

## Article

# Predictive Risk-Aware Reinforcement Learning for Autonomous Vehicles Using Safety Potential

Jinho Choi <sup>1,2</sup> and Shiho Kim <sup>1,2,\*</sup> <sup>1</sup> Seamless Trans-X Lab (STL), School of Integrated Technology, Yonsei University, Incheon 21983, Republic of Korea; cjh7529@yonsei.ac.kr<sup>2</sup> BK21 Graduate Program in Intelligent Semiconductor Technology, Yonsei University, Incheon 21983, Republic of Korea

\* Correspondence: shiho@yonsei.ac.kr

## Abstract

Safety remains a central challenge in autonomous driving: overly rigid safeguards can cause unnecessary stops and erode efficiency. Addressing this safety–efficiency trade-off requires specifying what behaviors to incentivize. In reinforcement learning, the reward provides that specification. Conventional reward surrogates—such as distance gaps and time-to-collision (TTC)—depend on instantaneous geometry and often miss unfolding multi-vehicle interactions, whereas sparse terminal rewards provide no intermediate guidance. Accordingly, we adapt Safety Potential (SP)—a short-horizon, time-weighted path-overlap forecast—into a dense reward-shaping term that provides a predictive risk-aware signal for anticipatory throttle/brake control. In the CARLA v0.9.14 roundabout environment, SP attains 94% success with 3% collisions; in percentage points, this is 16.00, 13.00, and 5.75 higher success and 18.75, 9.50, and 7.25 lower collisions than No-Safe, Distance, and TTC, respectively. Adding a lightweight reactive guard at inference further reduces collisions to below 1% without sacrificing success. These results indicate that injecting a predictive, overlap-based risk measure directly into the reward supplies temporally consistent safety cues and clarifies the trade-off between progress and risk in reinforcement-learning-based vehicle control.

**Keywords:** autonomous driving; risk-aware reinforcement learning; reward shaping; predictive risk; safety potential; collision avoidance; longitudinal control; anticipatory control; safety-critical decision-making



Academic Editors: Salvador Alepuz, Sergio Busquets-Monge, Daniel Gutiérrez Reina and Joan Nicolás-Apruzzese

Received: 17 September 2025

Revised: 11 November 2025

Accepted: 12 November 2025

Published: 14 November 2025

**Citation:** Choi, J.; Kim, S. Predictive Risk-Aware Reinforcement Learning for Autonomous Vehicles Using Safety Potential. *Electronics* **2025**, *14*, 4446. <https://doi.org/10.3390/electronics14224446>

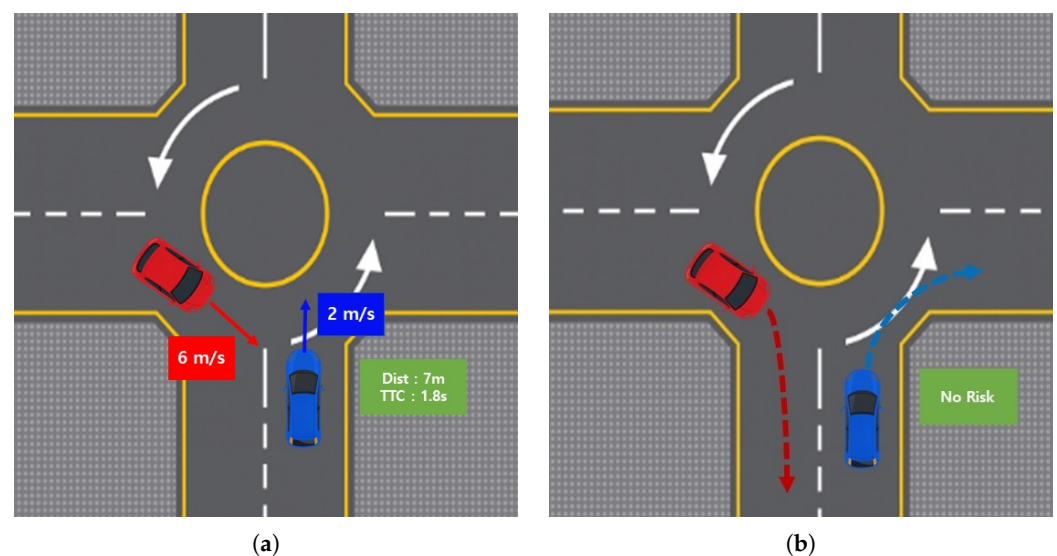
**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Reinforcement learning (RL) is increasingly used in autonomous driving, yet much of the literature emphasizes algorithmic sophistication while reward design remains comparatively underexplored [1,2]. In safety-critical road settings, the reward is not a peripheral detail; it encodes what the agent should value and therefore governs the balance between efficiency and safety. This work adopts a reward-centric stance. Rather than modifying the learning algorithm, we study how shaping the reward can deliver safer behavior without adding architectural complexity.

Most existing RL rewards for on-road driving use reactive risk surrogates constructed from instantaneous geometry, for example distance or gap penalties [3] and time-to-collision (TTC) thresholds [4]. Because these quantities are computed from a single snapshot, they frequently raise false alarms when vehicles are close but on diverging paths. Prior work

has explored proactive warning and predictive risk estimation [5–7], but these signals have seldom been used directly as step-wise reward terms within the RL training loop. Figure 1 illustrates that, even in visually similar traffic scenes, reactive gap/TTC surrogates and a predictive path-overlap signal can yield very different risk assessments. Moreover, works that introduce predictive cues typically compare only against a vanilla baseline, without disentangling how the new reward differs from conventional distance/TTC shaping; thus, the specific contribution of reward design remains unclear [8–10]. We introduce a predictive path-overlap reward that estimates short-horizon overlap between the ego’s planned trajectory and surrounding vehicles, yielding a temporally consistent risk signal that encourages anticipatory throttle and brake. We then conduct matched, reward-to-reward comparisons against TTC and the gap to isolate the contribution of reward design.



**Figure 1.** Reactive vs. predictive risk. (a) Snapshot-based distance/TTC flags risk whenever the instantaneous gap or TTC falls below a threshold—regardless of future motion—so close-but-diverging cases can raise false alarms. (b) Short-horizon path forecasts show no intersection; a path-overlap metric therefore assigns low risk even at small distance or high approach speed, filtering out false warnings.

To isolate the effect of reward shaping, the rest of the stack is deliberately simple: a standard off-policy RL algorithm (Soft Actor-Critic) [11] handles longitudinal speed control and a rule-based Pure Pursuit controller [12] provides lateral steering. Experiments are conducted in the unsignalized CARLA [13] roundabout with a single learning agent (ego) interacting with many surrounding vehicles (NPCs; non-player-controlled vehicles, used here interchangeably with “surrounding vehicles”). Although many works measure or signal risk, comparatively few feed either predictive or reactive risk estimators back into RL as rewards or auxiliary signals and evaluate them on concrete simulator tasks with side-by-side baselines. Our study addresses this gap by operationalizing a lightweight overlap forecast as a step-wise, dense reward and comparing it against conventional snapshot-based rewards in this roundabout setting, showing that careful reward shaping can improve safety without additional model complexity.

### 1.1. Literature Review

A longstanding difficulty in reinforcement learning is the credit assignment problem (CAP): when rewards are sparse, delayed, or weakly informative, attributing credit or blame to specific decisions becomes unreliable and destabilizes learning [14]. Reward shaping was proposed precisely to mitigate CAP by providing additional, task-relevant

feedback while preserving the original objective, and early studies showed that carefully designed shaping signals can accelerate learning and improve robustness across control domains [15,16].

In on-road driving, most shaping terms have been reactive and snapshot-based. Nagesh Rao et al. [17] penalize small headways when the ego-lead gap falls below a safety margin, whereas Huang et al. [18] grant positive reward within a desired spacing band. TTC has likewise been used directly as a safety reward: Zhu et al. [4] employ TTC as a dense term for velocity control, but they do not isolate their effect against alternative safety rewards under identical training conditions.

Beyond reactive measures, predictive approaches have been explored: Lv et al. [8] (lane change) add a short-horizon binary predictor to TTC/gap; Wang et al. [9] (roundabout) train a discrete IP-SAC with a binary penalty on predicted-path intrusion, reporting up to 90% SR and 10% CR; Dong and Guo [10] (intersection) use a transformer-based risk score as reward. These studies report setting-specific gains but rely on binary or black-box signals, do not isolate improvements over TTC/gap under matched conditions, and provide limited interpretability and fine-grained risk discrimination.

NVIDIA [19] introduced the Safety Force Field (SFF) and the notion of safety potential, where SP denotes the degree of overlap among the unions of vehicles' predicted trajectory sets when each agent follows prescribed safety procedures; SFF is the mechanism that seeks to reduce this overlap. The public exposition is largely conceptual and oriented toward globally coordinated control rather than an ego-centric learner. Closer to our setting, Suk et al. [20] train an SP predictor and then modulate longitudinal acceleration with a rule-based controller in CARLA, reporting increases in number of arrivals and longer accident-free time; the decision policy itself is not learned via RL.

Recent task-specific autonomous-driving reinforcement learning work includes the following. At intersections, Leng et al. [21] treat safety as a cost and project the chosen action onto a safe set when the expected collision or lane-departure cost exceeds a threshold. Yu et al. [22] estimate uncertainty over the return with an ensemble or a distributional critic and trigger conservative fallback when uncertainty is high. In roundabouts, Gan et al. [23] train a single PPO policy with a dynamically weighted reward and a compact, state-compressed representation, reporting a 35–80% success rate, while Lin et al. [24] combine a new Q-value estimation approach for discrete decisions with a TTC-based constraint and a tracking controller, citing a 1–2% collision rate. Overall, across recent autonomous-driving RL, most methods rely on conventional risk measures and focus on acting on them rather than on designing them; when new rewards are proposed, matched reward-to-reward comparisons under identical conditions are rare, leaving the contribution of reward design underexplored.

## 1.2. Contributions

We introduce Safety Potential for reward shaping in a standard off-policy SAC agent. In the unsignalized CARLA roundabout, SP outperforms the strongest baseline (TTC), showing a 5.75 percentage-point gain in success and a 7.25 percentage-point reduction in collisions. To isolate reward effects, we compare rewards head-to-head under matched architectures and budgets; performance improves via a simple predictive overlap signal that quantifies risk continuously, is readily extensible, and can also serve as a standalone risk metric or auxiliary signal.

The remainder of the paper is organized as follows. Section 2 describes the methods—predictive risk shaping, reactive baselines, and the experimental setup—and specifies the environment characteristics and task difficulty. Section 3 presents results and analysis: quantitative metrics with statistical comparisons and qualitative case studies, and includes

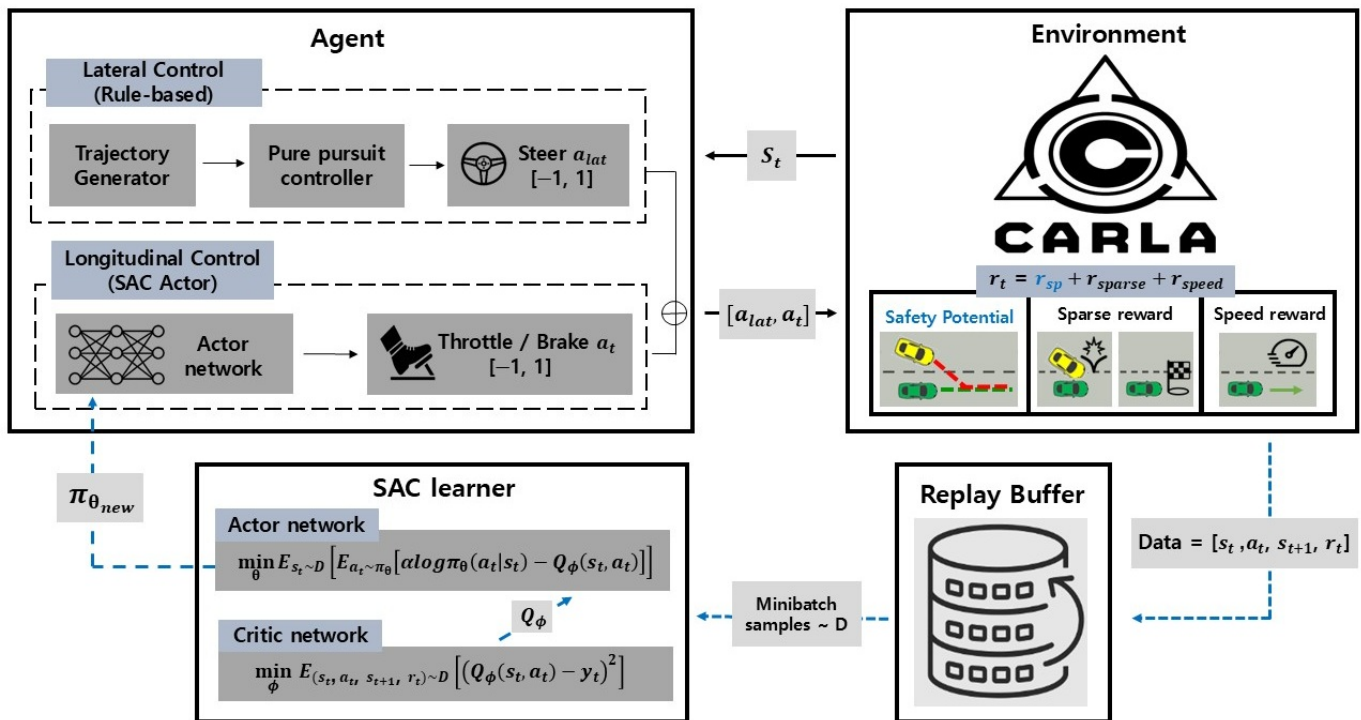
ablations that (i) toggle a simple hard guard and (ii) add Safety Potential as a state feature to separate reward effects from observation design. Section 4 concludes with limitations and directions for future work.

## 2. Method

This section presents a predictive risk-aware reinforcement-learning agent that augments the reward with an SP term while keeping the rest of the pipeline standard and off-policy. We first outline the overall pipeline, then define the fixed 49-dimensional ego-centric state and the action space, followed by the reward design (sparse and speed terms together with the SP shaping term) and the computation of SP from short-horizon path overlap with time weights. Finally, we describe the SAC training procedure with replay-based mini-batch updates and the hyperparameters used across experiments.

### 2.1. Overview

We implement a CARLA agent with rule-based lateral steering (Pure Pursuit) and SAC for longitudinal control. At each step, the agent observes the state and outputs both steering and throttle/brake; the simulator applies the combined control, but only the longitudinal component is used for learning. The resulting transition—state, longitudinal action, reward (including SP shaping), and next state—is logged to a replay buffer. During off-policy training, the learner samples minibatches from the buffer: the critic estimates Q-values from state–action pairs, and the actor is updated to favor actions with higher critic value. An overview of this control and learning framework is shown in Figure 2.



**Figure 2.** Overall framework of the proposed agent. The agent follows a standard off-policy pipeline: at each step, it logs a transition  $(s_t, a_t, r_t, s_{t+1})$  to a replay buffer, and mini-batches from the buffer update the critic and actor. The blue dashed line indicates how SP flows: it shapes the reward  $r_t$ , is stored with each transition, and propagates into learning via the critic target and the policy objective. Black solid arrows denote the interaction between the agent and the CARLA simulator.

## 2.2. State Representation

The agent observes a 49-dimensional ego-centric state  $s \in \mathbb{R}^{49}$  composed of ego features and up to six surrounding-vehicle slots; when fewer than six are present, the unused slots are zero-padded so the vector length remains 49 regardless of traffic count. To mitigate training instability from heterogeneous feature scales, all scalar state variables are normalized: distances and speeds are divided by 20, accelerations by 10, angular terms use a compact encoding in  $[-1, 1]$ , and values are clipped to bounded ranges.

$$s = [s_{\text{ego}} ; s_{\text{prog}} ; s_{\text{look}} ; s_{\text{sur}}], \quad \text{total dimension} = 49 \quad (1)$$

$$s_{\text{ego}} = [v, \theta_v, a, \theta_a], \quad s_{\text{prog}} = [g], \quad s_{\text{look}} = [d_\ell, \theta_\ell] \quad (2)$$

$$s_{\text{sur}} = [s_1, \dots, s_6], \quad s_k = [d_k, \theta_{d,k}, v_k, \theta_{v,k}, \theta_{\text{yaw},k}, a_k, \theta_{a,k}] \quad (3)$$

The overall state dimensionality is 49 (4 for ego, 1 for progress, 2 for lookahead, and  $6 \times 7$  for surrounding vehicles).  $g \in [0, 1]$  denotes the goal–progress ratio (0 at start, 1 at goal), and  $(d_\ell, \theta_\ell)$  is the lookahead target derived from the local route. See Appendix A for complete nomenclature and normalization details.

## 2.3. Action and Control

At each step, the actor network receives the current observation and outputs a single continuous longitudinal command  $a \in [-1, 1]$ . We interpret positive  $a$  as throttle and negative  $a$  as brake:

$$\text{throttle} = \max(a, 0), \quad \text{brake} = \max(-a, 0). \quad (4)$$

Steering is generated by a Pure Pursuit controller rather than the learned policy. We follow a fixed global route and, at each step, construct a speed-adaptive local path from the ego’s nearest point. The Pure Pursuit lookahead point is selected on the constructed local route at 5 m ahead of the ego, from which the steering angle is computed. The simulator then applies the neural policy’s throttle/brake together with this steering command.

## 2.4. Safety Potential: Discretized, Time-Weighted Overlap Risk

To keep the two-dimensional overlap computation tractable while preserving spatial fidelity, both the ego route (generated after the agent action) and each surrounding-vehicle continuation are discretized into short longitudinal rectangular cells; the per-frame Safety Potential is then derived from weighted cell intersections.

### 2.4.1. Route Discretization and Footprinting for SP

For each surrounding vehicle, we generate a short-horizon rule-based prediction by combining two simple elements: (i) a few body-aligned points sampled along the vehicle’s yaw to cover the immediate footprint and (ii) lane-following waypoints smoothly extended ahead from the current position and heading. Waypoints are sampled at a fixed 1 m spacing. Along this route, each NPC yields  $\max(2v + 5, 10)$  rectangular polygons (including the body footprint), while the ego route yields  $\max(2v, 8)$  polygons, where  $v$  is the vehicle speed (m/s) under the 1 m discretization. These deterministic, rule-based routes provide the waypoint sequences, which are converted to rectangular cells for overlap calculation.

### 2.4.2. Overlap Area Definition

Each waypoint is converted to a rectangular polygon oriented with the local path; the width is  $W = 2.8$  m (vehicle width) and the length is  $L = 1$  m. Denote the ego polygons by  $p_i$  ( $i = 0, \dots, N_e - 1$ ) and the  $k$ -th surrounding vehicle's polygons by  $q_{k,j}$  ( $j = 0, \dots, N_o - 1$ ). The rectangular intersection area is written  $\text{Area}(p_i \cap q_{k,j})$ .

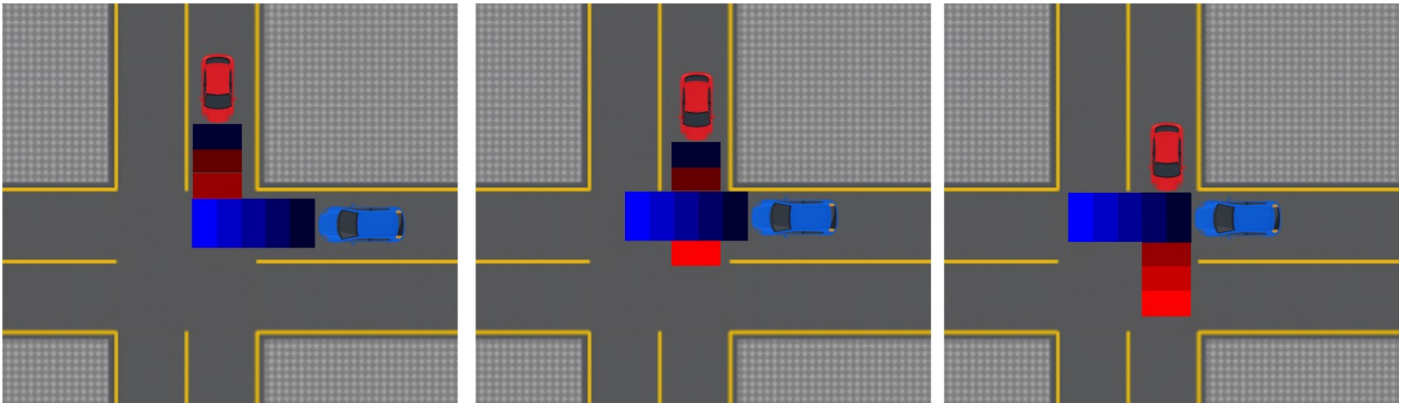
### 2.4.3. Normalized Weights (Time Weight)

To emphasize more imminent interactions, we apply exponential time weights along forward polygon indices and treat a small number of leading NPC cells as unit-weight body boxes, so that near-term overlaps are penalized more strongly, as illustrated in Figure 3.

$$u_{e,i} = \frac{i}{N_e - 1}, \quad u_{o,k,j} = \begin{cases} 0, & j < J_{\text{body}}, \\ \frac{j - J_{\text{body}}}{N_o - 1 - J_{\text{body}}}, & j \geq J_{\text{body}}, \end{cases} \quad (5)$$

$$w_i = \exp(-\eta u_{e,i}), \quad v_{k,j} = \begin{cases} 1, & j < J_{\text{body}}, \\ \exp(-\eta u_{o,k,j}), & j \geq J_{\text{body}}, \end{cases} \quad (6)$$

where  $J_{\text{body}}$  denotes the number of initial body-aligned cells and  $\eta > 0$  is the decay coefficient along the path. We set  $J_{\text{body}} = 5$  to approximately span one vehicle length, and fix  $\eta = 0.5$  since larger  $\eta$  more aggressively down-weights distant cells and may underestimate emerging risk.



**Figure 3.** Illustration of time-weighted overlap risk in a cross-traffic scenario. Red cells denote the predicted path of the NPC vehicle, and blue cells denote the ego vehicle's planned path; darker shades indicate larger time weights (more imminent cells). All three panels involve roughly two overlapped cells at the intersection, but the risk differs: **(left)** far-future overlap yields the lowest SP, **(middle)** mid-range overlap yields moderate SP, and **(right)** imminent overlap yields the highest SP.

### 2.4.4. Safety Potential Aggregate

The Safety Potential at time  $t$  is the maximum, over surrounding vehicles  $k$ , of the time-weighted overlap summed across ego and other-vehicle polygons:

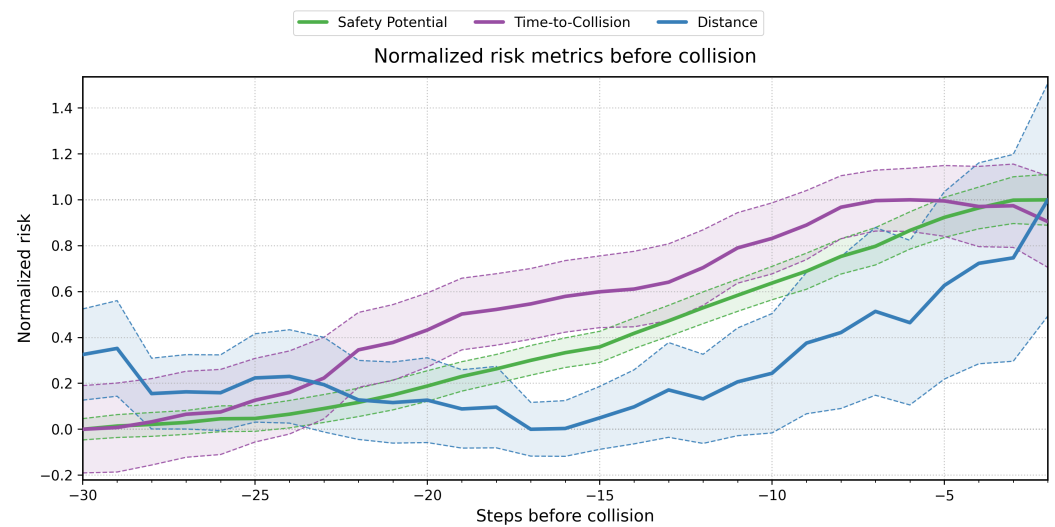
$$\text{SP}_t = \max_k \sum_{i=0}^{N_e-1} \sum_{j=0}^{N_o-1} \min(w_i, v_{k,j}) \text{Area}(p_i \cap q_{k,j}). \quad (7)$$

### 2.4.5. Comparison of Pre-Collision Risk Metrics

We generated collision episodes using a random policy and, for each episode, recorded Safety Potential (SP), Time-to-Collision (TTC), and forward distance at relative timesteps  $\tau = 30, \dots, 2$  before collision (one step = 0.1 s; total = 2.9 s). For visualization, we computed

the per-step average across episodes and estimated 95% confidence intervals via bootstrap across episodes. To place the proxies on a comparable scale, TTC and distance were sign-flipped so that larger values indicate higher risk, and each metric was then normalized to the  $[0, 1]$  range by min–max scaling.

In the episode-aligned averages shown in Figure 4, SP rises approximately monotonically as collision approaches and shows markedly smaller variability across episodes. By contrast, TTC and distance often show nonmonotonic trends, with dips or plateaus in the middle of the trajectory, and they do not consistently peak at collision. This indicates that SP supplies a denser and more temporally consistent signal before collision, whereas TTC and distance are susceptible to short-term fluctuations caused by relative geometry and motion. As a result, they are less suitable as dense shaping signals for learning.



**Figure 4.** Pre-collision risk proxies over the last 29 steps (2.9 s  $\rightarrow$  0.1 s before collision; 10 Hz). Curves show per-step means across episodes for SP, TTC, and distance, each independently min–max normalized (larger indicates higher risk); green, purple, and blue solid lines correspond to SP, TTC, and Distance, respectively, and dotted lines indicate 95% confidence intervals.

## 2.5. Reward Design

The episodic reward combines sparse terminal/event signals that determine success or failure with dense shaping terms that guide safe, efficient, stepwise behavior. Sparse terms dominate final returns, while dense terms (safety and speed shaping) help the agent trade efficiency against risk during training.

### 2.5.1. Safety Potential Reward

Safety Potential is a predictive, overlap-based risk measure. The SP penalty includes a state cost and an action-scaled component that activates above a threshold; braking (negative longitudinal action) reduces the action-scaled penalty, which prevents persistent penalty accumulation in congested or stopped situations:

$$r_{sp}(t) = -\lambda_{sp} SP_t - \lambda_{sp,a} 1_{\{SP_t > \sigma_{sp}\}} a_t SP_t. \quad (8)$$

We use  $\sigma_{sp} = 7$  and  $\lambda_{sp} = \lambda_{sp,a} = 0.005$ . Here  $a_t \in [-1, 1]$  is the longitudinal control.

### 2.5.2. Sparse Event Rewards

Terminal outcomes and checkpoint events are modeled as a single sparse component:

$$r_{\text{sparse}} = \begin{cases} -10, & \text{on collision,} \\ +5, & \text{on goal reached,} \\ +1, & \text{on predefined intermediate waypoint,} \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Here,  $r_{\text{sparse}}$  aggregates both terminal events and non-terminal checkpoints.

### 2.5.3. Speed Reward

Speed shaping encourages maintaining a target speed  $v^* = 8$  m/s but is gated by SP to avoid incentivizing unsafe acceleration into predicted overlaps. The shaping rewards small deviations from  $v^*$  under low risk, penalizes large overspeeding, and applies a small negative term to discourage persistent stalling when the vehicle remains idle without braking:

$$r_{\text{speed}} = \begin{cases} 0.02 \max\left(0, 1 - \frac{|v_t - v^*|}{v^*}\right), & \text{if } v_t \leq v^* + 2, \text{ SP}_t < 3, \\ -0.2 \frac{v_t - v^*}{v^*}, & \text{if } v_t > v^* + 2, \\ -0.015, & \text{if } \text{SP}_t < 0.01, v_t < 0.5, a_t < 0.1, \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

All dense shaping weights are intentionally small relative to terminal rewards so that collisions and goal attainment determine final returns, while SP gating ensures the speed objective does not distort behavior under risky conditions.

### 2.5.4. Total Reward

The per-step reward is the sum of dense shaping and terminal/event terms (the terminal term is nonzero only at the corresponding events):

$$r_t = r_{\text{sp}}(t) + r_{\text{sparse}}(t) + r_{\text{speed}}(t). \quad (11)$$

## 2.6. Training Procedure (Soft Actor–Critic)

We train a stochastic policy  $\pi_\theta(a|s)$  using the Soft Actor–Critic (SAC) framework with two Q-function critics  $Q_\phi^{(1)}, Q_\phi^{(2)}$  to mitigate overestimation. The critics are optimized by minimizing the soft Bellman residual with L2 regularization:

$$L_Q(\phi) = \mathbb{E} \left[ \sum_{i=1}^2 (Q_\phi^{(i)}(s_t, a_t) - \hat{Q}_t)^2 \right] + \lambda_{\text{wd}} \|\phi\|_2^2. \quad (12)$$

with target

$$\hat{Q}_t = r_t + \gamma \left( \min_{i=1,2} Q_\phi^{(i)}(s_{t+1}, a_{t+1}) - \alpha \log \pi_\theta(a_{t+1} | s_{t+1}) \right), \quad a_{t+1} \sim \pi_\theta(\cdot | s_{t+1}). \quad (13)$$

and the actor is updated to maximize the entropy-regularized objective:

$$L_\pi(\theta) = \mathbb{E} [\alpha \log \pi_\theta(a_t | s_t) - Q_\phi(s_t, a_t)], \quad a_t \sim \pi_\theta(\cdot | s_t). \quad (14)$$

We follow a standard SAC setup with automatic temperature tuning for  $\alpha$ , soft target-network updates for the critics, and Adam optimizers with gradient regularization. At each

control step, the agent receives an observation, selects a continuous longitudinal action, and the resulting tuple  $(s_t, a_t, r_t, s_{t+1}, \text{done})$  is appended to a replay buffer; the reward  $r_t$  already includes the SP-based shaping term (Section 2.4). Training proceeds off-policy: the actor continually collects experience, and at fixed intervals a minibatch is drawn from the buffer to (i) update the critics using soft target-network updates, (ii) update the policy under the entropy-regularized objective, and (iii) adjust the entropy temperature via the standard dual formulation. The training hyperparameters are summarized in Table 1.

**Table 1.** Training hyperparameters. Initialized from SAC defaults [11] and minimally adjusted for stability; the same configuration is used for all experiments.  $h_i$  denotes the  $i$ -th hidden layer.

| Parameter                                                                               | Value                                                  |
|-----------------------------------------------------------------------------------------|--------------------------------------------------------|
| Replay buffer capacity                                                                  | 200,000                                                |
| Batch size                                                                              | 256                                                    |
| Update frequency (env steps per optimize)                                               | 5                                                      |
| Actor/Critic/ $\alpha$ learning rate                                                    | $2 \times 10^{-4}/2 \times 10^{-4}/1 \times 10^{-4}$   |
| Discount factor $\gamma$                                                                | 0.99                                                   |
| Soft update coefficient $\tau$                                                          | 0.001                                                  |
| Target entropy                                                                          | −1.0                                                   |
| Initial log- $\alpha$                                                                   | −1.5                                                   |
| L2 regularization (critic)                                                              | $1.0 \times 10^{-4}$                                   |
| Actor network layer sizes (input $\rightarrow h_1 \rightarrow h_2 \rightarrow$ output)  | 49 $\rightarrow$ 256 $\rightarrow$ 256 $\rightarrow$ 1 |
| Critic network layer sizes (input $\rightarrow h_1 \rightarrow h_2 \rightarrow$ output) | 50 $\rightarrow$ 256 $\rightarrow$ 256 $\rightarrow$ 1 |

## 2.7. Experimental Setup

### 2.7.1. Traffic Generation

Background traffic is spawned over the whole map with a random count of 180–190 vehicles per episode. NPCs run the built-in autopilot (CARLA Traffic Manager) with the following parameters:

- headway to the leader: uniformly 6–20 m;
- desired speed scaling: uniformly  $\pm 10\%$ ;
- lateral lane offset: uniformly  $\pm 0.1$  m;
- ignore\_lights: fixed 100% (all NPCs ignore traffic lights to ensure steady inflow into the roundabout);
- ignore\_vehicles: fixed 20% (all NPCs occasionally ignore yielding, inducing near-collision conflicts).

This configuration increases interaction difficulty and exposes the ego policy to diverse risky situations, while being identical across all learning variants. Traffic density and driver-behavior parameters are calibrated not to replicate any particular real-world distribution, but to create a deliberately harsh environment that accelerates learning and magnifies performance differences. For the calibration baseline under the same protocol, see Section 2.7.3.

### 2.7.2. Episode Protocol and Termination

Each episode runs for up to  $T = 700$  fixed simulation ticks. An episode terminates early on (i) successful exit via the designated branch (goal reached), (ii) any collision with dynamic or static actors, or (iii) reaching the step limit. The experimental environment and representative training scenes are shown in Figure 5.



**Figure 5.** Experimental setup in Town 3. (a) Ego vehicle's fixed global route ( $\approx 150$  m): straight approach, roundabout entry with a lane change, and a designated exit branch (green path). (b) Safety Potential computation: yellow points and green cells denote the ego vehicle's planned path and occupancy cells, while purple points and blue cells denote surrounding vehicles' predicted trajectories and occupancy cells; time-weighted overlaps between these cells yield the per-frame SP. The same global route and traffic protocol are used for all experiments to ensure comparability.

### 2.7.3. Rule-Based Baseline (Calibration)

Because this setup is not a standardized community benchmark with a turnkey evaluation suite, a calibration step is necessary. We therefore use CARLA's rule-based stack (BehaviorAgent) as a baseline, and we also treat the No-Safe variant (Section 2.7.4) as an additional reference metric for policy performance.

To calibrate task difficulty, we evaluate BehaviorAgent ("normal" configuration) on the same route and traffic protocol. Across 400 episodes, this baseline yields a collision rate of 20.5%. A post-hoc heuristic audit classifies 6.0% of episodes ( $\approx 29\%$  of collisions) as primarily the responsibility of surrounding vehicles. These figures characterize the task as deliberately challenging; subsequent policy results should be interpreted in light of this difficulty level.

### 2.7.4. Agent Variants and Risk Shaping

We compare four longitudinal controllers that are identical in architecture and learning, differing only in the per-step risk shaping. Lv et al. [8] is a representative on-road RL work that explicitly applies safety rewards by combining TTC and distance gap; we adopt this formulation as the basis for our baselines. To isolate each surrogate, we separate the combined safety term into two baselines and couple each penalty to the longitudinal action. For fairness, baseline settings were lightly tuned to the roundabout scenario.

- No-Safe: no dense per-step risk penalty.
- Distance (forward-distance shaping). Based on Equation (13) [8]; detections are limited to a  $\pm 10^\circ$  forward cone to reduce false alarms. Unlike [8], we use a one-sided gap penalty (no penalty for large gaps) since the speed reward (Equation (10)) already captures efficiency. We set the gap margin to 15 m. With  $\lambda_d = 0.5$ , the shaping term is as follows:

$$r_{\text{dist}}(t) = -\lambda_d \max\left(0, 1 - \frac{d_t}{15}\right) - \lambda_d 1_{\{d_t < 10\}} a_t \left(1 - \frac{d_t}{15}\right). \quad (15)$$

- TTC (time-to-collision shaping). Following Section 4.3 and Equation (13) of Lv et al. [8]—which computes  $\tau_t = \Delta x / \Delta v$  and uses a 2.5 s margin—we adopt TTC as the surrogate and then apply roundabout-specific filters: we take the minimum over valid detections, ignore opposite-heading traffic, and disable the term when the ego is stationary. With  $\lambda_\tau = 0.1$ , the shaping term is

$$r_{\text{ttc}}(t) = -\lambda_\tau \max(0, 2.5 - \tau_t) - \lambda_\tau 1_{\{\tau_t < 1.5\}} a_t (2.5 - \tau_t). \quad (16)$$

- SP (ours): Safety Potential (time-weighted overlap; see Section 2.4) is used as the per-step shaping signal. The SP-based training procedure is summarized in Algorithm 1.

---

**Algorithm 1:** SP-based Training Procedure

---

```

1: Initialize networks and parameters  $(\theta, \phi_1, \phi_2, \alpha)$ .
2: for episode = 1, ..., N do
3:   Spawn the ego vehicle and NPCs.
4:   while not (collision  $\vee$  goal  $\vee$  max_step) do
5:     Set  $N_t$  (NPCs to observe at step  $t$ ) and compose the state  $s = [s_{\text{ego}}; s_{\text{prog}}; s_{\text{look}}; s_{\text{sur}}]$ .
6:     Plan a local ego trajectory  $\zeta_{\text{ego}}$ .
7:     Determine the steering command with the Pure Pursuit controller.
8:     Sample throttle  $a \sim \pi_\theta(a | s)$ , apply (steer,  $a$ ), and step the environment to obtain  $s'$ .
9:     Replan the ego local trajectory  $\zeta'_{\text{ego}}$  and update predicted trajectories for NPCs  $\{\zeta'_u\}_{u \in N_t}$ .
10:    For each  $u \in N_t$ , compute  $SP_u = \text{overlap}(\zeta'_{\text{ego}}, \zeta'_u)$ ; then set  $SP_t = \max_{u \in N_t} SP_u$ .
11:    Compute reward  $r$  and store  $(s, a, r, s', \text{done})$ .
12:    Periodically update  $\theta, \phi_1, \phi_2, \alpha$  with the SAC optimizer.
13:  end while.
14: Destroy the ego vehicle and all NPCs for the next episode.
15: end for.

```

Output: learned longitudinal control policy  $\pi_\theta^{\text{new}}$ .

---

### 3. Results

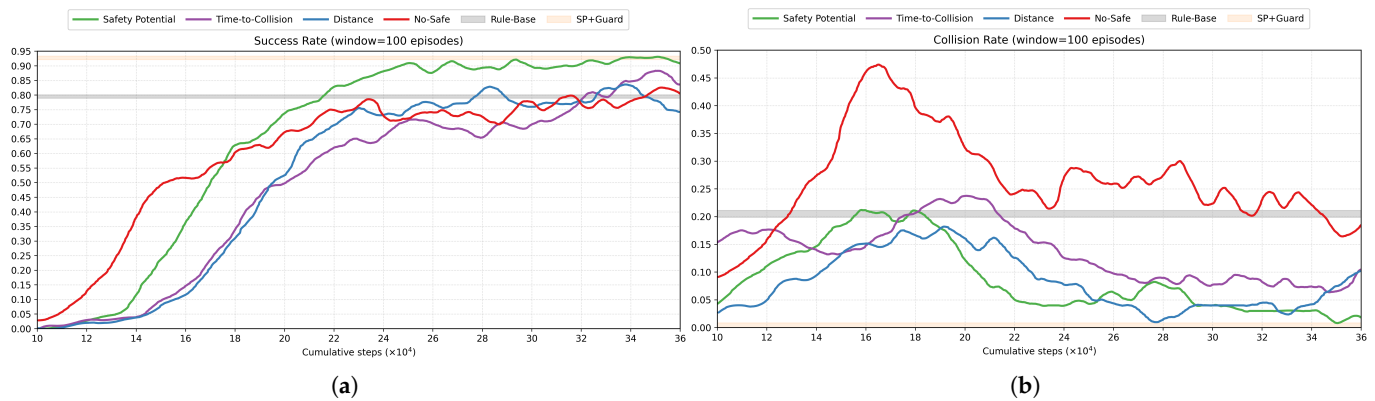
This section evaluates the policies in CARLA through quantitative analysis of task completion, safety, and driving quality, and qualitative diagnostics of risk-aligned behavior and representative collision cases. A separate ablation study examines the effects of enabling a lightweight hard guard at inference and appending a normalized SP term to the state.

#### 3.1. Quantitative Results

We evaluate performance along two primary driving objectives—task completion (success rate, SR) and safety (collision rate, CR)—and assess driving quality via the speed-reward component. Because the safety-related reward terms differ in both scale and formulation, direct comparisons of raw episode returns are not appropriate; reporting per-component metrics isolates task success, safety, and control quality.

##### 3.1.1. Success and Collision Rates over Training

Figure 6 shows that the SP-shaped agent climbs out of the low-performance regime earlier than all baselines and passes the rule-based reference (gray band; see Section 2.7.3) after fewer samples, indicating superior sample efficiency. As training proceeds, its success curve continues to rise smoothly while the collision curve steadily declines, with markedly fewer late-stage oscillations. This stability suggests that the predictive, overlap-based signal provides consistent guidance even as traffic interactions become more frequent, helping the policy consolidate rather than regress.



**Figure 6.** SR/CR. (a) Success-rate learning curves for four policies; SP + Guard (see Section 3.4.1) and the rule-based baseline (see Section 2.7.3) are shown as reference bands. (b) Collision-rate learning curves (same policies and bands).

By contrast, the Distance and TTC variants do not raise success or reduce collisions to the same extent. In Figure 4, their pre-crash curves exhibit higher dispersion and weaker temporal consistency; when used as rewards, such variability can destabilize training. Overall, SP dominates the safety–task frontier: at comparable success, it yields fewer collisions, and at comparable collisions, it achieves higher success. The thin SP+Guard band corresponds to pairing SP with a hard reactive constraint (see Section 3.4.1) and marks the attainable upper region under such a guard.

Deterministic, post-training evaluation in the same environment (Table 2) shows the same preference for SP: with networks frozen and actions taken as the deterministic mean (SAC samples only during training), SP attains the highest success and the lowest total failure (collisions + timeouts) while using more steps on average—consistent with safer, more deliberate behavior without additional mechanisms.

**Table 2.** Performance of learned policies after training (400 episodes in the same environment). Arrows indicate the preferred direction for each metric, and boldface highlights the best value in each column.

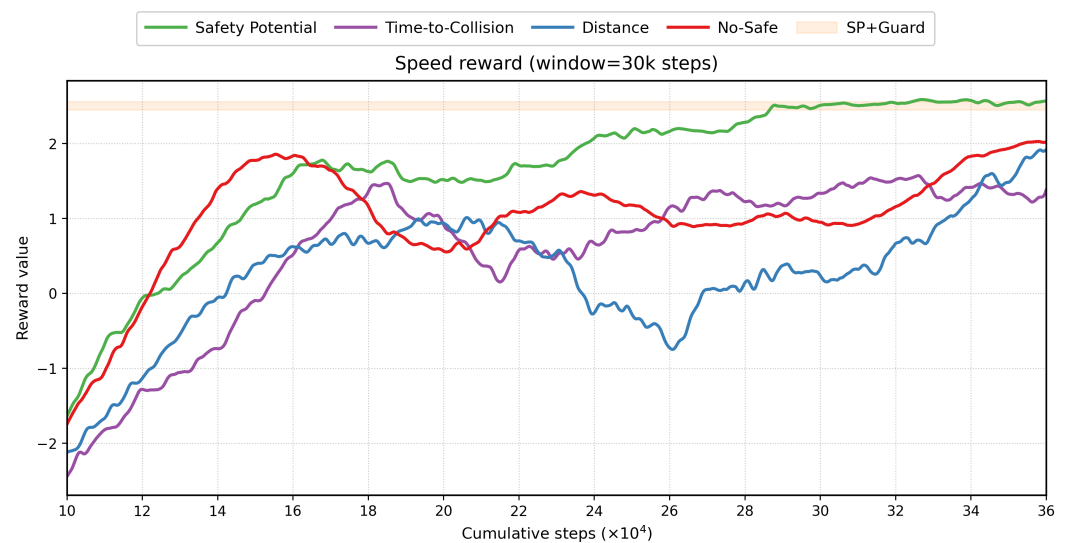
| Method    | Success Rate ↑ | Collision Rate ↓ | Timeout Rate ↓ | Steps (Mean) |
|-----------|----------------|------------------|----------------|--------------|
| No-Safe   | 78.00%         | 21.75%           | <b>0.25%</b>   | 249.5        |
| Distance  | 81.00%         | 12.50%           | 6.50%          | 286.3        |
| TTC       | 88.25%         | 10.25%           | 1.50%          | 345.2        |
| SP (ours) | <b>94.00%</b>  | <b>3.00%</b>     | 3.00%          | 369.2        |

### 3.1.2. Speed Reward Comparison

Figure 7 plots the speed–reward of Equation (10) with a 30k-step moving average to suppress short-term noise and highlight training trends. The metric increases when the agent, under safe conditions, tracks the target speed closely while avoiding overspeeding and unnecessary stops; it is designed to isolate longitudinal regulation quality from safety shaping.

The SP variant attains the highest and most stable plateau late in training, indicating tighter adherence to the target-speed band with fewer over-/underspeed excursions. Distance improves steadily but settles below SP, suggesting more frequent deviations and occasional stop-and-go behavior. TTC also improves but remains below SP, typically with a slower ramp and mid-training oscillations consistent with reactive corrections. No-Safe shows the largest fluctuations, with sharp spikes and dips, reflecting unstable throttle–brake regulation.

This speed-reward serves as a proxy for longitudinal driving quality—smooth attainment of cruising speed with minimal idling or overshoot—independent of SR and CR. In this comparison, SP improves the speed-reward without degrading safety outcomes.



**Figure 7.** Speed reward (step window = 30,000). Curves are smoothed with a 30 k step-based window and plotted against cumulative steps.

### 3.2. Stress-Test Scenarios

We run two within-distribution stress tests (200 episodes each). Test A changes driving style at the training density (headway 5–10 m, target speed +10–30%, lateral offset  $\pm 0.2$  m). Test B increases traffic to 250–260 vehicles while keeping the training style. We avoid unrealistically easy or broken settings: densities below training make the task too simple, while densities above Test B can cause a standstill and hide policy differences. Likewise, making driving style more conservative than training would make the task easier, and making it much more aggressive would make responsibility unclear in crashes. These two setups are therefore slightly harder than training and are meant to test robustness.

Table 3 shows that in Test A, all methods behave similarly to the training setting; safety metrics (SR/CR) remain comparable to training, indicating that the style change does not materially affect safety. In Test B, heavier congestion and tighter gaps raise collisions overall. No-Safe and TTC show the largest increases in collision rate, consistent with point-to-point measures struggling in dense, closely packed traffic. Distance shows a small drop in collisions but a clear rise in timeouts, indicating progress stalls under crowding. By contrast, SP holds up better: collision rates do not rise in the same way, and success remains competitive, suggesting less overfitting to the training setup.

**Table 3.** Stress-test results under distribution shift (200 episodes each). Arrows indicate the preferred direction for each metric, and boldface highlights the best value in each column.

| Method    | Test A: Aggressive Driving |                        |                      |              | Test B: High Density |                        |                      |              |
|-----------|----------------------------|------------------------|----------------------|--------------|----------------------|------------------------|----------------------|--------------|
|           | Success $\uparrow$         | Collision $\downarrow$ | Timeout $\downarrow$ | Steps (Mean) | Success $\uparrow$   | Collision $\downarrow$ | Timeout $\downarrow$ | Steps (Mean) |
| No-Safe   | 74.0%                      | 25.0%                  | <b>1.0%</b>          | 260.6        | 69.0%                | 30.5%                  | <b>0.5%</b>          | 276.9        |
| Distance  | 79.0%                      | 10.5%                  | 10.5%                | 274.4        | 80.4%                | 9.8%                   | 9.8%                 | 293.9        |
| TTC       | 89.5%                      | 9.0%                   | 1.5%                 | 347.9        | 78.5%                | 17.5%                  | 4.0%                 | 387.9        |
| SP (ours) | <b>90.0%</b>               | <b>5.5%</b>            | 4.5%                 | 365.2        | <b>92.0%</b>         | <b>3.5%</b>            | 4.5%                 | 397.7        |

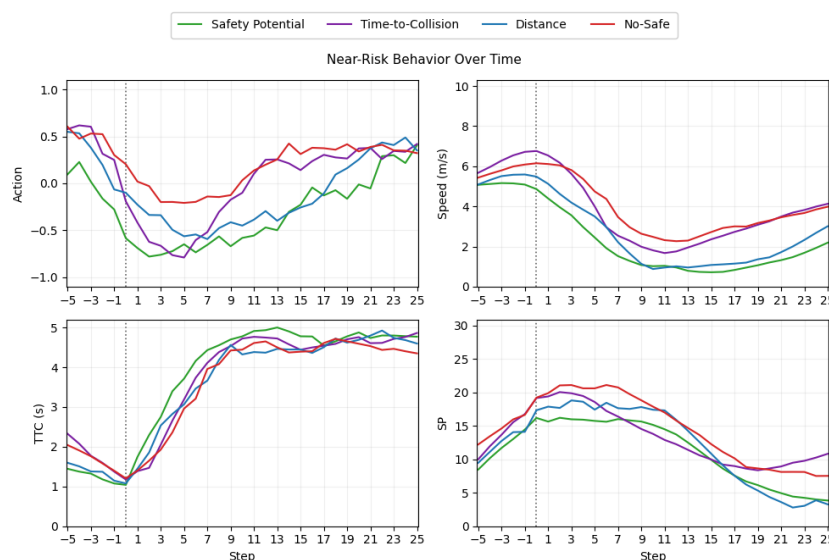
### 3.3. Qualitative Study

We complement the quantitative results with two qualitative analyses. We examine risk-aligned behavior around hazard onset by time-aligning events and summarizing action, speed, SP, and TTC, and we review representative collision cases to highlight typical failure modes and recovery patterns.

#### 3.3.1. Risk-Aligned Behavior Around Hazard Onset

Although TTC and SP are not perfect ground-truth measures of risk, their tail regimes are less prone to false alarms and align with genuinely hazardous states. Leveraging this, the trigger in the training environment is defined as the first time both  $TTC \leq 1.5$  s and  $SP > 15$  hold; events are aligned at step 0 and analyzed over a 3 s window spanning the preceding 5 steps (0.5 s) and the subsequent 25 steps (2.5 s).

Figure 8 shows that, relative to the alternatives, the SP-shaped policy initiates braking earlier and with greater magnitude before the hazard materializes, and its control trajectory around step 0 evolves more smoothly with fewer reversals. Following the trigger, TTC rises with a steeper initial slope and remains on a higher plateau, while SP drops more rapidly and stabilizes at a lower level, indicating quicker clearance and more stable resolution of the hazardous interaction. Speed recovers more slowly yet more monotonically, suggesting a deliberate choice to maintain a safety buffer once risk has been suppressed. Overall, the policy reallocates effort to earlier preemption, achieves stronger and more sustained TTC recovery with persistently low SP, and accepts slower post-event speed recovery as the trade-off for consistent risk reduction across events.

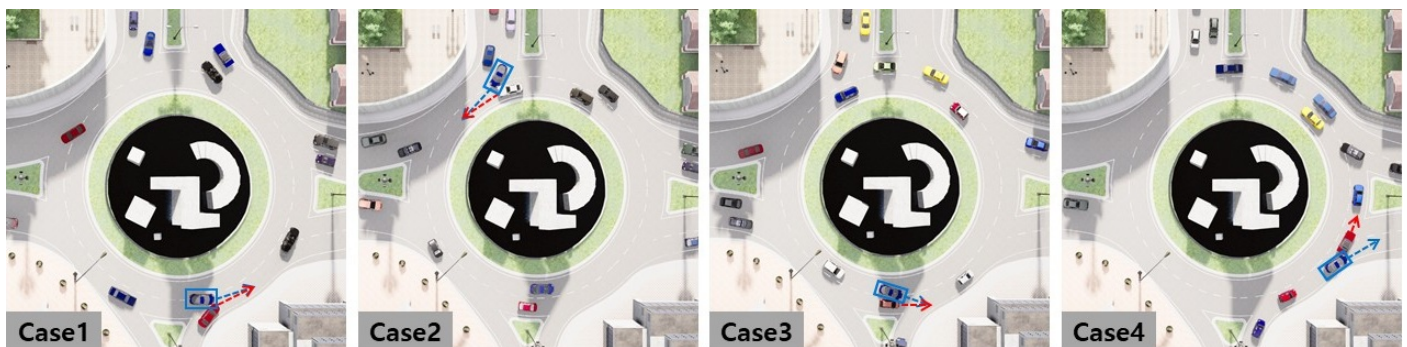


**Figure 8.** Risk-aligned averages around hazard onset ( $TTC \leq 1.5$  s and  $SP > 15$ ). For each detected event, we extract a window from  $-0.5$  s to  $+2.5$  s aligned at step 0 and plot mean action, speed, SP, and TTC (no confidence bands).

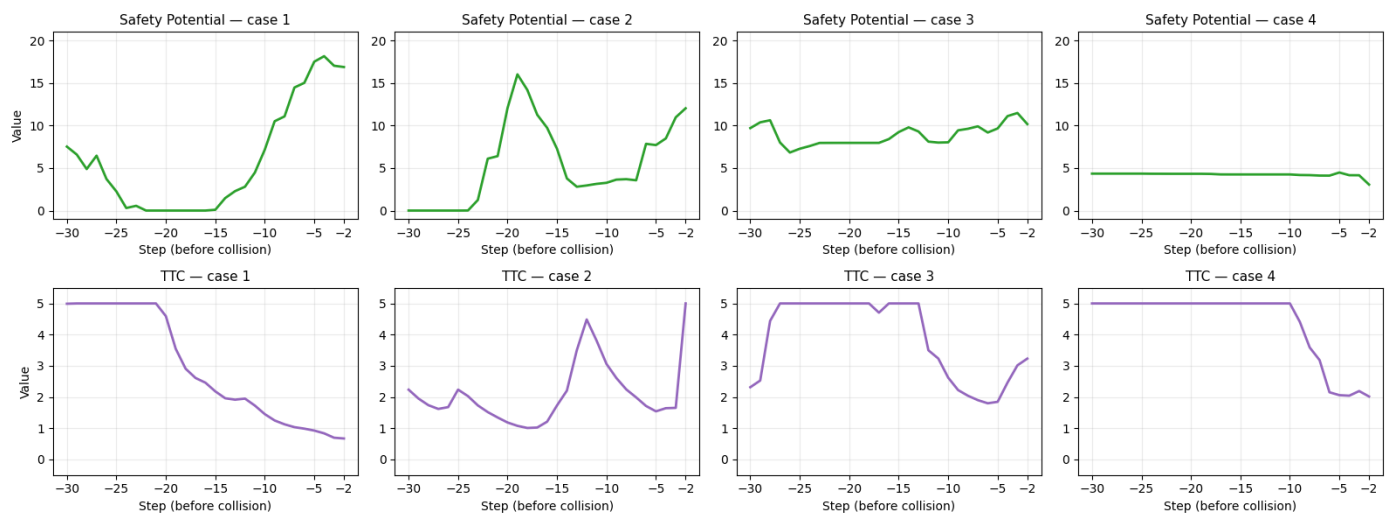
#### 3.3.2. Collision Case Analysis

After training, we analyzed four representative collision cases from additional rollouts using the SP policy to inspect typical failure modes and the limits of single-step scalar risk surrogates. Figures 9 and 10 show spatial overviews of the collisions and the temporal risk profiles for the 30 steps preceding each accident. Neither SP nor TTC exhibits a universally monotonic trend toward collision; although SP typically increases and TTC typically decreases as collision approaches, both metrics fluctuate with local geometry and interactions. This indicates the learned policy optimizes a mixture of objectives and that no single surrogate can guarantee safety in all situations. While SP provides a more forward-

looking signal useful for learning, reactive measures remain valuable complements for real-time safety assurance.



**Figure 9.** Top-down visualization of four collision cases in Town 3. Ego vehicle is highlighted with a blue box, and arrows indicate the driving directions of the ego and surrounding vehicles.



**Figure 10.** Risk progression in the 30 steps prior to collision. Each column shows Safety Potential (SP, green) and Time-to-Collision (TTC, purple) traces for one accident case.

#### Analysis of Collision Scenarios and SP/TTC Distributions:

- Case 1. The ego (blue box) changes from the inner to the outer lane as a red vehicle enters the same outer lane from an entrance. Because the first path overlap occurs only on a far segment of the ego's plan, SP rises late as the vehicles converge, consistent with the delayed increase before the collision.
- Case 2. The ego (blue box) merges from an entrance into the inner lane while a white vehicle is already traveling there. SP briefly rises as a passing outer-lane car dominates, dips when it clears, then rises again as the inner-lane conflict becomes critical. Because SP takes the maximum overlap across surrounding vehicles, this temporary dominance understated the inner-lane risk; the ego kept merging, and a collision followed.
- Case 3. The ego (blue box) moves outward from the inner lane while a reddish-orange vehicle on the outer lane restarts from a stop. Because the first path overlap lies farther along the ego's plan, SP stays at a moderate level without a sharp rise; risk is flagged but not high enough to trigger a stronger response, and a collision follows.
- Case 4. A red vehicle is blocked by a lead car ahead, while the ego (blue box) attempts to pass from behind. The path overlap is partial and oblique to the body, so SP stays very low ( $<5$ ) despite footprint intersection, leading to underestimated risk.

Overall, these cases indicate that a single scalar risk proxy can leave blind spots in complex interactions. When used solely as a reward signal, SP does not fully capture atypical temporal profiles or spatial distributions of risk, leaving some patterns under-penalized. The base learner remains SAC; we modify only the reward by adding an SP shaping term. Because the critic’s Q-value aggregates multiple objectives (speed, progress, arrival), the policy does not necessarily halt when SP rises and can drift into high-risk configurations. In the sequel, we therefore consider complementary design choices that improve robustness.

### 3.4. Ablation Study

We evaluate two small variants while keeping the learner and training setup unchanged: (i) Safety-augmented Policies in the Training Environment, which adds a lightweight hard guard only at inference; and (ii) SP as a State Feature, which appends a normalized SP scalar to the observation to test stability in the training environment.

#### 3.4.1. Safety-Augmented Policies in the Training Environment

Using the learned policy, we conduct additional rollouts in the same training environment (Town 3; identical route and traffic protocol with 180–190 NPCs) while enforcing a lightweight reactive guard only at execution time. The guard intervenes on longitudinal control by applying full brake whenever  $TTC < 2.0$  s or when the nearest object within a  $40^\circ$  forward cone ( $\pm 20^\circ$  about the ego heading) is closer than 8 m; steering remains unchanged and is generated as in the base setup. Each method (with/without the guard) is evaluated over 400 episodes to average out the small absolute differences.

Table 4 shows that the guard consistently reduces collisions across methods. Notably, the SP policy benefits the most: combining predictive shaping (SP) with a simple reactive brake rule achieves the lowest collision rate while maintaining competitive success. Out of a total of 400 episodes, only a single collision occurred under SP+Guard. This incident was primarily caused by the behavior of an opponent vehicle, rather than the ego’s policy.

**Table 4.** Performance of guard-enabled variants. Success, collision, timeout rates, and average steps per episode are reported (400 evaluation episodes). Arrows indicate the preferred direction for each metric, and boldface highlights the best value in each column.

| Method                | Success $\uparrow$ | Collision $\downarrow$ | Timeout $\downarrow$ | Steps (Mean) |
|-----------------------|--------------------|------------------------|----------------------|--------------|
| No-Safe + Hard Guard  | 90.50%             | 4.50%                  | <b>5.00%</b>         | 329.7        |
| Distance + Hard Guard | 82.75%             | 3.00%                  | 14.25%               | 382.0        |
| TTC + Hard Guard      | 87.25%             | 2.50%                  | 10.25%               | 414.7        |
| SP + Hard Guard       | <b>92.75%</b>      | <b>0.25%</b>           | 7.00%                | 412.6        |

#### 3.4.2. SP as a State Feature

We append a scalar  $SP_t/30$  to each observation, expanding the observation to 50 dimensions. Because SP is derived from near-future path overlap with surrounding vehicles, adding it as a state feature injects predictive interaction information, though only in a compressed scalar form. Evaluations are conducted on 400 episodes in the same environment as training.

As shown in Table 5, exposing SP only as a state feature improves success and reduces collisions relative to the No-SP baseline, but remains below the reward-only variant. Using SP solely as a reward yields larger gains in both success rate (SR) and collision rate (CR). Combining state and reward provides a small additional benefit over either alone and also lowers the mean step count, indicating quicker episode completion under the same task conditions.

**Table 5.** Evaluation in the same environment as training; 400 episodes per method. Arrows indicate the preferred direction for each metric, and boldface highlights the best value in each column.

| Method            | Success Rate ↑ | Collision Rate ↓ | Timeout Rate ↓ | Steps (Mean) |
|-------------------|----------------|------------------|----------------|--------------|
| No-SP (=No-Safe)  | 78.00%         | 21.75%           | <b>0.25%</b>   | 249.5        |
| SP state only     | 89.25%         | 8.00%            | 2.75%          | 292.5        |
| SP reward only    | 94.00%         | 3.00%            | 3.00%          | 369.2        |
| SP state + reward | <b>94.25%</b>  | <b>1.75%</b>     | 4.00%          | 339.2        |

#### 4. Discussion and Conclusions

This study introduced Safety Potential as a predictive risk signal in a SAC pipeline for longitudinal control and used it for reward shaping; in CARLA v0.9.14 roundabouts under a matched SAC setup, it outperformed distance- and TTC-based rewards on both quantitative metrics and qualitative driving behavior. We also evaluated a lightweight inference-time hard guard and a state-augmented variant that appends a normalized SP term. Overall, predictive overlap signals provide a practical basis for risk-aware policy learning; when used as rewards or states, they yield safer behavior and are expected to measure risk more faithfully.

Despite these promising results, several limitations remain. The action space was restricted to throttle and brake, which constrained evasive options and may have placed excessive burden on longitudinal control. Observations relied on simulator ground-truth states, including actors that would often be occluded in reality. In addition, there was no head-to-head comparison against standardized baselines in an identical environment, and community benchmarks for this specific task setting (roundabout, longitudinal-only control with fixed lateral policy) are largely absent, making fair cross-paper comparison difficult.

Looking ahead, several extensions are worth exploring. First, broaden the action space to include steering and richer maneuvers, enabling policies to exploit both lateral and longitudinal options. Second, move beyond the full-state assumption by using perception-derived inputs that reflect occlusion, noise, and latency, so the system is trained and evaluated under deployment conditions. Finally, adopt or help establish standardized benchmarks and reporting protocols for this task, and perform matched-environment comparisons against public baselines, with code, scenarios, and seeds released for reproducibility. These steps would better test the robustness and practical value of predictive risk shaping in reinforcement learning for autonomous driving.

**Author Contributions:** Conceptualization, data curation, investigation, methodology, software, validation, visualization, writing, J.C.; funding acquisition, supervision, S.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (RS-2025-02218237, Development of Digital Innovative Technologies for Enhancing the Safety of Complex Autonomous Mobility).

**Data Availability Statement:** The function for collecting the data used in this research from the CARLA simulator, as well as a partial implementation of the method proposed in this article, are available through a GitHub repository (SP\_RL, version v1.0): [https://github.com/JinhoChoi-study/SP\\_RL.git](https://github.com/JinhoChoi-study/SP_RL.git) (accessed on 11 November 2025).

**Conflicts of Interest:** The authors declare no conflicts of interest. The funders had no role in the design of this study; in the collection, analyses, or interpretation of data; in the writing of this manuscript; or in the decision to publish the results.

## Appendix A. Nomenclature (State Vector)

State vector (total dimension = 49).

$$\mathbf{s} = [\mathbf{s}_{\text{ego}} ; \mathbf{s}_{\text{prog}} ; \mathbf{s}_{\text{look}} ; \mathbf{s}_{\text{sur}}]. \quad (\text{A1})$$

**Normalization/encoding rules.** Magnitudes are divided and clipped: distances and speeds are divided by 20, accelerations by 10, then clipped to  $[0, 1]$ . Angular terms are ego-relative directions (velocity, acceleration, yaw) encoded on  $[-1, 1]$ . Surrounding-vehicle slots are zero-filled if fewer than six vehicles are present and truncated to the six nearest otherwise.

### Ego state

$$\mathbf{s}_{\text{ego}} = [v, \theta_v, a, \theta_a]. \quad (\text{A2})$$

- $v$ : ego-vehicle speed—divided by 20; clipped to  $[0, 1]$ .
- $\theta_v$ : direction of ego-vehicle velocity—angular encoding.
- $a$ : ego-vehicle acceleration—divided by 10; clipped to  $[0, 1]$ .
- $\theta_a$ : direction of ego-vehicle acceleration—angular encoding.

### Progress

$$\mathbf{s}_{\text{prog}} = [g]. \quad (\text{A3})$$

- $g$ : goal progress ratio,  $g \in [0, 1]$  (0 start, 1 goal).

### Lookahead target

$$\mathbf{s}_{\text{look}} = [d_\ell, \theta_\ell]. \quad (\text{A4})$$

- $d_\ell$ : lookahead distance—divided by 20; clipped to  $[0, 1]$ .
- $\theta_\ell$ : lookahead direction—angular encoding.

### Surrounding vehicles

$$\mathbf{s}_{\text{sur}} = [\mathbf{s}_1, \dots, \mathbf{s}_6]. \quad (\text{A5})$$

- If fewer than six vehicles are present, remaining slots are zero-filled; if more are present, the six nearest are kept (nearest-first).
- For each surrounding vehicle  $k$ :

$$\mathbf{s}_k = [d_k, \theta_{d,k}, v_k, \theta_{v,k}, \theta_{\text{yaw},k}, a_k, \theta_{a,k}]. \quad (\text{A6})$$

- $d_k$ : distance to vehicle  $k$ —divided by 20; clipped to  $[0, 1]$ .
- $\theta_{d,k}$ : ego-relative direction to vehicle  $k$ —angular encoding.
- $v_k$ : speed of vehicle  $k$ —divided by 20; clipped to  $[0, 1]$ .
- $\theta_{v,k}$ : direction of velocity of vehicle  $k$ —angular encoding.
- $\theta_{\text{yaw},k}$ : yaw angle of vehicle  $k$ —angular encoding.
- $a_k$ : acceleration of vehicle  $k$ —divided by 10; clipped to  $[0, 1]$ .
- $\theta_{a,k}$ : direction of acceleration of vehicle  $k$ —angular encoding.

## References

1. Czechowski, P.; Kawa, B.; Sakhai, M.; Wielgosz, M. Deep Reinforcement and IL for Autonomous Driving: A Review in the CARLA Simulation Environment. *Appl. Sci.* **2025**, *15*, 8972.
2. Delavari, E.; Khanzada, F.K.; Kwon, J. A Comprehensive Review of Reinforcement Learning for Autonomous Driving in the CARLA Simulator. *arXiv* **2025**, arXiv:2509.08221.
3. Hu, Z.; Zhao, D. Adaptive Cruise Control Based on Reinforcement Learning with Shaping Rewards. *J. Adv. Comput. Intell. Intell. Inform.* **2011**, *15*, 351–356.
4. Zhu, M.; Wang, Y.; Pu, Z.; Hu, J.; Wang, X.; Ke, R. Safe, Efficient, and Comfortable Velocity Control Based on Reinforcement Learning for Autonomous Driving. *Transp. Res. Part C* **2020**, *117*, 102662.
5. McLaughlin, S.B.; Hankey, J.M.; Dingus, T.A. A Method for Evaluating Collision Avoidance Systems Using Naturalistic Driving Data. *Accid. Anal. Prev.* **2008**, *40*, 8–16.
6. Hasarinda, R.; Tharuminda, T.; Palitharathna, K.W.S.; Edirisinghe, S. Traffic Collision Avoidance with Vehicular Edge Computing. In Proceedings of the 2023 3rd International Conference on Advanced Research in Computing (ICARC), Belihuloya, Sri Lanka, 23–24 February 2023; IEEE: Piscataway, NJ, USA, 2023.
7. Mahmood, A.; Szabolcsi, R. A Systematic Review on Risk Management and Enhancing Reliability in Autonomous Vehicles. *Machines* **2025**, *13*, 646.
8. Lv, K.; Pei, X.; Chen, C.; Xu, J. A Safe and Efficient Lane Change Decision-Making Strategy of Autonomous Driving Based on Deep Reinforcement Learning. *Mathematics* **2022**, *10*, 1551.
9. Wang, Z.; Liu, X.; Wu, Z. Design of Unsignalized Roundabouts Driving Policy of Autonomous Vehicles Using Deep Reinforcement Learning. *World Electr. Veh. J.* **2023**, *14*, 52.
10. Dong, C.; Guo, N. Biased-Attention Guided Risk Prediction for Safe Decision-Making at Unsignalized Intersections. *arXiv* **2025**, arXiv:2510.12428.
11. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor–Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018.
12. Coulter, R.C. *Implementation of the Pure Pursuit Path Tracking Algorithm*; Technical Report CMU-RI-TR-92-01; Carnegie Mellon University, The Robotics Institute: Pittsburgh, PA, USA, 1992.
13. Dosovitskiy, A.; Ros, G.; Codevilla, F.; López, A.; Koltun, V. CARLA: An Open Urban Driving Simulator. In Proceedings of the 1st Conference on Robot Learning (CoRL), Mountain View, CA, USA, 13–15 November 2017.
14. Minsky, M. Steps toward Artificial Intelligence. *Proc. IRE* **2007**, *49*, 8–30.
15. Mataric, M.J. Reward Functions for Accelerated Learning. In *Machine Learning Proceedings 1994*; Morgan Kaufmann: Burlington, MA, USA, 1994; pp. 181–189.
16. Randalø, J.; Alstrøm, P. Learning to Drive a Bicycle Using Reinforcement Learning and Shaping. In Proceedings of the 15th International Conference on Machine Learning (ICML), San Francisco, CA, USA, 24–27 July 1998.
17. Nagesh Rao, S.; Tseng, H.-E.; Filev, D. Autonomous Highway Driving using Deep Reinforcement Learning. In Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 6–9 October 2019; pp. 2326–2331.
18. Huang, Y.; Xu, X.; Li, Y.; Zhang, X.; Liu, Y.; Zhang, X. Vehicle-Following Control Based on Deep Reinforcement Learning. *Appl. Sci.* **2022**, *12*, 10648.
19. Alarcon, N. DRIVE Labs: Eliminating Collisions with Safety Force Field. NVIDIA Developer Blog. 2019. Available online: <https://developer.nvidia.com/blog/drive-labs-eliminating-collisions-with-safety-force-field> (accessed on 27 August 2025).
20. Suk, H.; Kim, T.; Park, H.; Yadav, P.; Lee, J.; Kim, S. Rationale-Aware Autonomous Driving Policy Utilizing Safety Force Field Implemented on CARLA Simulator. *arXiv* **2022**, arXiv:2211.10237.
21. Leng, B.; Yu, R.; Han, W.; Xiong, L.; Li, Z.; Huang, H. Risk-Aware Reinforcement Learning for Autonomous Driving: Improving Safety When Driving through Intersection. *arXiv* **2025**, arXiv:2503.19690.
22. Yu, R.; Li, Z.; Xiong, L.; Han, W.; Leng, B. Uncertainty-Aware Safety-Critical Decision and Control for Autonomous Vehicles at Unsignalized Intersections. *arXiv* **2025**, arXiv:2505.19939.
23. Gan, J.; Zhang, J.; Liu, Y. Research on Behavioral Decision at an Unsignalized Roundabout for Automatic Driving Based on Proximal Policy Optimization Algorithm. *Appl. Sci.* **2024**, *14*, 2889.
24. Lin, Z.; Tian, Z.; Lan, J.; Zhang, Q.; Ye, Z.; Zhuang, H.; Zhao, X. A Conflicts-Free, Speed-Lossless KAN-Based Reinforcement Learning Decision System for Interactive Driving in Roundabouts. *arXiv* **2024**, arXiv:2408.08242.

**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.