle model is the intelligent driver model (IDM) [35]. We compare TD3 [36] with CMA-ES [25], an evolution strategy, and with Random Search. This comparison follows the argument that evolution strategies and Random Search are competitive with deep RL on continuous control [26,27] and the more general caution that RL results depend strongly on seeds, hyperparameters, and implementation, so that several seeds must be reported [28]. A comparative survey discusses when each family is preferable [37].

2.7. Positioning Against Prior Work

An earlier preliminary investigation of the Brașov roundabout [4] slowed every tenth main-road vehicle with a single shared speed reduction tuned via a genetic algorithm; it did not evaluate individual parameter vectors, placement or efficiency, used no held-out evaluation protocol, and did not test main-road delay caps or execution noise. In the literature we surveyed, CAV roundabout and merging control either coordinates all vehicles, learns the control of the merging vehicle, or lets main-stream vehicles slow down to create gaps at motorway merges [32,33] or, reactively and at moderate-to-high CAV shares, at an unsignalized T-intersection [34]. We did not find a study that fixes the gap rule and the secondary traffic at an unsignalized priority junction and shapes the gaps with two parameterized slow-downs of upstream main-road vehicles at 10% penetration, quantifying the resulting throughput–delay frontier and comparing learning-based and black-box methods for selecting the parameters. Table 1 summarizes how this study differs from the related work.

Table 1. Related work and how this study differs from it.

Group	Examples	What Is Controlled	CAV Share Assumed	Merging or Entry Rule	Difference from This Study (for This Study: Main Features)
Reservation-based intersection management	[8,9]	Every vehicle, through slot requests to an intersection manager	Full (human-driven vehicles only in an extension of [8])	Reserved space-time slots	Control of all vehicles; no priority stream
Highway on-ramp merging, optimal control and scheduling	[5–7,29,30]	Merging and main-road vehicles; sequence and trajectories	Full, or varied (about 30% studied in [7])	Scheduled sequence, game-theoretic or MPC decisions	Merging CAVs are controlled; human-driven vehicles in [7] wait for a gap
Roundabout coordination	[10–13]	Vehicles entering and circulating; entry sequence and merge time	Full, or varied (benefit depends on penetration in [10])	Optimized entry sequence or bi-level control	Entering vehicles are controlled; comparison is by travel time and fuel [10,11], throughput [12], or capacity and delay [13]
Roundabout gap acceptance and capacity with automated vehicles	[2,3,14,15,31]	No active control; the parameters of the gap rule change with the AV share	0 to 100% in [15]	Critical and follow-up gap, HCM adjustment factors	The gap rule itself changes; here the rule is fixed, and the supply of gaps is shaped

Table 1. *Cont.*

Group	Examples	What Is Controlled	CAV Share Assumed	Merging or Entry Rule	Difference from This Study (for This Study: Main Features)
Few vehicles in mixed traffic (stop-and-go waves)	[16–19]	One or a few automated vehicles that smooth the whole flow	One vehicle to about 5–7%	None; flow smoothing, mostly on a ring road	Same principle (few vehicles act on many), different goal: open gaps for a merging stream
Gap creation by main-stream vehicles	[32–34]	One or a few main-stream vehicles slow down upstream of the merge: one per ramp-signal cycle [32], periodic gap redistribution [33], any CAV whose front gap is below the critical gap [34]	Full in [32]; varied in [33]; 30–70% in [34]	Ramp metering with gaps synchronized to releases [32]; batch lane changes [33]; critical gap from HCM [34]	Motorway or T-intersection; analytical control [32,33] or reactive rule [34]; here, 10%, one open-loop decision per platoon, optimized parameters, explicit throughput–delay frontier and delay cap
Learning-based merging	[20–24]	One merging vehicle, or several, per time step; speed or acceleration	CAVs only in [21,22,24]; mixed traffic in [20]; uncontrolled IDM main-road vehicles in [23]	Learned policy with per-step control	A per-step policy for merging vehicles; whereas here, there is one decision per episode
Earlier study of the same junction	[4]	Every tenth vehicle on the priority road, one shared speed reduction and distance	About 10%	Distance-based yield rule (11 ± 3 m)	Shared parameters tuned by a genetic algorithm; no individual anchors, no placement analysis, no held-out protocol
This study		Two main-road vehicles (cars 5 and 15 of 20), four parameters	10%	Fixed gap rule	Individual anchors; placement study; fairness cap; five selection methods compared on held-out scenarios; safety and robustness

3. Methodology

The symbols used below are listed in Table A1 (Appendix A). All simulations, the TD3 agent (Reinforcement Learning Toolbox, rlTD3Agent), CMA-ES and Random Search were implemented in MATLAB R2026b (The MathWorks, Inc., Natick, MA, USA). The map in Figure 1 is from OpenStreetMap.

3.1. The Merge Simulator

The junction is reduced to two one-dimensional paths that meet at the conflict point $x = 0$. Main-road vehicles move along the coordinate x; secondary-road vehicles move along their own approach, with $x < 0$ before the yield line at $x = 0$. A secondary vehicle that merges continues on the main road, in the same coordinate, and becomes part of the main stream. All vehicles are removed at $x = 80$ m. The speed limit is 50 km/h outside $|x| = 50$ m and 30 km/h within $|x| \leq 50$ m of the conflict point, on both approaches and downstream. No lane changes, overtaking, or circulating geometry are modeled. The model is therefore a stylized priority junction, not a replica of the Brașov roundabout (Figure 1).

The model is deliberately simple. It represents one single-lane priority approach and one single-lane secondary approach that meet at a single conflict point, so lane changing and overtaking are not modeled. All human-driven vehicles follow the same IDM parameters,

and only two vehicles on the main road (10% penetration) are automated. This abstraction isolates the gap-creation mechanism and allows the large number of simulations needed to compare selection methods on held-out scenarios. It also means that the absolute numbers, in particular the merge counts, should be read as indicative for single-lane approaches. The sensitivity of the results to the car-following and gap-acceptance assumptions is examined in Section 5.5. Analogous simplifications are standard in research on gap generation by the main-stream vehicles: one lane of the mainstream traffic [33] and homogeneous drivers without lane changing into the lane where gaps occur [32].

3.2. Vehicle Dynamics

All vehicles follow the intelligent driver model (IDM) [35]. The acceleration of vehicle i with speed v, desired speed v_0, bumper gap s, and approach rate Δv to its leader is (1):

$$a = a_{max}\left[1 - \left(\frac{v}{v_0}\right)^{\delta} - \min\left(\left(\frac{s^*}{s}\right)^2, 4\right)\right], \qquad s^* = s_0 + \max\left(0,\ vT + \frac{v\,\Delta v}{2\sqrt{a_{max}\,b}}\right) \tag{1}$$

with the acceleration clipped to $[-4.5, 2.0]$ m/s^2, with jam distance $s_0 = 2.0$ m, desired time headway $T = 0.6$ s, maximum acceleration $a_{max} = 2.0$ m/s^2, comfortable deceleration $b = 2.2$ m/s^2, acceleration exponent $\delta = 6$ and vehicle length $L = 4.5$ m. The desired speed is the local limit, overridden for the anchors by (1). Time is advanced with the forward Euler method at 0.05 s, and speeds cannot become negative. All vehicles use the same parameters, so differences between scenarios come from the initial state, not from driver heterogeneity.

Using one car-following model for automated and human-driven vehicles is a simplification. Road tests with production vehicles [38] showed that an IDM controller follows its leader smoothly but with a slow response and large variations of the clearance, while CACC operated at a 0.6 s time gap gives stable following; simple feedback models of ACC and CACC were calibrated on these data. One model is nevertheless used here, as in other gap-creation studies [33,34], for two reasons. First, the anchors act only through their desired speed, so the gap they open is set mainly by the speed difference to their leader and by the time spent at the reduced speed, and depends little on the details of the car-following law. Second, a common model keeps the controlled and uncontrolled runs of each scenario directly comparable, so that the differences come only from the anchor commands. The jam distance $s_0 = 2$ m is that of the original IDM [35]. The time headway $T = 0.6$ s represents dense rush-hour traffic and equals the CACC time gap tested in [38]; because it is shorter than the 1.1 s used there for ACC and shorter than typical human headways, Section 5.5 also uses $T = 1.0$ s and 1.4 s, for which the fixed anchor vector still yields 7.1 and 5.5 merges. The values $a_{max} = 2.0$ m/s^2 and $b = 2.2$ m/s^2 describe urban driving at 30–50 km/h and are close to the comfortable deceleration of 2 m/s^2 used for the IDM in [38].

3.3. Secondary Approach and the Rule-Based Gap Arbiter

Secondary vehicles form a first-in-first-out queue. A vehicle follows its leader and stops at the yield line with the same IDM. When the vehicle at the head of the queue is within 30 m of the yield line, and no other secondary vehicle is committed, an arbiter checks the main stream. Vehicle positions refer to the front bumper. Let x_d be the position of the nearest main-stream vehicle already past the conflict point, x_u the position of the nearest main-stream vehicle that has not yet reached it, and v_u its speed. The free gap G is the clear distance between the rear bumper of the downstream vehicle, located at $x_d - L$ ($L = 4.5$ m is the vehicle length), and the front bumper of the upstream vehicle, as in the definition of the merging gap of [32] (2):

$$G(t) = (x_d(t) - L) - x_u(t). \tag{2}$$

G is the space into which the merging vehicle must fit. It is not a fixed value: it is recomputed at every time step (0.05 s) from the current positions and changes as the vehicles move; in particular, it grows when an anchor slows down behind a leader that keeps its speed. What is fixed is the acceptance threshold $G_{min} = 16$ m, which corresponds to the length of the merging vehicle (4.5 m) plus a clearance of about 5.75 m to each of the two main-road vehicles. The time t_u the upstream vehicle needs to reach the conflict point is (3):

$$t_u = \frac{|x_u|}{\max(0.5,\ v_u)} \tag{3}$$

where the lower bound of 0.5 m/s avoids a division by zero for a stopped vehicle.

The vehicle commits to merging when all the following hold:

- $G \geq G_{min}$ (16 m);
- $x_d \geq 8$ m;
- $t_u \geq 2\text{ s} + t_{entry}$, where $\Delta t_m = 2$ s is the arbiter time margin and t_{entry} is the time needed for the merging vehicle to reach the conflict point.

The time t_{entry} is estimated at the moment of the decision from the distance $d = |x_s|$ between the vehicle at the head of the secondary queue and the conflict point and from its current speed v_s, assuming acceleration at a_{max} up to the local limit $v_{lim} = 30$ km/h and constant speed afterwards (4):

$$t_{entry} = \frac{\sqrt{v_s^2 + 2 \cdot a_{max} \cdot d} - v_s}{a_{max}} \text{ if } d \leq d_{acc};$$

$$t_{entry} = \frac{(v_{lim} - v_s)}{a_{max}} + \frac{(d - d_{acc})}{v_{lim}}; \text{otherwise with} \tag{4}$$

$$d_{acc} = \frac{(v_{lim}^2 - v_s^2)}{2 \cdot a_{max}}$$

The 30 km/h limit is therefore considered. Because the IDM acceleration decreases as the speed approaches v_{lim}, Equation (4) slightly underestimates the actual entry time, a difference absorbed by the margin Δt_m. This last condition also prevents a slow upstream vehicle from arriving together with the merger.

Because a purely spatial gap ignores the speed of the upstream vehicle, it is combined with the temporal condition of Equation (3), in line with gap-acceptance theory [3,31]. Section 5.5 will show that, for thresholds between 12 m and 20 m, the spatial condition does not bind, and the temporal margin is what limits the merges. A committed vehicle accelerates freely toward the 30 km/h limit, and its merge is counted when it passes $x = 0.5$ m, after which it joins the main stream as an ordinary IDM follower. Only one vehicle can be committed at a time.

3.4. Scenario Generation

In the analyzed asymmetric intersection, each episode models a sparse main-road priority platoon ($N_p = 20$ vehicles, a flow of about 1300 to 2100 vehicles per hour while the platoon passes) and a dense secondary queue ($N_s = 20$ vehicles). Initial states are generated with uniform random sampling (Table 2). The ranges are not calibrated to measured data. The initial positions define plausible rush-hour traffic, describing the situation in which the question of gap creation arises.

The rejection rule concerns car-following conflicts only: a scenario is rejected if, at $t = 0$, a vehicle is closer to its leader than its IDM desired gap $s^*(v, \Delta v)$, so that it would have to brake because of the vehicle ahead. Decelerations caused by the speed limit are not a reason for rejection: the first vehicles of the main platoon start inside the 30 km/h zone above the limit and decelerate to it during the first seconds, as prescribed by the IDM. This

transient is identical in the controlled and the uncontrolled run of each scenario and ends before the anchors reach their trigger positions, so it affects neither the delays, which are measured against the uncontrolled run, nor the number of merges. The 50 km/h limit is the general urban speed limit in Romania and most European countries, and the 30 km/h zone corresponds to the entry speeds recommended for single-lane roundabouts (about 30–40 km/h [39]). The initial main-road speeds (45 to 52 km/h) represent free flow close to the urban limit; with the sampled spacing they correspond to headways of about 1.7 to 2.8 s, and the secondary-queue speeds (25 to 32 km/h) to vehicles approaching inside the reduced-speed zone.

Table 2. Scenario parameters and random seeds (simulator and arbiter parameters are given in Sections 3.1–3.3).

Group	Parameter	Value
Episode	End	last main-road vehicle reaches x = 0, or 150 s
Main platoon	Vehicles	20; leader at −20 to −10 m; spacing 25 to 35 m; speed 45 to 52 km/h
Secondary queue	Vehicles	20; leader at −25 to −12 m; spacing 8 to 12 m; speed 25 to 32 km/h
Sampling	Rule	uniform draws; scenarios in which a vehicle starts closer to its leader than its IDM desired gap s* (Equation (1)) are rejected; vehicles that start inside the 30 km/h zone above the limit decelerate to it
Seeds	Placement training	700001 to 700008
Seeds	Placement evaluation	1100001 to 1100100
Seeds	Learning study, held out	900001 to 900100

The episode ends when the last main-road vehicle reaches x = 0, or after 150 s. Secondary vehicles that have not merged by then do not count.

3.5. The Sparse Automated Anchors

The scheme controls exactly two automated vehicles of the 20-car main-road priority platoon: cars 5 and 15, located at the midpoints of the two 10-car sub-platoons (10% market penetration). All other priority vehicles and all secondary road vehicles are human-driven (simulated as IDM agents) and follow standard car-following and gap-acceptance rules.

An anchor k has two parameters, a trigger position $x_{slow,k}$ and a reduced speed $v_{slow,k}$. After crossing $x_{slow,k}$, the vehicle follows its car-following model with desired speed

$$v_{0,k}(x) = \begin{cases} \min\left(v_{slow,k},\, v_{lim}(x)\right), & x_{slow,k} \le x < 0 \\ v_{lim}(x), & \text{otherwise} \end{cases} \tag{5}$$

where $v_{lim}(x)$ is the local speed limit (50 km/h upstream, 30 km/h within 50 m of the conflict point). The anchor therefore never brakes harder than its car-following model prescribes and resumes the limit after the conflict point. Vehicles ahead of the anchor are not affected; the gap between the anchor and its leader grows at the rate $v_{leader}-v_{anchor}$, and the vehicles behind the anchor close behind it (an illustrative example is shown in Figure 2). As a first-order estimate, the gap created when the anchor reaches the conflict point is about $(v_{leader}-v_{slow})$ times the time spent slowing. A larger gap helps the secondary queue but delays the vehicles behind the anchor, which is the trade-off studied in this paper.

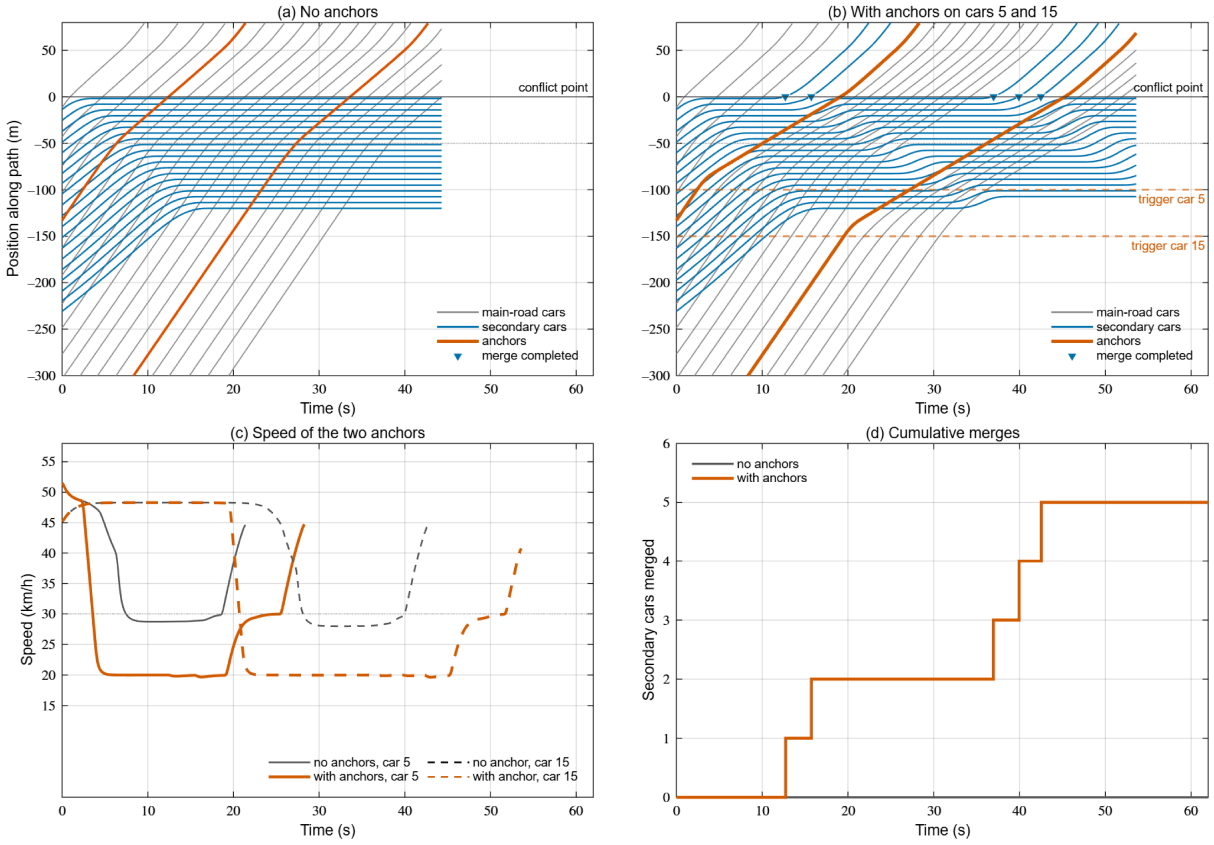


Figure 2. Gap creation in one logged scenario: (**a**) Without anchors, the 20-car main-road platoon (gray) passes the conflict point and no secondary vehicle (blue) merges. (**b**) With anchors on cars 5 and 15 (orange), which slow to 20 km/h at x = −100 m and x = −150 m, gaps open ahead of each anchor and five secondary vehicles merge (triangles). (**c**) Speed of the two anchors with and without control. (**d**) Cumulative number of merged secondary vehicles. The anchor setting is illustrative, not optimal, and the worst-case delay of a main-road car is 11.6 s.

The scheme used throughout has two anchors, cars 5 and 15, with parameter vector

$$\boldsymbol{\theta} = \left(x_{slow,5},\ v_{slow,5},\ x_{slow,15},\ v_{slow,15}\right) \tag{6}$$

The choice of positions is examined in Section 4; cars 5 and 15 are a representative operating point, not an optimum.

3.6. Objective and Reward

Because θ is selected once per episode, before the platoon reaches $x_{slow,5}$, parameter selection is a single-decision optimization problem rather than a multi-step Markov Decision Process. For a scenario ξ (initial positions and speeds of all vehicles) and a parameter vector θ, the simulator returns:

- $n(\xi, \theta)$: number of secondary vehicles that complete the merge.
- Individual vehicle delay d_i—delay of main-road vehicle i, where t_i is the time at which it reaches the conflict point and $t_i(\text{none})$ the time in the uncontrolled run of the same scenario. Mean delay $\bar{d}$, and peak delay D_{max} (7):

$$\begin{aligned} d_i &= \max(0, t_i(\theta) - t_i(\text{none})) \\ \bar{d} &= \left(1/N_p\right) \Sigma_i\, d_i, \\ D_{max} &= \max_i d_i \end{aligned} \tag{7}$$

- J: root-mean-square (RMS) jerk of the main-road vehicles:

$$J = \sqrt{\frac{1}{N_p}\sum_{i=1}^{N_p}\frac{1}{t_{end}}\int_0^{t_{end}}\left(\frac{da_i}{dt}\right)^2 dt} \tag{8}$$

- E: braking energy summed across all main-road vehicles, in J/kg:

$$E = \sum_{i=1}^{N_p}\int_0^{t_{end}} \max(0, -a_i(t))\cdot v_i(t) dt \tag{9}$$

- Minimum Time-to-Collision (TTC_{min}): for each car-following pair on the priority road, in which vehicle i follows vehicle $i-1$ at bumper spacing s_i, the instantaneous TTC is defined as (10):

$$\begin{aligned} TTC_i(t) &= \tfrac{s_i(t)}{v_i(t)-v_{i-1}(t)} \text{ if } v_i(t) > v_{i-1}(t); \\ TTC_i(t) &= +\infty \text{ if } v_i(t) \le v_{i-1}(t). \end{aligned} \tag{10}$$

A collision course exists only while the follower is faster than its leader. When the follower is slower or equally fast, for example while it decelerates behind an anchor, the spacing does not decrease and the TTC is infinite, so TTC never takes negative values [40]. TTC_{min} is the minimum of $TTC_i(t)$ over all pairs and over the whole episode.

- Efficiency (η): merges per second of worst-case delay, mean(n)/mean (D_{max}) over scenarios; used in Section 4.2 to compare placements without the weights of the reward.

To balance secondary road throughput against priority-road traffic disturbances, discomfort, energy expenditure, and extreme delays, the physical environment reward function $R(\xi, \boldsymbol{\theta})$ is defined as (11):

$$R(\xi, \boldsymbol{\theta}) = w_n n^{1.1} - w_d \sum_{i=1}^{N_p} d_i^2 - w_J J^2 - w_E E - w_f \max\left(0, D_{max} - D_{cap}\right)^2 \tag{11}$$

The throughput term is slightly convex (exponent 1.1) so that each additional merge is worth a little more than the last. The delay term is quadratic, so that a few long delays cost more than many short ones. The jerk and energy terms penalize discomfort and braking. The last term is a fairness penalty, applied only when a worst-case delay cap D_{cap} is imposed. The weights used are w_n, w_d, w_J, w_E, and w_f (Table A1). These weights are design choices. Their effect on the trade-off is examined in Sections 4.3.1 and 5.1, and their effect on the placement in Section 5.3, and the ranking of placements by efficiency does not depend on them.

In the learning experiments of Section 4.3, the reward also contains a centering term and a barrier term on the normalized action (Equation (14)) and no cap penalty. Rewards of the placement study (Section 4.2) and of the learning study are therefore not comparable and are reported in separate tables.

3.7. Parameter Selection Method and Policy Learning Framework

As stated in Section 3.6, before the priority platoon reaches the first trigger line, the controller emits a single 4-dimensional normalized action vector a. Four methods, which differ in the state information and computation they require, are compared.

3.7.1. Deep Reinforcement Learning with TD3

A Twin Delayed Deep Deterministic Policy Gradient (TD3) agent tests whether a state-feedback policy can exploit scenario-specific vehicle arrangements:

- State observation space ($s \in \mathbb{R}^{80}$). The network receives an 80-dimensional state vector s containing the normalized initial positions x and initial speeds v of all 40 vehicles (20 priority-road platoon cars and 20 secondary-queue cars) sampled at t = 0 s. State observation assumes full V2X fleet visibility.

- Actor network architecture. Multi-layer perceptron (MLP) with layer structure 80–256–128–4, utilizing Rectified Linear Unit (ReLU) hidden activations and a Tanh output layer scaled to the normalized action box $[-1, 1]^4$.
- Twin critic network architecture. Two independent critic networks, each taking the concatenated state-action pair $(s, \boldsymbol{a}) \in \mathbb{R}^{84}$ as input through an 84–256–128–1 architecture with ReLU activations.
- Training hyperparameters. A total of 20,000 single-step episodes, discount 0.99, mini-batch 256, actor learning rate 1×10^{-4}, critic learning rate 1×10^{-3}, target smoothing 0.005, delayed policy updates every 2 steps, replay buffer 2×10^5.

3.7.2. Offline Black-Box Evolutionary and Stochastic Optimizers

Two derivative-free black-box algorithms search, with the same budget as TD3, for a single fixed setting used for all scenarios:

1. Covariance Matrix Adaptation Evolution Strategy (CMA-ES)—a $(\mu/\mu_w, \lambda)$-CMA-ES [25] algorithm operating directly within the normalized action box $[-1, 1]^4$. In this standard notation, λ is the number of candidate vectors sampled in each generation, μ is the number of best candidates (parents) used to update the search distribution, the subscript w indicates weighted recombination (the new mean is a weighted average of the μ best candidates, with weights decreasing with their rank), and the comma indicates that parents are selected only among the new candidates. We use $\lambda = 12$, $\mu = \lambda/2 = 6$ and an initial step size $\sigma_0 = 0.5$. In each generation, candidate vectors evaluate 20 fresh training scenarios, completing 20,000 total simulation evaluations (approx. 83 generations) per run.
2. Uniform Random Search—candidate action vectors are drawn uniformly from $[-1, 1]^4$ and evaluated on 20 training scenarios each; the best vector found within 20,000 simulation evaluations is retained.

3.7.3. Per-Scenario Oracle Upper Bound

A per-scenario Oracle gives an upper bound on open-loop performance (12):

$$\boldsymbol{\theta}^*_{\xi} = \operatorname{argmax}_{\boldsymbol{\theta}} R(\xi, \boldsymbol{\theta}) \text{ for each scenario } \xi \tag{12}$$

The Oracle searches separately on every evaluation scenario ξ for the parameter set $\boldsymbol{\theta}^*_{\xi}$ that is best for that scenario. Because it uses the outcome of the scenario itself, which no deployed real-time controller can know, it serves only as a theoretical upper bound.

3.7.4. Protocol Parity, Seed Partitioning and Budget Equivalence

To ensure a fair comparison of all methods:

- Equal simulation budget: 20,000 simulations per training run for TD3, CMA-ES and Random Search.
- Disjoint seeds: parameters are optimized only on training seeds (seeds 700001–700008) and all reported results are computed on 100 held-out evaluation scenarios (seeds 900001–900100; Table 2).
- Replication: 5 independent training runs for TD3 (seeds 1–5) and 3 each for CMA-ES and Random Search. Methods are compared at the level of runs. Differences are tested with Welch's *t*-test, and equivalence with the two one-sided tests procedure (TOST, $\alpha = 0.05$) and a margin of $\pm 5\%$ of the reward.

4. Choice of Anchor Positions and Selecting the Anchor Parameters

4.1. Question and Design

Two design questions are addressed: how many anchor vehicles are required, and where within the priority platoon they should be placed to increase secondary throughput with little main-road disruption.

Every configuration gets its own anchor parameters, tuned by an evolution strategy, and is evaluated on 100 fresh scenarios that were never seen for tuning. The scenarios are identical across configurations, so reward differences are paired.

4.2. Comparative Study of Anchor Configurations

Table 3 lists ten configurations. The notation (a,b) means anchors on cars a and b of the 20-car platoon.

Table 3. Ten anchor configurations (fresh seeds 1100001 to 1100100; Section 4.2 protocol).

Configuration	Merges	Mean Delay (s)	Worst Delay (s)	RMS Jerk (m/s^3)	Min TTC (s)	Reward	Reward Minus (5,15)
No anchor	0.03 ± 0.03	0.00	0.0	3.21	1.90	−70.8	−331.7 ± 8.7
Single 5	3.90 ± 0.09	5.48	10.9	1.55	1.45	62.5	−198.4 ± 8.1
Single 15	6.87 ± 0.08	4.84	17.5	2.69	1.44	190.6	−70.3 ± 8.6
(5,15)	8.42 ± 0.12	6.73	18.7	1.93	1.43	260.9	reference
(4,15)	8.50 ± 0.11	6.45	18.3	2.00	1.43	284.5	+23.6 ± 9.2
(3,15)	9.86 ± 0.08	7.58	20.8	1.88	1.44	304.7	+43.8 ± 8.2
(6,15)	8.66 ± 0.10	7.08	19.3	2.03	1.43	255.1	−5.9 ± 8.0
(7,15)	8.06 ± 0.11	6.76	18.8	2.13	1.43	217.5	−43.4 ± 9.6
(5,12)	7.72 ± 0.10	7.50	17.8	2.23	1.43	156.6	−104.3 ± 7.5
(5,10,15)	7.84 ± 0.09	5.58	17.8	2.54	1.44	256.6	−4.3 ± 7.9

Table 3 shows that without anchors almost no secondary vehicle merges. A single anchor already helps, and a pair helps more: (5,15) achieves about 1.2 times the merges of a single anchor at car 15 and about 2.2 times those of a single anchor at car 5, and the corresponding reward differences (−70 and −198) are far outside the confidence intervals. The position of the first anchor matters little within a small range: car 5 or 6 gives similar results (8.4 to 8.7 merges), whereas moving it back to car 7 lowers the reward by 43. The position of the second anchor is more critical: bringing it forward to car 12 is clearly worse (−104), because the later gap is then no longer opened where the queue needs it.

A third anchor does not improve the outcome in a detectable way. The configuration (5,10,15) is tied with (5,15) in reward (−4.3 ± 7.9): it trades fewer merges (7.84 instead of 8.42) for a lower mean delay (5.58 s instead of 6.73 s), at a higher jerk (2.54 instead of 1.93 m/s^3). Finally, the two anchors do not contribute equally: with (5,15), about 20% of the secondary vehicles merge ahead of cars 1 to 14, in the gap of the first anchor, and about 80% ahead of cars 15 to 20, in the gap of the second.

Moving the second anchor towards the end of the platoon keeps raising the reward: in an uncapped scan with the first anchor at car 5, a second anchor at car 19 gives 19.2 merges but a worst-case delay of 43.3 s (reward 881). Because the platoon is finite, a late anchor delays only a few vehicles behind it, while the secondary vehicles merging in its gap still count in full. A reward-optimal design would therefore drive the second anchor to the tail of the platoon and accept very long delays for the last cars, the opposite of what a traffic manager wants. Two safeguards are used: a fairness penalty on the worst-case delay (21 s) in the reward, and comparison of placements by an efficiency measure that does not depend on the reward weights.

4.2.1. An Efficiency Plateau

Tables 4 and 5 and Figure 3 show the efficiency η for different positions of the second anchor (first anchor at car 5) and of the first anchor (second anchor at car 15), without a delay budget and with a budget of 21 s; two independent runs were made for each position. The pair (5,15) was fitted separately in each scan; hence, $\eta = 0.452$ in Table 4 and 0.450 in Table 5.

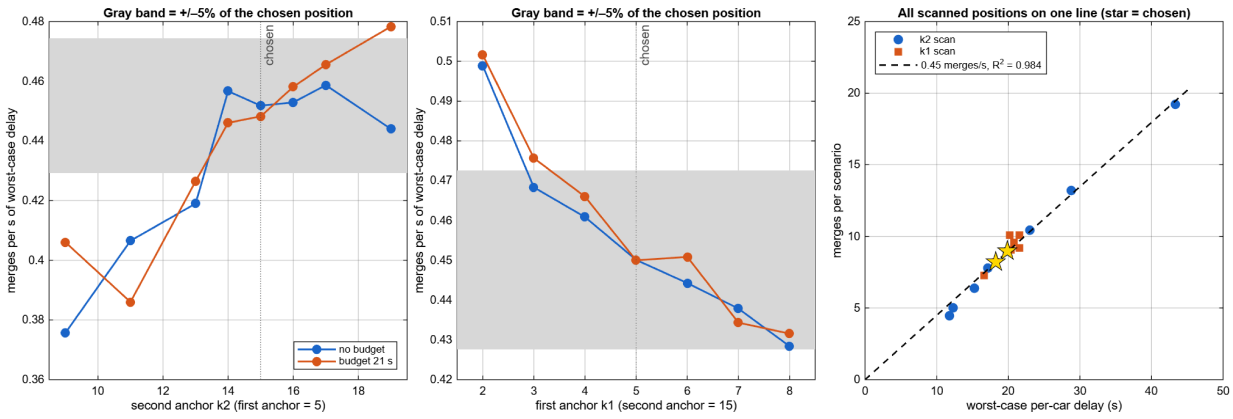
Table 4. Efficiency (η) against the position of the second anchor, first anchor at car 5 (Section 4.2 protocol).

Second Anchor Position	Efficiency η (No Budget)	Relative to Car 15	Efficiency (21 s Budget)	Relative to Car 15, No Budget
9	0.376	−16.9%	0.406	−10.1%
11	0.407	−10.0%	0.386	−14.6%
13	0.419	−7.3%	0.426	−5.6%
14	0.457	+1.1%	0.446	−1.3%
15 (baseline)	0.452	0.0%	0.448	−0.8%
16	0.453	+0.2%	0.458	+1.4%
17	0.459	+1.5%	0.466	+3.0%
19	0.444	−1.7%	0.478	+5.9%

Table 5. Efficiency against the position of the first anchor, second anchor at car 15 (Section 4.2 protocol).

First Anchor Position	Efficiency η (No Budget)	Relative to Car 5	Efficiency (21 s Budget)	Relative to Car 5, No Budget
2	0.499	+10.8%	0.502	+11.5%
3	0.468	+4.1%	0.476	+5.7%
4	0.461	+2.4%	0.466	+3.5%
5 (baseline)	0.450	0.0%	0.450	0.0%
6	0.444	−1.3%	0.451	+0.2%
7	0.438	−2.7%	0.434	−3.5%
8	0.428	−4.8%	0.432	−4.1%

Within a band of ±5% around the efficiency of the chosen position (Figure 3), the second anchor can be at car 14, 15, 16, 17 (and 19 without a budget); the first anchor can be at car 4 to 8 (and 3 without a budget). Efficiency is clearly lower when the second anchor is at car 13 or earlier (−5.6% to −16.9%), and it decreases slowly when the first anchor moves back from car 2 to car 8 (from 0.499 to 0.428). The pair (5,15), at the midpoint of each 10-vehicle block, is thus a representative operating point on an efficiency plateau rather than a fragile optimum.



(**a**) (**b**) (**c**)

Figure 3. Efficiency (η, merges per second of worst-case delay) against anchor position. (**a**) Second anchor moved from car 9 to car 19 with the first anchor at car 5, without a delay budget and with a 21 s budget. (**b**) First anchor moved from car 2 to car 8 with the second anchor at car 15. (**c**) Merges per scenario against the worst-case delay for all scanned positions, with the fitted line (0.45 merges/s, $R^2 = 0.984$); the stars mark the chosen pair (5,15). The gray bands in (**a**,**b**) show ±5% around the efficiency of the chosen position.

4.2.2. One Trade-Off Curve

Across 15 scanned anchor positions without delay caps, secondary throughput (n) and peak priority delay (D_{max}) follow a strict linear trade-off relationship (Figure 4):

$$n \approx 0.449 \cdot D_{max} \left(R^2 = 0.984,\ \text{residual SD } = 0.45 \text{ merges}\right) \quad (13)$$

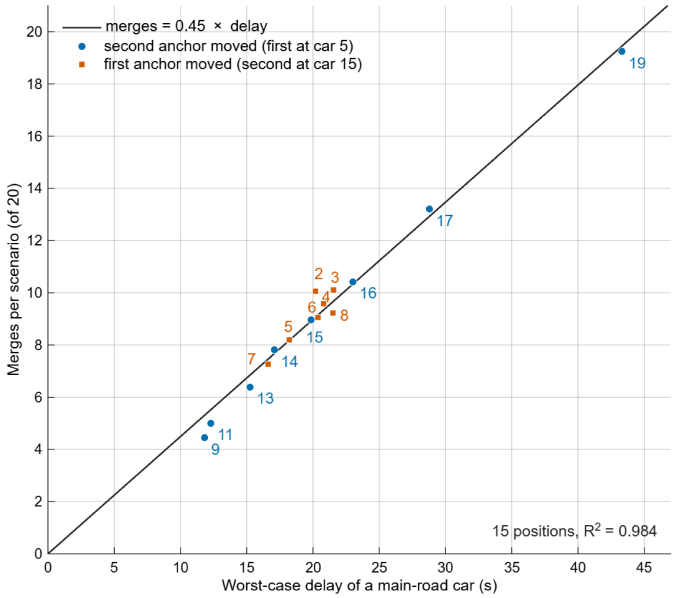


Figure 4. Merges per scenario against the worst-case delay of a main-road car for 15 scanned points (14 distinct anchor configurations) without a delay budget: the second anchor moved to cars 9 to 19 with the first at car 5 (blue, labeled with the car number), and the first anchor moved to cars 2 to 8 with the second at car 15 (orange; the pair (5,15) appears in both series). Each point is the mean of two fits evaluated on 100 fresh scenarios (seeds 1100001 to 1100100).

Modifying anchor positions or delay weights therefore moves the operating point along a single curve (about 0.45 additional merges per second of peak delay) rather than shifting the frontier; the choice of position becomes the choice of the delay the operator is

ready to accept. Delays along the platoon also differ: for the pair (5,15), the first-anchor delay falls from 5.5 s at car 5 to 0.9 s at car 14, whereas for a single anchor at car 5 it falls from 10.9 s to 6.1 s over the same cars.

4.3. Selecting Anchor Parameters: Policy Learning vs. Black-Box Optimization

The parameter-selection methods of Section 3.7 are compared under the protocol of Section 3.7.4. All methods are evaluated on the same 100 held-out scenarios (seeds 900001 to 900100), so differences are paired. Training, selection and comparison use the same environment reward for all methods:

$$R_{env}(\xi, \mathbf{a}) = R_{phys}(\xi, \boldsymbol{\theta}(\mathbf{a})) - 15\sum_{j=1}^{4} a_j^2 - 75\sum_{j=1}^{4} \max\left(0, \left|a_j\right| - 0.82\right)^2 \quad (14)$$

The second and third terms (centering and barrier) keep the normalized action away from the bounds of the action box ($|a_j| > 0.82$) and are not part of the physical objective; parameter limits for the learner are given in Table A1.

4.3.1. The Weight on Delay Decides the Operating Point

Before comparing methods, Table 6 shows the effects of the delay weight. CMA-ES was run with five weights on the delay term (one run each, held-out seeds).

Table 6. Effect of the delay weight on the CMA-ES solution (Section 4.3 held-out seeds, one run per weight).

Delay Weight	Merges (of 20)	Mean Delay (s)	Worst Delay (s)	Min TTC (s)	RMS Jerk, Main-Road
0.02	19.96	14.75	47.3	1.72	2.19
0.05	19.78	15.03	45.7	1.54	1.92
0.15 (used)	9.74	7.41	21.0	1.44	2.07
0.5	2.98	1.82	5.8	1.54	2.67
1.5	1.95	1.01	3.0	1.72	3.09

With a small weight, the optimizer lets almost all secondary vehicles merge at the cost of a worst-case delay of about 46 s; with a large weight, it does almost nothing. The weight is therefore a policy decision about how much delay the main road should carry, not a tuning detail; the methods below are compared at one weight (0.15, worst-case delay about 20 s), which we do not claim is the right value for a real junction. For weights up to 0.5, the ratio of merges to worst-case delay stays between about 0.42 and 0.51 merges per second, close to the common curve of Section 4.2.2, and rises to 0.65 only at the largest weight, where the worst-case delay is 3 s.

4.3.2. Benchmark Results: Deep RL vs. Evolutionary and Stochastic Search

Table 7 shows the 5 independent TD3 policies (mean held-out environment reward 273.6). The seed-to-seed standard deviation (7.4) is comparable to the scenario-level confidence interval of an individual policy (approx. 5.6), so multi-seed averages are reported rather than single best-case RL runs.

The five runs end in nearly the same place (Figure 5a). The curves flatten once the exploration noise has decayed to its minimum at 60% of training (episode 12,000), and they converge to almost the same value. The problem is a single decision with a broad, smooth optimum whose location hardly depends on the scenario, so every run learns a nearly constant action close to the setting found by CMA-ES and Random Search. This is expected for three reasons. First, each episode is a single decision on four static parameters, so learning is a low-dimensional optimization rather than a sequential control task. Second, the reward landscape is smooth and has one broad maximum in the interior of the parameter

box (Figure 5c), so independent runs cannot settle in distinct local optima. Third, the best parameters vary little between scenarios (Section 4.3.3), so the policy learned by the networks is nearly a constant vector, the same for all seeds. Across seeds, the standard deviation of the mean action is 4 m and 0.6 km/h for the first anchor and 9 m and 0.4 km/h for the second.

Table 7. TD3 results for five training seeds (Section 4.3 held-out seeds).

Training Seed	Environment Reward R_{env}	Secondary Merges	Mean Priority Delay (s)	RMS Jerk (m/s^3)	Minimum TTC (s)	Mean Executed Action (x_5 m, v_5 km/h, x_{15} m, v_{15} km/h)
1	271.1 ± 4.2	9.01	7.04	1.83	1.42	−98, 21.3, −191, 18.1
2	265.3 ± 6.0	9.08	7.11	1.84	1.42	−92, 20.6, −199, 18.4
3	283.1 ± 5.8	8.87	6.82	1.84	1.42	−89, 20.0, −183, 17.9
4	268.9 ± 5.8	8.98	6.98	1.84	1.42	−96, 21.1, −207, 19.1
5	279.4 ± 6.3	8.88	6.85	1.83	1.42	−99, 21.5, −191, 18.4
Mean	273.6 (sd 7.4)	8.96	6.96	1.84	1.42	−95, 20.9, −194, 18.4

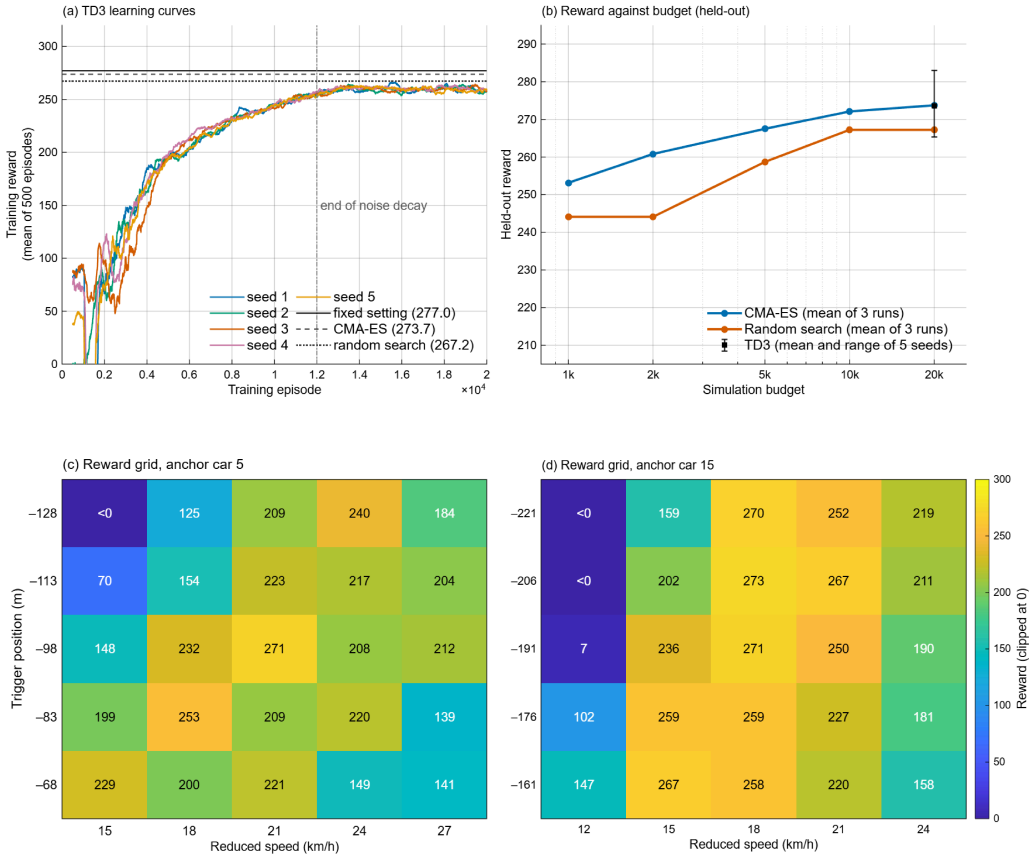


Figure 5. Selection of the anchor parameters. (**a**) Training reward of five independent TD3 runs (random seeds 1–5, identical settings), as a trailing mean over 500 episodes. The early rewards are negative and lie below the axis. The reward includes exploration noise and the centering and barrier terms. Horizontal lines: final held-out reward of the best fixed setting (black solid), CMA-ES (dashed) and Random Search (dotted) at 20,000 simulations (Table 8). Vertical dash-dot line: the exploration noise reaches its minimum (episode 12,000). (**b**) Held-out reward against the simulation budget for CMA-ES and Random Search (mean of three runs) and the final TD3 result (mean and range of five training seeds, 20,000 simulations). (**c**,**d**) Mean reward on a 5 × 5 grid of trigger position and reduced speed for each anchor with the other anchor held at the mean action of the first TD3 agent (seed 1, Table 7) (50 scenarios, seeds 800001 to 800050); the best cells lie in the interior of the grid and many neighboring cells score within about 10% of the best.

Table 8. TD3, CMA-ES and Random Search at equal budget (20,000 simulations; Section 4.3 held-out seeds).

Method/ Algorithm	Independent Runs	Held-Out Reward, Mean (SD) [Range]	Merges	Statistical Outcome
TD3	5	273.6 (7.4) [265,283]	8.96	Reference
CMA-ES	3	273.7 (11.5) [261,281]	9.51	Equivalent (TOST)
Random Search	3	267.2 (4.6) [263,272]	9.11	Equivalent (TOST)

At equal simulation budget (20,000 simulations), TD3 and the derivative-free black-box optimizers are statistically equivalent (Section 3.7.4; Table 8).

The mean paired reward difference is −0.2 between TD3 and CMA-ES and +6.3 between TD3 and Random Search. Their 90% confidence intervals, [−12.6, 12.2] and [−0.8, 13.4], lie within the equivalence margin of ±13.7 (TOST $p = 0.04$ and $p = 0.045$), so the methods are statistically equivalent. A Welch t-test over the independent training runs gives the same conclusion ($p = 0.99$ and $p = 0.18$), although with three to five runs per method, it has low power. Both CMA-ES and Random Search reach the TD3 level within 10,000 simulations (Figure 5b), so Deep RL offers no sample-efficiency advantage here.

The best fixed setting on the 5 × 5 grid of Figure 5c, θ = (−98 m, 21.3 km/h, −206 m, 18.1 km/h), achieves a held-out environment reward of 277.0, slightly above TD3 agent #1 (271.1). The fixed setting is better by 5.9 reward units (90% confidence interval [2.0, 9.8]), a significant difference that lies within the equivalence margin. TD3 thus converges to the vicinity of the best static parameter point rather than uncovering new trade-offs.

4.3.3. Upper-Bound Headroom and Policy Fragility Under Execution Noise

Evaluating the Per-Scenario Oracle—which optimizes parameter vector θ separately for each test scenario—establishes an optimistic upper bound reward of 311.2 (compared to 281.2 for the best constant setting fitted on those scenarios), as shown in Section 5.4 for $\sigma_{\text{noise}} = 0$. This reveals a theoretical headroom of +30.0 ± 5.0 reward units (+9.6%).

The Oracle parameters vary little across scenarios ($x_5 = -90 \pm 11$ m, $v_5 = 20.6 \pm 1.5$ km/h, $x_{15} = -201 \pm 21$ m, $v_{15} = 18.5 \pm 1.2$ km/h) and cannot be predicted from the initial state. A ridge regression on 11 initial-state features (position, speed and spacings of each anchor and of the first secondary vehicles), evaluated by 5-fold cross-validation, gave out-of-fold R^2 of −0.18, −0.16, −0.16 and −0.11, i.e., worse than the mean. This held over 50 random fold splits, for a random forest and in a permutation test ($p \geq 0.13$). A dependence on the initial state is therefore not excluded, but with these features and 100 scenarios, it is too weak to exploit, partly because the flat reward near the optimum leaves the best parameters of a single scenario poorly identified.

4.3.4. Safety and Comfort Assessment

Table 9 compares the TD3 setting with uncontrolled traffic on the held-out test scenarios. The CMA-ES solutions of Table 6 likewise produced no collisions, with minimum TTC between 1.44 and 1.72 s.

Key safety and comfort findings include:

- Ride comfort: anchor controls reduces main-road RMS jerk from 3.20 m/s^3 to 2.06 m/s^3 (−1.14 ± 0.02 m/s^3), as planned deceleration replaces sudden braking reactions.
- Time-to-collision: main-road minimum TTC decreases from 1.89 s to 1.42 s (−0.46 ± 0.02 s); it is below 2.0 s in 92% of uncontrolled runs and 100% of controlled runs, so traffic remains dense under both regimes.
- No collisions occurred in any run or setting; merging safety is governed by the rule-based gap arbiter.

Table 9. Safety and comfort relative to uncontrolled traffic, Section 4.3 held-out seeds 900001 to 900100 (n = 100). Jerk is the RMS jerk of main-road vehicles; TTC is the minimum per scenario, mean (worst case).

Setting	Merges	Mean Delay (s)	Worst Delay (s)	Jerk, Main-Road (m/s^3)	Min TTC (s)	Collisions
Uncontrolled	0.01	0.00	0.00	3.20	1.89 (1.72)	0
TD3 mean setting	8.96	6.96	19.4	2.06	1.42 (1.42)	0

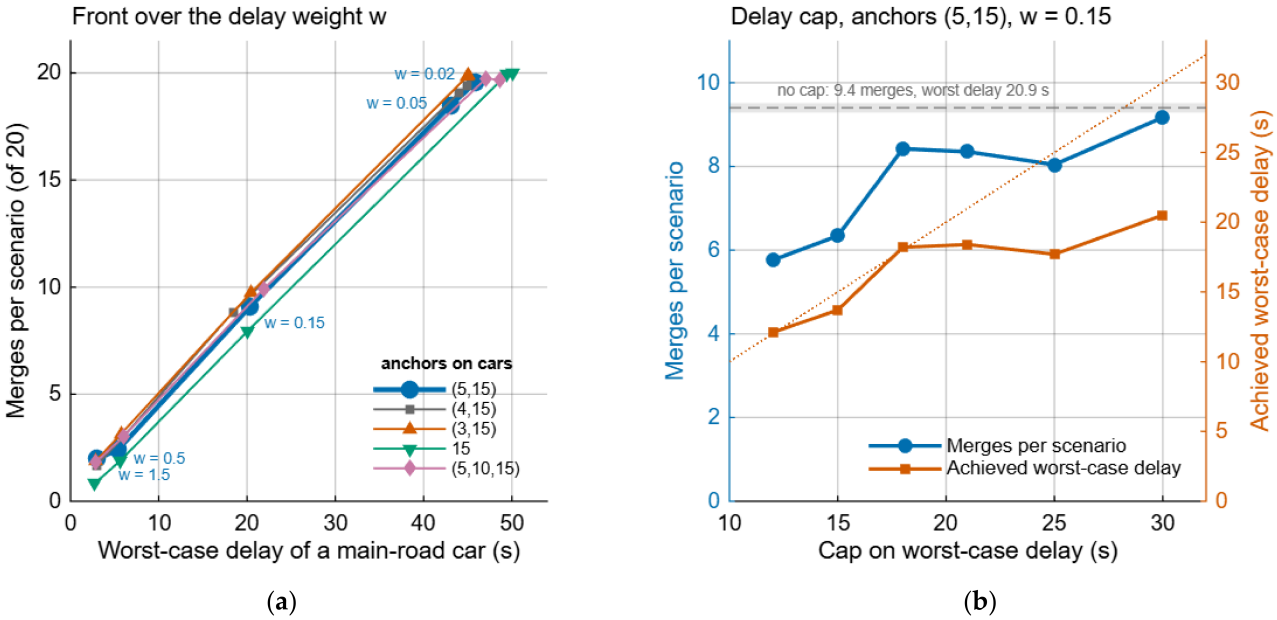
5. Trade-Offs, Fairness Caps and Robustness

5.1. The Throughput–Delay Performance Frontier

The trade-off between side-road throughput and main-road disruption is governed by the delay weight w_d. For the pair (5,15), parameters were optimized for five values of w_d (0.02, 0.05, 0.15, 0.50, 1.50) (Table 10, Figure 6a). Low weights (0.02, 0.05) let 18 to 20 secondary vehicles merge at worst-case delays of 43 to 46 s; the nominal weight 0.15 lies near the knee of the curve and clears about half of the queue at about 20 s; high weights keep the worst-case delay under 5.4 s but permit only 2–3 merges.

Table 10. Throughput-delay front of the pair (5,15) for five delay weights (Section 5 protocol, no cap).

Delay Weight	Merges (of 20)	Mean Delay (s)	Worst Delay (s)
0.02	19.58	17.70	45.9
0.05	18.49	14.78	43.1
0.15	9.07	7.29	20.3
0.5	2.41	1.71	5.4
1.5	1.98	1.01	3.0



(**a**) (**b**)

Figure 6. Trade-off between merges and worst-case scenario delay (Section 4.2 protocol, 100 fresh scenarios, one fit per point). (**a**) Front over the delay weight w_d = 0.02, 0.05, 0.15, 0.5, 1.5 for five anchor configurations; all lie close to one line. (**b**) Effect of a cap on the worst-case delay for anchors (5,15) at w_d = 0.15: merges (left axis) and achieved worst-case delay (right axis, dotted line = cap). Caps of 18 s and above do not bind.

Figure 6a shows the same behavior for the other pairs. The choice of w_d thus acts as a direct policy knob for traffic operators: moving from 20 s to 43 s of peak main-road delay roughly doubles side-road throughput, whereas reducing peak delay below 5 s makes the anchor intervention ineffective.

5.2. Fairness Delay Caps and Equity Bounds

A worst-case delay cap D_{cap} enters the reward through the quadratic penalty term of Equation (11). For the pair (5,15) at $w_d = 0.15$, caps from 12 s to 30 s were tested (Table 11). Caps of 18 s or higher do not bind, as the unconstrained optimization already settles at peak delays of 18.2–20.9 s. Stricter caps restrict the worst-case delay at a roughly linear throughput penalty: a cap of 12 s gives a peak delay of 12.1 s and 5.76 merges, over 60% of the uncapped throughput.

Table 11. Worst-case delay caps for the pair (5,15), delay weight 0.15 (Section 5 protocol, 100 fresh scenarios).

Fairness Cap Threshold (s)	Secondary Merges	Mean Priority Delay (s)	Achieved Peak Delay (s)
12	5.76	5.01	12.1
15	6.34	5.46	13.7
18	8.42	6.58	18.2
21	8.35	6.73	18.4
25	8.04	6.40	17.7
30	9.17	7.35	20.5
none	9.40	7.53	20.9

The linear slope in the binding regime (approx. 0.40 merges per second of delay reduction) closely corresponds to the general efficiency frontier slope (approx. 0.45 merges/s), showing that equity bounds can be enforced explicitly without degrading efficiency.

5.3. Placement Topology and Parameter Sensitivity

To check whether the best anchor placement depends on w_d, the configurations were re-optimized in three independent evolution strategy (ES) runs at weights 0.05, 0.15, and 0.50 (Table 12). Differences among neighboring anchor positions remain small relative to the spread across fits:

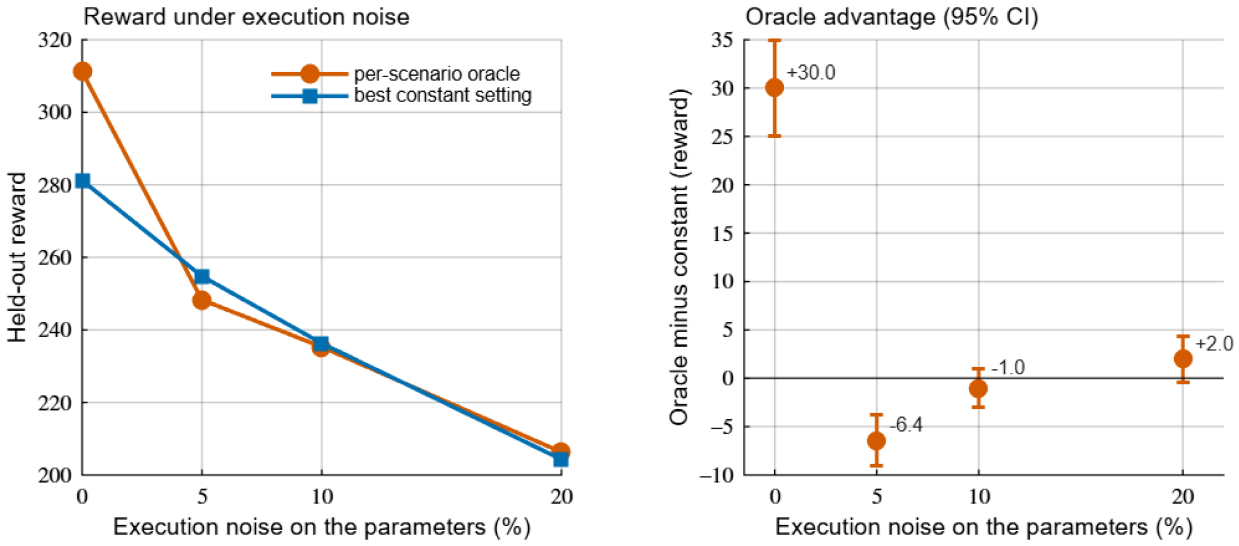
- Positioning the first anchor at car 3 or car 4 provides minor reward variations with respect to car 5, confirming that placing the first anchor anywhere within cars 3–6 produces robust performance.
- The presence of the second anchor at car 15 is the primary driver of performance; single-anchor settings perform significantly worse across all operational delay weights.

Table 12. Reward differences relative to (5,15) across delay weights over 3 independent runs.

Delay Weight w_d	Configuration (4,15) vs. (5,15)	Configuration (3,15) vs. (5,15)
0.05	-2.8 ± 35.7	$+25.7 \pm 25.5$
0.15	$+20.2 \pm 6.9$	$+34.4 \pm 13.0$
0.5	-8.8 ± 7.3	$+14.1 \pm 8.1$

5.4. Robustness to Actuation Noise and Execution Disturbances

To evaluate real-world feasibility, anchor control commands were subjected to multiplicative Gaussian execution noise $\sigma_{noise} \in \{0.00, 0.05, 0.10, 0.20\}$ across 100 held-out test scenarios (Figure 7).



(**a**) (**b**)

Figure 7. Robustness to execution noise (100 scenarios, 20 noise draws). (**a**) Reward of the per-scenario Oracle and of the best constant setting when the executed parameters are perturbed multiplicatively. (**b**) Oracle minus constant with 95% confidence intervals. Orange circles: mean difference, error bars: 95% confidence intervals.

Key robustness observations:

- Under the fixed setting, 5% and 10% execution noise reduce the reward by 9% and 16%, respectively; the gap-generation mechanism remains functional.
- The +9.6% theoretical advantage of a per-scenario Oracle disappears at 5% execution noise, so scenario-specific fine-tuning offers no practical value under realistic sensing or execution uncertainty.

5.5. Sensitivity to the Simulator Assumptions

To verify that findings do not depend on specific simulation constants, we test seven variants of the IDM time headway T (0.6 s in the base case), the gap threshold G_{min} (16 m), and its time margin Δt_m (2 s). For each setting, we compare no anchors, a fixed anchor vector tuned only under the base setting (x = −191 m and 26.1 km/h for car 5, x = −207 m and 18.4 km/h for car 15), and an anchor vector re-tuned for that setting (evolution strategy with 12 candidates and 10 generations on 8 training scenarios), on the 100 fresh evaluation scenarios (Table 13).

As summarized in Table 13, in all seven settings, anchors turn an almost blocked merge (0 to 0.5 merges) into a flowing one (5.5 to 10.3 merges with a fixed vector). The qualitative conclusion does not depend on T, G_{min} or Δt_m within the ranges tested.

- The absolute number of merges does depend on T. Compared with the base, a fixed vector gives 23% fewer merges at T = 1.0 s and 41% fewer at T = 1.4 s. The base value T = 0.6 s is shorter than typical car-following headways, so the merge counts of the main results are probably an upper estimate; the effect of the anchors (a large increase over no anchors) is not.

- The gap threshold G_{min} does not bind in these scenarios: changing it from 16 m to 12 m leaves the fixed-vector results identical (9.32 merges, same delays), and changing it to 20 m changes them by 0.03 merges. The time margin Δt_m is what limits the merges.
- A larger T raises the minimum TTC (2.57 s and 3.27 s), as expected for larger headway.

Table 13. Sensitivity of the results to the simulator assumptions [1]: seven variants of the time headway T, the gap threshold G_{min} and the time margin Δt_m.

Simulator Variant Parameter	Uncontrolled Merges	Merges Under Fixed Setting	Merges Under Re-Tuned Setting	Worst-Case Delay (Fixed/Re-Tuned, s)	Minimum TTC (Fixed Setting, s)
Base ($T = 0.6$, G_{min} 16 m, $\Delta t_m = 2$ s)	0.03	9.32 ± 0.11	9.32 ± 0.11	20.8/20.8	1.42
$T = 1.0$ s	0.00	7.14 ± 0.10	6.28 ± 0.11	19.6/17.1	2.57
$T = 1.4$ s	0.00	5.54 ± 0.10	4.70 ± 0.09	17.5/13.6	3.27
$G = 12$ m	0.03	9.32 ± 0.11	8.86 ± 0.08	20.8/19.4	1.42
$G = 20$ m	0.03	9.29 ± 0.11	9.38 ± 0.12	20.8/21.1	1.42
$\Delta t_m = 1.5$ s	0.49	10.26 ± 0.13	7.98 ± 0.14	20.8/15.3	1.42
$\Delta t_m = 2.5$ s	0.00	8.57 ± 0.10	8.91 ± 0.06	20.8/20.7	1.42

[1] Intervals are 95% confidence intervals over the 100 scenarios.

6. Discussion and Limitations

6.1. The Mechanics of Cooperative Gap Generation

With two automated "anchor" vehicles in a 20-car priority platoon (10% penetration), a virtually blocked merge becomes a flowing one (around 9 of 20 merges per episode), without high CAV penetration or complex multi-agent coordination.

Three features of the gap-generation mechanism explain this result:

- Differential speed propagation. An anchor slowing from 50 km/h to approx. 20 km/h lets the vehicles ahead of it move away while its followers close up behind it, so the gap grows according to first-order kinematics ($\Delta s \approx \Delta v \cdot t_{slow}$).
- The efficiency plateau. Efficiency stays near $\eta \approx 0.45$ merges/s, and shifting anchor positions within $\pm$ 2 vehicles (e.g., cars 4 to 8 and 14 to 17) keeps it within $\pm$ 5%.
- The invariant performance frontier. Anchor placements, delay weights (w_d) and delay caps (D_{cap}) all move the operating point along one linear frontier (Equation (13)); they do not shift the frontier, but let traffic managers select an operating point on it.

The results also question the assumption that such bottlenecks necessitate sequential Deep Reinforcement Learning (RL) policies. At equal budget on 100 held-out test scenarios, TD3 did not differ significantly from CMA-ES, Random Search or an offline-tuned fixed setting (Welch's t-test, $p \geq 0.19$). It was equivalent to the fixed setting within $\pm$5% (TOST, $p = 0.017$). Equivalence with CMA-ES and Random Search held only within a margin of about $\pm$6%, because of the small number of runs (Section 4.3.2). In no case did deep RL perform better than the simpler methods. Three reasons explain this equality:

- Dimensionality of the control task. The decision is made once per episode and is a 4-dimensional static optimization over a smooth reward landscape (Section 4.3.2).
- Negligible scenario variance. The best parameters vary little across scenarios, so TD3 learns a near-constant vector (Section 4.3.2).
- Fragility of fine-tuned Oracles. The +9.6% reward gain of a per-scenario Oracle disappears under 5% execution noise (-6.4 ± 2.6 relative to the fixed setting), because it relies on micro-adjustments that realistic V2X sensing latency or actuator jitter would erase.

6.2. Practical Traffic Engineering and Operational Deployment Strategy

These results have direct operational implications, presented below.

- Selection of the anchors. Anchors need only be near the target positions: efficiency stays within $\pm$5% for shifts of up to two positions (Section 4.2.1). A roadside unit

(RSU) about 300 m upstream, as in [34], would detect the platoon and designate the automated vehicles closest to one quarter and three quarters of its length; an anchor can decline if its follower's safe distance is not ensured [34]. A single anchor still yields 4 to 7 merges (Table 3).

- Determination of the control parameters. Since a fixed setting matches scenario-specific policies, the four parameters can be tuned offline on a calibrated model of the junction, for the delay weight or cap chosen by the operator (Sections 5.1 and 5.2), stored in a look-up table by time of day or flow, and they tolerate 5–10% execution errors (Section 5.4).
- Sensing and communication. The RSU needs only the platoon composition, a stop-line queue detector and a one-way I2V message giving each anchor its trigger position and reduced speed. The anchors must accept external speed advice [32] and localize the trigger within a few meters.
- Limits. The qualitative findings are robust to the variations of Section 5.5, but the absolute numbers are not: they fall by up to 41% with longer headways and may fall further under the simplifications of Section 6.3. Confirmation in a calibrated microsimulation is required before deployment.

6.3. Scope and Further Developments

The method was evaluated in a simplified setting: single-lane approaches meeting at one conflict point, identical IDM drivers, a finite 20-car platoon and two automated vehicles at given positions. Within this setting, it works without coordination of the other vehicles. This makes it a baseline on which several extensions can be built:

1. More realistic traffic. Multi-lane and circulating geometries in microscopic simulators such as SUMO or VISSIM, where lane-changing vehicles may take part of the created gap [32,33], heterogeneous drivers, and car-following models calibrated on trajectory data, including measured ACC and CACC responses for the anchors [38].
2. Variable CAV penetration and positions. Adaptive assignment of the automated vehicles that happen to be in the platoon, at different penetration rates and for symmetric or more heavily loaded junctions.
3. Closed-loop and network-level control. Anchor commands adapted to the detected secondary queue (e.g., by Model Predictive Control), and continuous arrivals with queue spillback and upstream shockwave propagation.

7. Conclusions

This study shows that two automated "anchor" vehicles in a 20-car priority platoon (10% CAV market penetration), slowing down slightly upstream of an unsignalized priority junction, transform a virtually blocked merge into a flowing one.

The main findings are:

- Gap creation under sparse autonomy. Without anchors almost no secondary vehicle merges; with anchors at cars 5 and 15, 8 to 9 vehicles merge per episode at a worst-case priority-stream delay of around 19 s.
- One trade-off curve. Anchor positions, delay weights and delay caps move the operating point along a single, linear performance frontier ($n \approx 0.449\ D_{max}$, $R^2 = 0.984$); positions within $\pm$ 2 vehicles of (5,15) keep the efficiency within $\pm$ 5%.
- No advantage of Deep RL. At equal budget on 100 held-out test scenarios, TD3 (273.6 ± 7.4) did not differ significantly from CMA-ES (273.7 ± 11.5), Random Search (267.2 ± 4.6) or an offline-tuned fixed setting (277.0), and was equivalent to the fixed setting within $\pm$5%.

- Weakness of scenario fine-tuning. The +9.6% advantage of a per-scenario Oracle (311.2 vs. 281.2) vanishes under 5% execution noise (-6.4 ± 2.6 relative to the fixed setting).

For single-decision, low-dimensional gap-generation tasks, simple offline-tuned deceleration profiles therefore offer an effective, robust, and low-complexity alternative to Deep Reinforcement Learning policies or full-fleet V2X coordination, and a practical option for improving mixed-traffic bottlenecks on single-lane approaches in near-term CAV deployment.

Author Contributions: Conceptualization, C.A.; methodology, C.A. and S.B.; software, C.A.; validation, S.B.; formal analysis, S.B.; investigation, S.B.; resources, S.B.; writing—original draft preparation, C.A.; writing—review and editing, S.B.; visualization, S.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: This work was supported by the iDrive Project (https://idriveproject.ro/ accessed on 19 October 2026) through a grant of the Ministry of Research, Innovation and Digitization, CNCS-UEFISCDI, project number PN-IV-P2-2.1-TE-2023-1434, within PNCDI IV. During the preparation of this manuscript, the authors used Claude for Windows, Version 5.11492.5 (Anthropic, San Francisco, CA, USA) for the purposes of language editing, grammar checking, and improving readability. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Adaptive cruise control
AV	Automated vehicle
CACC	Cooperative Adaptive Cruise Control
CAVs	Connected and Automated Vehicles
CMA-ES	Covariance Matrix Adaptation Evolution Strategy
DDPG	Deep Deterministic Policy Gradient
ES	Evolution strategy
HV	Human-driven vehicle
HCM	Highway Capacity Manual
I2V	Infrastructure-to-vehicle
IDM	Intelligent driver model
MLP	Multi-layer perceptron
MPC	Model predictive control
ReLU	Rectified linear unit
RL	Reinforcement learning
RMS	Root-mean-square
RSU	Roadside unit
TD3	Twin Delayed Deep Deterministic Policy Gradient
TOST	Two one-sided tests
TTC/TTC_{min}	Time-to-collision/Minimum time-to-collision
V2X	Vehicle-to-everything communication

Appendix A

The symbols, parameters and ranges used in this study are listed in Table A1.

Table A1. Symbols, parameters and ranges.

Symbol	Meaning	Value or Range
N_p, N_s	main-road and secondary-road vehicles per scenario	20, 20
k	index of an anchor vehicle in the main-road platoon	5 and 15 (placement study: 2 to 19)
$x_{slow,k}$	trigger position of anchor k (conflict point at x = 0)	−300 m to −30 m (placement study); −130 m to −30 m for car 5 and −250 to −50 m for car 15 (learning study)
$v_{slow,k}$	reduced speed of anchor k	10 to 30 km/h (placement and learning studies)
θ	parameter vector ($x_{slow,5}$, $v_{slow,5}$, $x_{slow,15}$, $v_{slow,15}$)	Equation (6)
ξ	scenario: initial positions and speeds of the 40 vehicles	Section 3.4
n	secondary vehicles that complete the merge	0 to 20
d_i, D_{max}	delay of main-road vehicle *i* against the uncontrolled run; worst case over *i*	s
J	RMS jerk of the main-road vehicles	m/s^3
E	braking energy of the main-road vehicles	J/kg
TTC_{min}	minimum time-to-collision over car-following pairs	s
η	efficiency: mean merges divided by mean worst-case delay	Merges/s
R	reward, Equation (11)	scale depends on protocol
w_n, exponent	throughput weight and exponent	60, 1.1
w_d	delay weight	$0.15/s^2$ (sweep: 0.02, 0.05, 0.15, 0.5, 1.5)
w_J, w_E	jerk and energy weights	$1/(m/s^3)^2$, 0.05 kg/J
w_f, D_{cap}	fairness penalty weight and worst-case delay cap	$20/s^2$; 21 s unless stated otherwise
a	normalized action of the learner	in $[-1, 1]^4$
L	vehicle length (all vehicles)	4.5 m
G, G_{min}	free gap (Equation (2)), recomputed at every time step; acceptance threshold of the arbiter	m; G_{min} = 16 m
T, Δt_m	IDM time headway; arbiter time margin	0.6 s, 2 s (Section 5.5 varies them)
t_u, t_{entry}	time of the upstream main-road vehicle and of the merging vehicle to reach the conflict point	s, Equations (3) and (4)

References

1. National Academies of Sciences, Engineering, and Medicine. Roundabouts. In *Highway Capacity Manual 7th Edition: A Guide for Multimodal Mobility Analysis*; Interrupted Flow; Chapter 22; The National Academies Press: Washington, DC, USA, 2022; Volume 3. [CrossRef] [PubMed]
2. Zhou, H.; Ivan, J.N.; Garder, P.E.; Ravishanker, N. Gap Acceptance for Left Turns from the Major Road at Unsignalized Intersections. *Transport* **2017**, *32*, 252–261. [CrossRef]

3. Campi, E.; Mastinu, G.R.M.; Previati, G.; Studer, L.; Uccello, L. Roundabouts: Traffic Simulations of Connected and Automated Vehicles—A State of the Art. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 3305–3325. [CrossRef]
4. Antonya, C.; Iclodean, C.; Roșu, I.-A. Enhancing Roundabout Efficiency Through Autonomous Vehicle Coordination. *Vehicles* **2025**, *7*, 9. [CrossRef]
5. Rios-Torres, J.; Malikopoulos, A.A. A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 1066–1077. [CrossRef]
6. Rios-Torres, J.; Malikopoulos, A.A. Automated and Cooperative Vehicle Merging at Highway On-Ramps. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 780–789. [CrossRef]
7. Ding, J.; Peng, H.; Zhang, Y.; Li, L. Penetration Effect of Connected and Automated Vehicles on Cooperative On-ramp Merging. *IET Intell. Transp. Syst.* **2020**, *14*, 56–64. [CrossRef]
8. Dresner, K.; Stone, P. A Multiagent Approach to Autonomous Intersection Management. *J. Artif. Intell. Res.* **2008**, *31*, 591–656. [CrossRef]
9. Au, T.-C.; Shahidi, N.; Stone, P. Enforcing Liveness in Autonomous Traffic Management. *Proc. AAAI Conf. Artif. Intell.* **2011**, *25*, 1317–1322. [CrossRef]
10. Zhao, L.; Malikopoulos, A.A.; Rios-Torres, J. Optimal Control of Connected and Automated Vehicles at Roundabouts: An Investigation in a Mixed-Traffic Environment. *IFAC-PapersOnLine* **2018**, *51*, 73–78. [CrossRef]
11. Bakibillah, A.S.M.; Kamal, M.A.S.; Tan, C.P.; Susilawati, S.; Hayakawa, T.; Imura, J. Bi-Level Coordinated Merging of Connected and Automated Vehicles at Roundabouts. *Sensors* **2021**, *21*, 6533. [CrossRef] [PubMed]
12. Wang, C.; Wang, Y.; Peeta, S. Cooperative Roundabout Control Strategy for Connected and Autonomous Vehicles. *Appl. Sci.* **2022**, *12*, 12678. [CrossRef]
13. Wu, Y.; Zhu, F. Junction Management for Connected and Automated Vehicles: Intersection or Roundabout? *Sustainability* **2021**, *13*, 9482. [CrossRef]
14. Jiang, Q.; Schroeder, B.J.; Ma, J.; Rodegerdts, L.A.; Cesme, B.; Bibeka, A.; Morgan, A. Developing Highway Capacity Manual Capacity Adjustment Factors for Connected and Automated Traffic on Roundabouts. *J. Transp. Eng. Part A Syst.* **2022**, *148*. [CrossRef]
15. Boualam, O.; Borsos, A.; Koren, C.; Nagy, V. Impact of Autonomous Vehicles on Roundabout Capacity. *Sustainability* **2022**, *14*, 2203. [CrossRef]
16. Stern, R.; Cui, S.; Delle Monache, M.L.; Bhadani, R.K.; Bunting, M.; Churchill, M.; Hamilton, N.P.; Haulcy, R.; Pohlmann, H.; Wu, F.; et al. Dissipation of Stop-and-Go Waves via Control of Autonomous Vehicles: Field Experiments. *Transp. Res. Part C Emerg. Technol.* **2018**, *89*, 205–221. [CrossRef]
17. Zheng, Y.; Wang, J.; Li, K. Smoothing Traffic Flow via Control of Autonomous Vehicles. *IEEE Internet Things J.* **2020**, *7*, 3882–3896. [CrossRef]
18. Wang, J.; Zheng, Y.; Dong, J.; Chen, C.; Cai, M.; Li, K.; Xu, Q. Implementation and Experimental Validation of Data-Driven Predictive Control for Dissipating Stop-and-Go Waves in Mixed Traffic. *IEEE Internet Things J.* **2024**, *11*, 4570–4585. [CrossRef]
19. Wu, C.; Kreidieh, A.R.; Parvate, K.; Vinitsky, E.; Bayen, A.M. Flow: A Modular Learning Framework for Mixed Autonomy Traffic. *IEEE Trans. Robot.* **2022**, *38*, 1270–1286. [CrossRef]
20. Kherroubi, Z.E.A.; Aknine, S.; Bacha, R. Novel Decision-Making Strategy for Connected and Autonomous Vehicles in Highway On-Ramp Merging. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 12490–12502. [CrossRef]
21. Nakka, S.K.S.; Chalaki, B.; Malikopoulos, A.A. A Multi-Agent Deep Reinforcement Learning Coordination Framework for Connected and Automated Vehicles at Merging Roadways. In Proceedings of the American Control Conference, Atlanta, GA, USA, 8–10 June 2022; pp. 3297–3302.
22. Zhou, S.; Zhuang, W.; Yin, G.; Liu, H.; Qiu, C. Cooperative On-Ramp Merging Control of Connected and Automated Vehicles: Distributed Multi-Agent Deep Reinforcement Learning Approach. In Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022; pp. 402–408.
23. Lin, Y.S.; McPhee, J.; Azad, N.L. Anti-Jerk On-Ramp Merging Using Deep Reinforcement Learning. *arXiv* **2019**, arXiv:1909.12967. [CrossRef]
24. Ren, T.; Xie, Y.; Jiang, L. Cooperative Highway Work Zone Merge Control Based on Reinforcement Learning in a Connected and Automated Environment. *Transp. Res. Rec. J. Transp. Res. Board* **2020**, *2674*, 363–374. [CrossRef]
25. Hansen, N.; Ostermeier, A. Completely Derandomized Self-Adaptation in Evolution Strategies. *Evol. Comput.* **2001**, *9*, 159–195. [CrossRef] [PubMed]
26. Salimans, T.; Ho, J.; Chen, X.; Sidor, S.; Sutskever, I. Evolution Strategies as a Scalable Alternative to Reinforcement Learning. *arXiv* **2017**, arXiv:1703.03864. [CrossRef]
27. Mania, H.; Guy, A.; Recht, B. Simple Random Search Provides a Competitive Approach to Reinforcement Learning. *arXiv* **2018**, arXiv:1803.07055. [CrossRef]

28. Henderson, P.; Islam, R.; Bachman, P.; Pineau, J.; Precup, D.; Meger, D. Deep Reinforcement Learning That Matters. *Proc. AAAI Conf. Artif. Intell.* **2018**, *32*. [CrossRef]
29. Hang, P.; Lv, C.; Huang, C.; Cai, Y.; Hu, Z.; Xing, Y. Cooperative Decision Making of Connected and Automated Vehicles at Multi-Lane Merging Zone: A Coalitional Game Approach. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 3829–3841. [CrossRef]
30. Scholte, W.J.; Zegelaar, P.W.A.; Nijmeijer, H. A Control Strategy for Merging a Single Vehicle into a Platoon at Highway On-Ramps. *Transp. Res. Part C Emerg. Technol.* **2022**, *136*, 103511. [CrossRef]
31. Giuffrè, O.; Granà, A.; Tumminello, M.L. Gap-Acceptance Parameters for Roundabouts: A Systematic Review. *Eur. Transp. Res. Rev.* **2016**, *8*, 2. [CrossRef]
32. Scarinci, R.; Hegyi, A.; Heydecker, B. Definition of a Merging Assistant Strategy Using Intelligent Vehicles. *Transp. Res. Part C Emerg. Technol.* **2017**, *82*, 161–179. [CrossRef]
33. Chen, D.; Srivastava, A.; Ahn, S. Harnessing Connected and Automated Vehicle Technologies to Control Lane Changes at Freeway Merge Bottlenecks in Mixed Traffic. *Transp. Res. Part C Emerg. Technol.* **2021**, *123*, 102950. [CrossRef]
34. Alanazi, F.; Yi, P.; El Gehawi, N. Improving the Performance of Unsignalized T-Intersections within CAVs Mixed Traffic. *J. Appl. Eng. Sci.* **2022**, *20*, 464–476. [CrossRef]
35. Treiber, M.; Hennecke, A.; Helbing, D. Congested Traffic States in Empirical Observations and Microscopic Simulations. *Phys. Rev. E Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.* **2000**, *62*, 1805–1824. [CrossRef] [PubMed]
36. Fujimoto, S.; van Hoof, H.; Meger, D. Addressing Function Approximation Error in Actor-Critic Methods. *arXiv* **2018**, arXiv:1802.09477. [CrossRef]
37. Majid, A.Y.; Saaybi, S.; François-Lavet, V.; Prasad, R.R.V.; Verhoeven, C.J.M. Deep Reinforcement Learning Versus Evolution Strategies: A Comparative Survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 11939–11957. [CrossRef] [PubMed]
38. Milanés, V.; Shladover, S.E. Modeling Cooperative and Autonomous Adaptive Cruise Control Dynamic Responses Using Experimental Data. *Transp. Res. Part C Emerg. Technol.* **2014**, *48*, 285–300. [CrossRef]
39. Transportation Research Board, U.S. Department of Transportation. *Roundabouts: An Informational Guide*; Transportation Research Board, U.S. Department of Transportation: Washington, DC, USA, 2010.
40. Minderhoud, M.M.; Bovy, P.H. Extended Time-to-Collision Measures for Road Traffic Safety Assessment. *Accid. Anal. Prev.* **2001**, *33*, 89–97. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.