2026 IEEE/SICE International Symposium on System Integration (SII)
January 11-14, 2026. Cancún, México

From Specification to Certification: TORQ-Ordered Rulebooks and Robust HOCBF Optimization for Safe Autonomous Driving

Hadi Hajieghrary[1], Benedikt Walter[2], and Paul Schmitt[3]

Abstract— **Autonomous vehicle (AV) planners must satisfy complex, often conflicting, safety constraints, traffic laws, and comfort norms. Conventional methods like formal logics and optimal control may fail under rule conflicts, while learning-based policies lack the necessary formal guarantees for certification. This paper introduces a unified rulebook framework that encodes heterogeneous driving rules as differentiable violation metrics structured by total order over equivalence classes (TORQ). This removes rule incomparability and enables lexicographical optimization of trajectories. The specification integrates into real-time control using robust High-Order Control Barrier Functions (HOCBFs) and Control Lyapunov Functions (CLFs) solved via Sequential Quadratic Programming (SQP). A recursive relaxation algorithm maintains the hierarchy of the rules, allowing violations of only the lowest-priority rules necessary to resolve conflicts. Extensive simulations, including urban intersections and lane drift scenarios on roads, demonstrate that the system consistently prioritizes high-level safety mandates. By combining formal specification, real-time synthesis, and verification, this framework offers a robust, certifiable, and transparent approach to AV behavior planning.**

I. INTRODUCTION

Autonomous vehicle (AV) decision-making must simultaneously satisfy diverse and often conflicting requirements: safety constraints, traffic laws, comfort heuristics, and driving conventions. Emergency maneuvers may necessitate violating traffic laws, highlighting a central challenge: developing a specification formalism that can (i) comprehensively enumerate heterogeneous rules, (ii) explicitly define their relative priorities, and (iii) integrate with real-time control synthesis and formal verification pipelines [1], [2], [3].

Current AV planning approaches have significant limitations under rule conflicts:

- **Specification Logic (LTL, STL, scLTL):** Binary satisfaction predicates struggle with conflicting constraints, leading to infeasibility without compromise mechanisms.
- **Safety-Critical Control (CLFs, CBFs):** Effective for individual constraints, but QP-based methods render infeasible when high-order CBFs clash with other limitations.
- **Learning-Based Planners:** Promising for complex situations, but "black-box" nature creates a verification gap hindering certification.
- **Rulebooks and TORQ:** Use continuous violation metrics and structured priorities [4], [5], but seamless integration into real-time certified control remains challenging.

This work introduces a deterministic rulebook framework encoding each rule as a smooth, differentiable violation metric, enabling gradient-based optimization to find minimum-violation solutions when perfect compliance is impossible. A strict TORQ priority structure imposes complete ordering, enabling robust lexicographic synthesis [5]. This framework provides verifiable "guardrails" for powerful but unpredictable AI systems.

International standards (ISO/PAS 8800:2024, UNECE WP.29) underscore the imperative for "explicability and controllability" in AI systems, which our framework directly addresses. Our deterministic rulebook framework unifies specification, control, and verification for certifiable real-time AV decisions. Unlike penalty-based methods, our lexicographic procedure provably finds the Pareto minimal solution for a given rule hierarchy. Key contributions:

- **Formal Rulebook:** Continuously differentiable violation metrics with Totally Ordered Rule-based QoS (TORQ) structure providing computationally tractable, unambiguous conflict resolution.
- **Robust Real-Time Solver:** Sequential Quadratic Programming (SQP) solver using High-Order Control Barrier Functions (HOCBFs) and Control Lyapunov Functions (CLFs), finding Pareto minimal solutions with guaranteed graceful degradation during conflicts.
- **Auditable Justification:** Ex-post oracle verifying trajectory optimality, producing auditable post-hoc justifications satisfying regulatory transparency demands.
- **Simulation Validation:** Challenging driving simulations demonstrating real-time decisions consistently honoring rule hierarchies while maintaining safety with clear compromise explanations.

By unifying specification, synthesis, and verification, our framework provides a robust path toward safe and certifiable autonomous systems, directly addressing critical engineering and regulatory demands.

A. Background and Related Work

Decision-making for AVs integrates formal specification, optimal control, and machine learning. This section surveys field evolution, highlighting limitations our framework addresses.

1) From Binary Logic to Verified Quantitative Rules

Early efforts encoded traffic laws using temporal logics (LTL, STL) [6], enabling automated monitoring but proving brittle under unavoidable conflicts. Quantitative semantics introduced violation degree measurement rather than binary predicates, enabling minimum-violation synthesis [5]. Recent work uses theorem provers (Isabelle/HOL) to formally verify rule formalization soundness and auditability [7].

[1]Hadi Hajieghrary Hadi.Hajieghrary@Torc.ai, [2]Benedikt Walter Benedikt.Walter@Torc.ai, and [3]Paul Schmitt Paul.Schmitt@Torc.ai are with TORC Robotics LLC, an independent subsidiary of Daimler Truck AG

2) Rulebooks and Lexicographic Optimization

Censi et al. introduced rulebooks—finite sets of differentiable rules ordered by priority [4]—allowing nuanced trade-offs but suffering ambiguity under partial ordering. The TORQ framework enforces strict total order, enabling deterministic lexicographic optimization where higher-priority rules are always satisfied first [5]. This differs from rigid industry standards like RSS and has been successfully applied in model predictive control [1]. Alternative asymptotic penalty representations encode hierarchy into a single differentiable objective function, natively compatible with gradient-based optimization.

3) Safety Enforcement with Control Barrier Functions

Control Barrier Functions (CBFs) define safe state-space regions, ensuring systems never leave them. For mechanical systems where control inputs have delayed effects, high-order control barrier functions (HOCBFs) are essential [5], [8]. Key challenges include ensuring solver feasibility when multiple constraints conflict, which is often addressed through penalty-based relaxations [9], [10]. Advances include Robust CBFs (RCBFs) for disturbances and Stochastic CBFs (SCBFs) for uncertainty.

4) Learning-Augmented Control

Recent research combines control theory guarantees with machine learning adaptability: (1) using Large Language Models (LLMs) for high-level reasoning and translating regulations into formal specifications; (2) learning Neural Barrier Functions for systems with unmodeled dynamics; and (3) developing formal verification for "black-box" neural components.

Despite promise, learning-based systems often lack hard safety guarantees, auditability, and real-time responsiveness. Our framework complements these approaches, providing a formally verifiable, deterministic backbone that can serve as safety-critical fallback controller, monitor and validate learning-based planner outputs, and enable rigorous certification of hybrid systems.

II. RULEBOOK FRAMEWORK

This section formalizes the Rulebook Framework, substantially extending earlier formulations [4], [5] to systematically encode, prioritize, and evaluate traffic rules with mathematical precision.

A. Notation and Preliminaries

Let the autonomous vehicle state space be $X \subseteq \mathbb{R}^n$ with state vector $\mathbf{x} \in X$. We consider a control-affine system:

$$\dot{\mathbf{x}} = \mathrm{f}(\mathbf{x}) + \mathrm{g}(\mathbf{x})\mathbf{u}, \tag{1}$$

where $\mathbf{u} \in U \subseteq \mathbb{R}^m$ is the control input. A trajectory is a continuous function $\tau : [t_0, t_f] \to X$.

Assumption 1: The functions $\mathrm{f} : \mathbb{R}^n \to \mathbb{R}^n$ and $\mathrm{g} : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are locally Lipschitz continuous (standard for nonlinear vehicle models).

Assumption 2: Each rule r_i is defined by function $\mathrm{h}_i : \mathbb{R}^n \to \mathbb{R}$ where $\mathrm{h}_i(\mathbf{x}) \geq 0$ indicates satisfaction. Functions h_i are at least C^2 (required for HOCBF formulation; nonsmooth rules require approximation or alternative solvers).

We quantify trajectory adherence via differentiable violation metric $\rho_r(\xi) \in \mathbb{R}_{\geq 0}$ for each rule $r \in \mathcal{R}$, where $\rho_r(\xi) = 0$ indicates perfect compliance. The violation metric derives from the temporal integral of violations of the instantaneous constraint function $h_r(\mathbf{x}(t))$.

The core specification framework is the Total Order over Equivalence Classes (TORQ), providing complete, unambiguous trajectory comparison.

Definition 2.1: A TORQ is a triple $\langle R, \sim, \prec \rangle$ where R is a finite set of rules, $\sim$ partitions R into equivalence classes, and $\prec$ orders those classes. For trajectory τ, define the satisfaction map $\Phi(\tau) \coloneqq (\rho_1(\tau), \ldots, \rho_m(\tau))$ as the violation metric vector sorted by priority order $\prec$. Then $\tau_1 \prec_R \tau_2$ if and only if $\Phi(\tau_1)$ is lexicographically smaller than $\Phi(\tau_2)$.

This eliminates incomparability ambiguity, ensuring consistent decision-making. A small violation of a high-priority rule always outweighs any lower-priority breach.

B. Lexicographical Trajectory Comparison

TORQ induces complete preorder in the trajectory space $\mathcal{T}$. Given trajectories ξ_1 and ξ_2, we compare them lexicographically by violation metrics ordered by priority.

Definition 2.2: A trajectory ξ_1 is strictly preferred to ξ_2 (denoted $\xi_1 \prec_{\mathcal{R}} \xi_2$) if and only if exists $r^* \in \mathcal{R}$ such that:

1) $\rho_{r^*}(\xi_1) < \rho_{r^*}(\xi_2)$,
2) For all higher-priority rules r' (i.e., $r^* \prec r'$): $\rho_{r'}(\xi_1) = \rho_{r'}(\xi_2)$.

C. Theoretical Guarantees

Our framework combines TORQ specification with recursive relaxation for HOCBF-based optimal control. The guarantees arise from recursive relaxation itself; asymptotic formulation offers an alternative for gradient-based solvers.

Theorem 2.3 (Existence): Under Assumptions 1 and 2, for any state $\mathbf{x}$ and active HOCBF constraints with actuator constraints $\mathbf{u} \in U$, Algorithm 1 guarantees finding a feasible control input if one exists satisfying all higher-priority class constraints.

Proof sketch: By induction on priority classes. Base case: if the highest-priority QP is feasible, proceed; if not, the problem is inherently unsolvable. Inductive step: if the QP including constraints through level k is infeasible, relaxing level k strictly enlarges the feasible set, ensuring eventual solution if one exists respecting level $k+1$ and higher. □

Numerical note. In practice, normalize $\rho_i \in [0, 1]$ and cap weight scale (e.g., $\lambda \leq 10^6$) to avoid ill-conditioning; alternatively, use log-sum-exp surrogate.

Theorem 2.4 (Minimality): Under Assumptions 1 and 2, control input $\mathbf{u}^*$ and trajectory τ^* synthesized by Algorithm 1 are Pareto minimal for violation vector $\Phi(\tau)$ in lexicographical order $\prec_R$.

Proof sketch: The recursive top-down structure directly mirrors lexicographical minimization. The algorithm minimizes highest-priority violations, fixes that level, then minimizes next-priority violations. Rules relax only when impossible to satisfy without violating higher-priority rules, establishing Pareto minimality. □

Notation. Let L_0 denote priority classes already satisfied at initial state, with cardinality $|L_0|$.

TABLE I: Abridged Rule Catalogue: An illustrative subset of rules (including priority, description, and metrics) to be expanded for scenarios like adverse weather, emergencies, and construction.

ID	Level	Description	Violation Metric ρ_i	Rel. deg
r_0	10	Avoid VRU Collision	Severity integral	2
r_1	9	Avoid Non-VRU Collision	Severity integral	2
r_2	8	Comply with Mandatory Stops	Penalty for rolling/short stops	2
r_3	7	Stay on Drivable Surface	Integral of distance outside boundaries	2
r_4	7	Obey Traffic Lights	Normalized time in intersection on red/unsafe yellow	2
r_5	7	Avoid Oncoming Lane	Normalized time in opposing lane with oncoming traffic	2
r_{16}	3	Obey Speed Limits	Integral of speed exceeding limit	2
r_{17}	3	Maintain Headway	Integral of time-headway deficit	2
r_{18}	3	Maintain Lateral Clearance	Integral of lateral distance deficit	2
r_{19}	3	Respect Friction Limits	Integral of required friction deficit	1
r_{24}	1	Limit Overall Jerk	Integral of jerk magnitude	1
r_{25}	1	Limit Lateral Acceleration	Integral of lateral acceleration excess	1
r_{27}	0	Minimize Path Deviation	Integral of lateral deviation from reference	2
r_{29}	0	Minimize Energy Use	Integral of squared acceleration	1

Theorem 2.5: The recursive relaxation algorithm terminates in at most $m - |L_0|$ steps, where m is the total number of rule equivalence classes.

Proof sketch: Algorithm iterates through m priority classes in single downward pass. Each feasible QP terminates; otherwise, one class relaxes per step. No relaxation needed for initially satisfied classes. □

D. Certification Oracle

A pass/fail oracle formally verifies candidate trajectory optimality within TORQ structure, providing auditable compliance artifacts.

Definition 2.6: Given candidate trajectory ξ_c and bounded perturbations, the oracle:

- Solves robust TORQ-based optimization to find optimal trajectory ξ^* under worst-case perturbations.
- If $\Phi(\xi^*) <_{lex} \Phi(\xi_c)$ (i.e., $\xi^* \prec_{\mathcal{R}} \xi_c$), reject ξ_c.
- Otherwise, certify ξ_c complies with the rulebook.

This guarantees robust optimality under the disturbance model, though computational demand is a practical consideration.

Recursive relaxation (Algorithm 1) provides procedural lexicographical minimization. Alternatively, recent work offers direct declarative formulation via single differentiable utility function:

$$\min_x f(x, \lambda) = \min_x \sum_{i=1}^{m} \lambda^{k_i} \rho_{r_i}(x)$$

where x represents decision variables, $\rho_{r_i}(x)$ is the violation metric for rule r_i, and exponents k_i form strictly decreasing sequence (e.g., $k_i = m - i + 1$). As $\lambda \to \infty$, the highest-priority violation term dominates, forcing lexicographically minimal solutions. While elegant, this can introduce numerical conditioning issues for large λ. If the objective is presented as $\sum_{\ell=1}^{m} \lambda^{\ell} S_{\ell}(\tau)$ with $S_{\ell} = \sum_{i:k_i=\ell} \rho_i$, boundedness implies improvements in higher levels dominate lower-level changes as $\lambda \to \infty$, reproducing lexicographic order.

III. FORMALIZATION OF TRAFFIC RULES

We derive continuously differentiable violation metrics for common traffic rules as the foundation of the TORQ-based optimal control architecture. Each rule is expressed as a differentiable cost function—often quadratic—for direct compatibility with the Sequential Quadratic Programming (SQP) solver. This integrated design of rule specification and control synthesis ensures real-time tractability and formal guarantees. Rules are organized into priority levels (Level 10 $\to$ Level 0), establishing a strict lexicographic order for trajectory comparison within the TORQ framework [9]. The hierarchy reflects a principled decomposition of the driving task into thematic clusters, capturing the logic of AV decision-making. While based on established safety principles, practical deployment would require empirical validation of priorities to balance ethical and operational objectives. The clusters are organized as follows:

- **Levels 10-7: Core Safety and Legality.** This highest-priority cluster contains rules fundamental to the safety of other road users and adherence to non-negotiable statutes.
- **Levels 6-4: Societal and Strategic Objectives** This group governs the behavior of the vehicle as a 'good citizen'. This includes its "roadmanship" [11] (i.e., predictability and legibility, motion anticipation, and courtesy), and its long-term strategic risk management.
- **Levels 3-1: Dynamic Compliance and Comfort.** This cluster manages fine-grained interactions, optimizes passenger comfort, and minimizes vehicle wear and tear.
- **Level 0: Efficiency.** This is the baseline objective, pursued only when all higher priority rules are satisfied to the greatest extent possible.

The violation metrics serve a dual purpose. First, they provide the scalar costs $\rho_r(\xi)$ used for the lexicographical comparison of trajectories. Second, and more critically, they are the direct source for defining the continuously differentiable functions $h(x)$ used to formulate the High-Order Control Barrier Function (HOCBF) constraints within the QP solver. The condition $h(x) \geq 0$ defines the safe set for a given rule. This one-to-one mapping from a human-readable rule to a mathematical constraint is a key feature of our framework's transparency and verifiability. We choose (τ_i, α_i) so that a zero raw violation maps near zero in the normalized scale, and $V_{\text{raw},i} = \tau_i$ maps to 0.5; this avoids unintended baselines.

IV. RULE-BASED OPTIMAL CONTROL

This section describes the control layer computing rule-compliant control actions in real time via prioritized, constrained optimal control. A nonlinear vehicle model couples with the deterministic rulebook via High-Order Control Barrier Functions (HOCBFs) and Control Lyapunov Functions (CLFs). Continuously differentiable rules embed in a Sequential Quadratic Programming (SQP) pipeline with lexicographic slack strategy, guaranteeing feasibility and graceful degradation under conflicts.

RHC usage. Algorithm 1 applies at each control tick as a receding-horizon QP with warm-starts from the previous tick. Only the first control is applied before shifting the horizon, preserving lexicographic guarantees per tick.

Over prediction horizon T discretized into N steps, ego state is $\mathbf{x}_k \in \mathbb{R}^n$ and control input $\mathbf{u}_k \in \mathbb{R}^m$ in each step k. We seek control sequence $\mathbf{U} \coloneqq [\mathbf{u}_0^\top, \ldots, \mathbf{u}_{N-1}^\top]^\top$ minimiz-

Algorithm 1 Recursive HOCBF Relaxation

Require: State $\mathbf{x}$ and input $\mathbf{u}_{\text{ref}}$, and rulebook $\langle R, \sim, \prec \rangle$,
Ensure: Feasible, minimal-violation control $\mathbf{u}^*$
$C_1, \ldots, C_m \leftarrow$ rules sorted from high to low priority
$S_{\text{relaxed}} \leftarrow \emptyset$
for $k = m$ **downto** 1 **do**
  Form QP with constraints hard for $R \setminus \bigcup_{C_j \in S_{\text{relaxed}}} C_j$ and soft for rules in S_{relaxed}
  if QP is feasible **then**
    return optimal control $\mathbf{u}^*$ from QP solution
  else
    $S_{\text{relaxed}} \leftarrow S_{\text{relaxed}} \cup \{C_k\}$ ▷ Relax next lowest priority class
  end if
end for

ing cumulative rule violation cost, weighted by priority:

$$\min_{\mathbf{U},\,\boldsymbol{\varepsilon}} \ \tfrac{1}{2}\,\mathbf{U}^\top H\,\mathbf{U} + f^\top \mathbf{U} + \sum_{i=1}^{m} \lambda_i\,\varepsilon_i^2, \tag{2}$$

subject to vehicle dynamics, actuator limits, and rule constraints. Weights satisfy $\lambda_i \gg \lambda_j$ when rule r_i outranks r_j in TORQ priority, forcing relaxation of only lowest-priority rules under conflict. We adopt discrete-time control-affine model with state $\mathbf{x}_k = [p_x, p_y, \theta, v, \ldots]^\top$ and control $\mathbf{u}_k = [a, \delta]^\top$ (longitudinal acceleration and steering angle).

Each rule r_i is represented by continuously differentiable function $h_i(\mathbf{x})$ where $h_i(\mathbf{x}) \geq 0$ implies compliance. We adopt robust HOCBF (RCBF) formulation for real-world disturbances. Modeling disturbances as additive term $\dot{x} = f(x) + g(x)u + d$ with $\|d\| \leq D_{\max}$, the robust constraint for relative degree one becomes:

$$L_f h(x) + L_g h(x) u - D_{\max}\,\|\nabla h(x)\| \geq -\alpha(h(x)) - \varepsilon_i, \tag{3}$$

where $\nabla h(x)$ is the gradient, the bound follows from $\min_{\|d\| \leq D_{\max}} \nabla h(x)^\top d = -D_{\max}\|\nabla h(x)\|$, $\alpha(\cdot)$ is class-$\mathcal{K}$, and $\varepsilon_i \geq 0$ is slack. This ensures constraint satisfaction under worst-case disturbance. Choice of $D_{\max}$ represents robustness-performance trade-off [12]. Goal-seeking objectives formulate as CLF inequalities. After linearization, constraints are affine in $\mathbf{u}_k$.

Algorithm 1 performs top-down search on the rule hierarchy, iteratively relaxing lower-priority active classes until the QP admits solution. Strict slack penalty ordering ($\lambda_1 \gg \lambda_2 \gg \cdots \gg \lambda_m$) ensures higher-priority rules never violate further to improve lower-priority ones.

Collecting horizon-stacked control vector $\mathbf{U}$ and slack vector $\boldsymbol{\varepsilon}$, the real-time QP at every control step:

$$\min_{\mathbf{U},\,\boldsymbol{\varepsilon}} \ \tfrac{1}{2}\,\mathbf{U}^\top H\,\mathbf{U} + f^\top \mathbf{U} + \sum_{i=1}^{m} \lambda_i\,\varepsilon_i^2$$
$$\text{s.t.} \quad A_{\text{dyn}}\mathbf{U} = b_{\text{dyn}},\ \textit{and}\ A_{\text{ineq}}\mathbf{U} \leq b_{\text{ineq}} + E\,\boldsymbol{\varepsilon},\ \boldsymbol{\varepsilon} \geq \mathbf{0}.$$

The QP embeds in receding-horizon loop with warm-starts from previous solution.

Dense QP with n decision variables and N constraints has worst-case complexity $\mathcal{O}(Nn^3)$. Real-time feasibility is achieved through warm-starting, sparse solvers, and state-of-the-art techniques. The recursive nature of Algorithm 1 requires multiple QP solves in worst case; methods like Consensus ADMM can parallelize by decomposing long-horizon problems. High-performance solvers (e.g., SNOPT) and code generation toolchains (e.g., CasADi) precomputing Jacobians and Lie derivatives offline are crucial for onboard vehicle control.

Penalty-weighted slacks can obscure trade-offs; our recursive relaxation preserves strict priorities without weight tuning, ensuring transparent and predictable behavior in safety-critical scenarios.

V. Case Study: Highway Lane Drift (Slowing Drifter) and Tailgater

This section presents illustrative case studies to validate the decision-making logic of the proposed framework. This example is designed to demonstrate how TORQ-based optimization resolves complex rule conflicts in representative, safety-critical driving scenarios. The actor's behaviors are assumed to be known perfectly over the planning horizon, allowing for the isolation and analysis of the core decision logic of the ego vehicle's planner.

This section presents in-depth analysis of the framework applied to a challenging highway scenario involving simultaneous threats from a lead vehicle and tailgater, testing the decision-making system's ability to resolve complex, multi-objective conflicts. The simulated environment parameters are in Table II. The three-lane highway has the ego vehicle initially in the center lane, with three actors: ego vehicle, "Slowing Drifter," and "Tailgater."

TABLE II: Environmental and Rule Parameters

Parameter	Value	Parameter	Value
Road Geometry	3-lane highway	**Regulatory**	
Lane Width	$3.70\,\text{m}$	Speed Limit	$30.00\,\text{m}\,\text{s}^{-1}$
Lane Center	$y = 0.00\,\text{m}$	**Simulation**	
Lane Boundaries	$y = \pm 1.85\,\text{m}$	Planning Horizon	$5.00\,\text{sec}$
Shoulder Boundary	$y = \pm 3.00\,\text{m}$	Time Step	$0.50\,\text{s}$
Rule Constants			
Collision Threshold	$4.80\,\text{m}$	Min. Headway	$2.00\,\text{s}$
Max. Lat. Acc.	$2.00\,\text{m}\,\text{s}^{-2}$	Max. Acc.	$6.00\,\text{m}\,\text{s}^{-2}$
Max. Jerk	$10.00\,\text{m}\,\text{s}^{-3}$		

A. Actor Behaviors and Predictions

Actor behaviors are defined by predictive kinematic models over the planning horizon, assuming perfect deterministic predictions to isolate ego decision-making logic.

1) Actor 1: The Slowing Drifter

Initially ahead of ego in the adjacent left lane, this vehicle simultaneously decelerates and drifts laterally toward ego's lane, creating a closing gap both longitudinally and laterally:

$$x_d(t) = 20 + 29t - t^2, \quad y_d(t) = 3.7 - 0.5t$$

2) Actor 2: The Tailgater

Starting significantly behind ego but traveling at much higher constant velocity, this vehicle creates rapidly closing

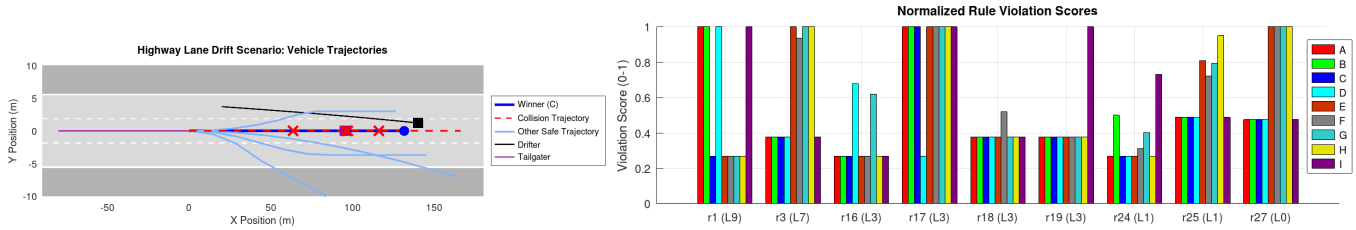

Fig. 1: Simulation outputs for Highway Lane Drift scenario. Trajectory D avoids collisions ($r_1 = 0.2689$, minimal) with both vehicles. High normalized scores for speed limit ($r_{16} = 0.9987$) and friction limit ($r_{19} = 1.0000$) are lower priority (L3) than critical collision avoidance (L9) and operational safety (L7) rules eliminating other trajectories.

gap from the rear, posing significant risk if ego decelerates:

$$x_t(t) = -80 + 35t, \quad y_t(t) = 0$$

B. Ego Vehicle Candidate Trajectories

The following nine candidate trajectories represent possible reactions from passive to aggressive:

- **A: No reaction**: Maintain constant velocity.
- **B: Early Hard Braking**: Strong immediate deceleration.
- **C: Late Mild Braking**: Initial speed maintenance, then mild deceleration.
- **D: Accelerate**: Increase speed to separate from tailgater and pass drifter.
- **E: Lateral Evasion (Left)**: Sharp move to left shoulder.
- **F: Mild Brake + Right Nudge**: Longitudinal and lateral maneuver.
- **G: Accelerate + Left Nudge**: Maneuver to pass left.
- **H: Hard Left Nudge**: Aggressive lateral move to far left.
- **I: Yield to Drifter**: Very hard braking to let drifter pass.

C. Rule Evaluation and Lexicographical Selection

Violation metrics are calculated dynamically based on kinematic states at each time step. Raw violation metrics integrate rule violations throughout the planning horizon, then normalize to $[0, 1]$ using a sigmoid function. Normalized scores for relevant rules appear in Table III.

1) Relevant Rule Definitions

Rule r_1 (Level 9) - Avoid collision with non-VRU vehicles Penalizes collision with any non-VRU object with severity-weighted penalty:

$$V_1 = \sum_{e \in E_{\text{coll}}} \int_{t_{\text{start},e}}^{t_{\text{end},e}} f_{\text{sev}}\big(x_{\text{ego}}, x_{\text{obj},e}, v_{\text{ego}}, v_{\text{obj},e}, \text{type}_e\big)\, \mathrm{d}t \quad (4)$$

where E_{coll} is the set of collision events (continuous overlap for minimum duration $\Delta t_{\text{min,coll}}$), and f_{sev} is severity model (e.g., $w(\text{type})\|v_{\text{ego}} - v_{\text{obj}}\|^2$).

Rule r_3 (Level 7) - Stay within drivable surface Penalizes distance–time beyond legal road edge:

$$V_3 = \int_0^T \left[\max(0, -d_{\text{boundary}}(t))\right]^p \mathrm{d}t$$

where $d_{\text{boundary}}(t)$ is signed lateral distance from ego's nearest wheel to drivable surface boundary (positive inside, negative outside), and p is exponent.

Rule r_{16} (Level 3) - Obey speed limits Penalizes exceeding posted speed limits:

$$V_{16} = \int_0^T \left[\max\{0, v(t) - v_{\text{limit}}(t)\}\right]^p \mathrm{d}t$$

TABLE III: Normalized Safety Rule Evaluations

ID	A	B	C	D	E	F	G	H	I
L9: Critical Collision Avoidance									
r_1	1.00	1.00	1.00	**0.2689**	**0.2689**	**0.2689**	0.27	**0.2689**	1.00
L7: High-Priority Operational Safety									
r_3	0.38	0.38	0.38	**0.3775**	1.00	0.93	1.00	1.00	0.38
L3: Dynamic Safety & Compliance									
r_{16}	0.27	0.27	0.27	**1.00**	0.27	0.27	0.99	0.27	0.27
r_{17}	1.00	1.00	1.00	**0.2689**	1.00	1.00	0.27	1.00	1.00

where $v(t)$ is ego speed, $v_{\text{limit}}(t)$ is posted speed limit, and p is exponent.

Rule r_{17} (Level 3) - Maintain safe following headway Penalizes insufficient time headway to lead vehicle:

$$V_{17} = \int_0^T \left[\max\{0, \text{THW}_{\min} - d_k(t)/v(t)\}\right]^p \mathbb{I}_{\text{follow},k}(t) \mathrm{d}t$$

where $d_k(t)$ is gap to lead vehicle k, $\text{THW}_{\min}$ is minimum progress time, and p is exponent.

2) Normalization Logic

Raw violation V_i normalizes to $V_{\text{norm},i} \in [0, 1]$ via sigmoid $V_{\text{norm},i} = 1/(1 + e^{-\alpha_i(V_i - \tau_i)})$, providing continuously differentiable score for gradient-based optimization and mapping violations with different units to common scale. Parameters requiring manual calibration:

- **Severity Threshold** (τ_i): Raw score for half-maximal normalized violation (0.5), set from empirical data, regulatory limits, or comfort benchmarks.
- **Steepness Parameter** (α_i): Controls transition sharpness; raw score $\tau_i \pm 1/\alpha_i$ yields normalized violation ≈ 0.73 or 0.27.

The continuous bounded metric with lexicographic slack strategy enables graceful degradation under conflicts, finding minimum-violation solutions by selectively relaxing lower-priority rules.

Selection proceeds by lexicographically comparing violation vectors, level by level:

1) **L10** (r_0): No VRUs present;
2) **L9** (r_1): Trajectories **A, B, C, I** result in collision (score 1.0000). Trajectories **D, E, F, G, H** avoid collisions (score 0.2689). A, B, C, I eliminated.
3) **L8** (r_2): Not applicable. D, E, F, G, H proceed.
4) **L7** (r_3): Trajectory **D** remains in lane ($y \leq 1.85\,\text{m}$), minimal score. **E, F, G, H** involve lateral maneuvers crossing shoulder (high violations). E, F, G, H eliminated.

5) **Outcome**: Only **Trajectory D** remains.

Selection of **trajectory D (accelerate)** demonstrates the framework's ability to navigate complex trade-offs. Though counterintuitive when a lead vehicle slows, it is the only candidate satisfying highest-priority safety rules.

- **Primacy of Collision Avoidance**: Trajectories with braking (B, C, I) correctly identified as causing collision with tailgater or drifter, eliminated at highest safety level (L9).
- **Adherence to Operational Boundaries**: Lateral evasion trajectories (E, F, G, H) eliminated at L7 for leaving drivable surface. The system correctly judged leaving roadway unacceptable when compliant alternative exists.
- **Acceptable Lower-Level Violations**: Trajectory D violates speed limit (r_{16}) and dynamic limits (r_{19}) at lower priority (L3). Lexicographical process correctly determines accepting L3 violations preferable to catastrophic L9 (collision) or high-risk L7 (off-road) violation.

VI. Acknowledgment

The rules outlined in this document are defined in strict mathematical, rather than legal, context.

VII. Conclusion and Future Work

This work introduces a robust, auditable, and verifiable rulebook for autonomous-vehicle planning that resolves real-time rule conflicts by encoding regulations as a prioritized hierarchy of differentiable violation metrics. Its theoretical core couples asymptotic lexicographic optimization with robust control barrier functions to guarantee safety under bounded uncertainty. By yielding formal guarantees and auditable decision artifacts, the framework serves as a governance layer for learning-based systems, addressing the verification gap and meeting transparency requirements. We extend the approach to formalize rules under uncertainty, govern learned components, and automate rule creation for scalability. Future work targets (i) Online Rule Adaptation—learning or adjusting priorities in real time via inverse reinforcement learning on human driving data, (ii) Scalable Formal Synthesis—automating rulebook generation from regulatory text using LLMs to produce verifiable code, and (iii) multi-agent/game-theoretic extensions for cooperative decision making. Meta-rules can also govern learned modules by penalizing low perception confidence or constraining a policy network's Lipschitz constant [1], which requires augmenting the system state x with internal learner states so TORQ can jointly verify physical and computational behavior. To overcome manual codification bottlenecks, LLM-based pipelines (e.g., TR2MTL [2]) can translate legal text into formal specifications—improving scalability while introducing the challenge of verifying translation accuracy.

Our use of continuously differentiable violation metrics is advantageous over alternatives like Metric Temporal Logic (MTL) [13] or Defeasible Deontic Logic [14] due to inherent compatibility with gradient-based optimizers (e.g., SQP). The current framework's assumption of perfect state knowledge can be addressed by redefining metrics to handle state uncertainty, which increases computational complexity. This can be achieved through:

- **Stochastic Violation Metrics:** For a state distribution $p(x)$, the metric is its expected value: $\rho_r^{\text{stoch}}(\xi) = \mathbb{E}_{x\sim p(x)}[\rho_r(\xi, x)]$.
- **Robust Violation Metrics:** For a set of uncertainty in the bounded state $x \in \mathcal{X}_{\text{uncert}}$, the metric is the worst-case violation: $\rho_r^{\text{robust}}(\xi) = \max_{x\in\mathcal{X}_{\text{uncert}}} \rho_r(\xi, x)$.

These formulations allow the optimization framework to reason about and minimize violations robustly against sensor noise and estimation errors, providing a clear path toward real-world applicability.

References

[1] N. Mehdipour, M. Althoff, R. D. Tebbens, and C. Belta, "Formal methods to comply with rules of the road in autonomous driving: State of the art and grand challenges," *Automatica*, vol. 152, p. 110692, 2023.

[2] J. Collenette, L. A. Dennis, and M. Fisher, "Advising autonomous cars about the rules of the road," *Electronic Proceedings in Theoretical Computer Science*, vol. 371, p. 62–76, Sep. 2022.

[3] K. Esterle, L. Gressenbuch, and A. Knoll, "Formalizing traffic rules for machine interpretability," in *2020 IEEE 3rd Connected and Automated Vehicles Symposium (CAVS)*, 2020, pp. 1–7.

[4] A. Censi, K. Slutsky, T. Wongpiromsarn, D. Yershov, S. Pendleton, J. Fu, and E. Frazzoli, "Liability, ethics, and culture-aware behavior specification using rulebooks," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8536–8542.

[5] W. Xiao, N. Mehdipour, A. Collin, A. Y. Bin-Nun, E. Frazzoli, R. D. Tebbens, and C. Belta, "Rule-based optimal control for autonomous driving," in *Proceedings of the ACM/IEEE 12th International Conference on Cyber-Physical Systems*, ser. ICCPS '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 143–154. [Online]. Available: https://doi.org/10.1145/3450267.3450542

[6] W. Liu, S. Alsalehi, N. Mehdipour, E. Bartocci, and C. Belta, "Quantifying the satisfaction of spatio-temporal logic specifications for multi-agent control," *IEEE Transactions on Automatic Control*, pp. 1–16, 2025.

[7] A. Rizaldi, J. Keinholz, M. Huber, J. Feldle, F. Immler, M. Althoff, E. Hilgendorf, and T. Nipkow, "Formalising and monitoring traffic rules for autonomous vehicles in isabelle/hol," in *Integrated Formal Methods*, N. Polikarpova and S. Schneider, Eds. Cham: Springer International Publishing, 2017, pp. 50–66.

[8] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *2016 American Control Conference (ACC)*, 2016, pp. 322–328.

[9] W. Xiao, C. A. Belta, and C. G. Cassandras, "Sufficient conditions for feasibility of optimal control problems using control barrier functions," *Automatica*, vol. 135, p. 109960, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109821004866

[10] W. Xiao, C. A.Belta, and C. G. Cassandras, "Feasibility-guided learning for constrained optimal control problems," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 1896–1901.

[11] J. Plaum, A. Tejada, J. Günther, and E. Sax, "Toward the definition of competent driving for the assessment of automated driving systems."

[12] E. Daş and J. W. Burdick, "Robust control barrier functions using uncertainty estimation with application to mobile robots," *IEEE Transactions on Automatic Control*, vol. 70, no. 7, pp. 4766–4773, 2025.

[13] K. Manas, S. Zwicklbauer, and A. Paschke, "Tr2mtl: Llm based framework for metric temporal logic formalization of traffic rules," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, 2024, pp. 1206–1213.

[14] H. Bhuiyan, G. Governatori, A. Bond, S. Demmel, M. Badiul Islam, and A. Rakotonirainy, "Traffic rules encoding using defeasible deontic logic," in *Legal Knowledge and Information Systems*. IOS Press, 2020, pp. 3–12.