

Traffic Rule Compliant Autonomous Driving Using Reward Machines and Reinforcement Learning

Sicong Guo, Mohamad Louai Shehab, Alperen Tercan, Ruya Karagulle, and Necmiye Ozay

Abstract—Many real-world autonomous scenarios require compliance with history-dependent traffic behaviors, whose correct execution depend on sequences of past events rather than the current state alone, such as stop-and-go, first-come-first-served priority, and right-of-way yielding at unsignalized intersections. Standard Deep Reinforcement Learning approaches assume the Markov property, making them unsuitable to capture this history dependence. To that end, we propose an reward-machine-based framework for non-Markovian traffic rule compliance that encodes stop-and-go and first-come-first-served behaviors as Reward Machines. We develop a safety-aware labeling function and an reward machine reward specification that exposes intermediate task structure from continuous traffic states. Experiments in highway-env (stop sign) and SUMO (all-way unsignalized intersection) show near-100% rule compliance with strong safety and stopping precision, performing significantly better than two baseline approaches.

I. INTRODUCTION

In recent years, autonomous vehicles (AVs), especially robotaxis, have demonstrated promising performance in structured environments. Yet, unstructured scenarios, where right-of-way must be negotiated instead of enforced by traffic signals, are challenging for these systems. For example, Waymo, a leading robotaxi company, acknowledged that its fleet struggled at unsignalized intersections during a city-wide blackout in San Francisco [1]. These challenges are not unique to AVs. In fact, unsignalized intersections are challenging for human drivers as well. Data from the U.S. National Highway Traffic Safety Administration indicates that approximately 40% of all vehicular crashes happen at intersections [2], and the Federal Highway Administration reports that 70% of fatalities resulting from intersection-related collisions occur specifically at unsignalized junctions [3]. Moreover, when AVs and human drivers interact in such environments, one study finds that AV’s conservative behavior can create unpredictable interactions, potentially increasing risk for surrounding human drivers [4]. To operate successfully in these scenarios, AVs must adhere to socially grounded traffic behaviors such as first-come-first-served (FCFS), zipper merges, stop-and-go, and right-of-way yielding. What makes these behaviors particularly difficult is their history-dependent nature. Regulations like FCFS [5] require monitoring the order of vehicles arriving at the intersection and reacting based on that order. Therefore, a

decision cannot be made by using only the current state information.

Several approaches have been explored to enable traffic rule compliance in history-dependent scenarios. For autonomous navigation at urban intersections, Deep Reinforcement Learning (DRL) has been widely adopted [6]–[9]. Recent advancements at unsignalized intersections have further improved learning performance through techniques such as curriculum learning, hierarchical architectures with low-level MPC, mixed-traffic PPO, and multi-agent coordination [10]–[14]. However, these methods typically leave task phases implicit within handcrafted reward conditions rather than representing them as explicit, temporally structured states [15]. Consequently, they fail to account for history-dependent temporal conventions. Furthermore, Standard DRL assumes the Markov property, fundamentally conflicting with history-dependent traffic interactions. To address this, temporal logics—such as Linear Temporal Logic (LTL) and Signal Temporal Logic (STL)—are used to specify and verify traffic regulations over sequences of events [16], [17]. These frameworks excel at expressing complex intersection scenarios [16], [18], [19], synthesizing control tasks [20], and learning from human feedback [21]. However, pure formal methods heavily rely on explicit environmental model descriptions.

To circumvent this, hybrid approaches combine formal methods with RL to leverage both temporal logic’s expressiveness and DRL’s model-free nature [22]–[25]. Yet, a major bottleneck remains: compiling LTL formulas into a binary satisfaction signal yields sparse rewards, preventing the agent from exploiting intermediate task structures and severely limiting sample efficiency [23], [24]. For example, authors in [25] acknowledge that history structure may result in exponential state-space growth for the learning problem, and suggest a more compact approach to capture state history in a new augmented state-space representation, by proposing an approximation to the objective value.

To overcome these limitations and enforce traffic rules in history-dependent scenarios while preserving the advantages of end-to-end DRL, we propose using and designing Reward Machines (RMs) [23] consistent with these rules. RMs are finite-state machines that explicitly represent the logical structure underlying complex reward functions. They enable an agent to track its progress through multi-step behavioral sequences by encoding task-relevant events in a structured, state-based form. By combining this RM with the underlying MDP, we transform the environment into a Product MDP, recovering the Markov property. This leads to optimized

This work is supported in part by NSF Grant TI-2303564. Mohamad Louai Shehab and Necmiye Ozay are with the Robotics Department, University of Michigan, Ann Arbor. Sicong Guo, Alperen Tercan, Ruya Karagulle, and Necmiye Ozay are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA.

policies in the product space which, by construction, are incentivized to satisfy the encoded rules. While prior work has applied RMs to highway driving scenarios [26], these tasks are essentially Markovian in nature. In contrast, our work targets non-Markovian compliance at unsignalized intersections, where the agent must track temporal histories such as vehicle arrival order to determine correct behavior. The main contributions of this paper are as follows:

- RM-based formulation for non-Markovian traffic compliance: We model history-dependent traffic rules (stop-and-go and FCFS at unsignalized intersections) using Reward Machines and synchronize them with the underlying MDP to obtain a Markovian product MDP that can be optimized with standard DRL (PPO).
- Safety-aware labeling and reward design: We design a labeling function that extracts task-relevant events (e.g., compliance, stopping, passing, priority) from continuous traffic states, grounding safety/stop enforcement using the Responsibility-Sensitive Safety (RSS) Model [27], and provide an RM reward specification that exposes intermediate task structure.
- Empirical validation on two simulators with strong compliance gains: We demonstrate in highway-env (stop sign) and SUMO (all-way unsignalized intersection) that the RM framework achieves near-100% rule compliance while maintaining safety and good stopping precision, whereas scalar reward-shaping baselines collapse to degenerate (overly defensive) behaviors with near-zero compliance.

II. PRELIMINARIES

A. Markov Decision Process (MDP)

A Markov decision process (MDP) is a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mu_0, \gamma, r)$ with a state space $\mathcal{S}$, a finite action space $\mathcal{A}$, a Markovian transition kernel that admits a conditional density $\mathcal{P}(s' | s, a)$ over next states $s' \in \mathcal{S}$ given the current state $s \in \mathcal{S}$ and action $a \in \mathcal{A}$, an initial state distribution μ_0 over $\mathcal{S}$, a discount factor $\gamma \in [0, 1)$, and a reward function $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$. We use $\mathcal{M}/r$ to denote all components of $\mathcal{M}$ except r . In this work, the transition kernel and the initial state distribution are unknown but we assume to have access to a simulation environment instead of the detailed model.

B. Reward Machines (RM)

A *reward machine* (RM) is a finite-state Mealy machine that compactly represents history-dependent (non-Markovian) rewards by maintaining a discrete *memory state*. Formally, we write $\mathcal{R} = (\mathcal{U}, u_0, \text{AP}, \delta_u, \delta_r)$, which consists of a finite set of RM states $\mathcal{U}$, an initial state $u_0 \in \mathcal{U}$, a set of atomic propositions AP, a deterministic transition function $\delta_u : \mathcal{U} \times \text{AP} \rightarrow \mathcal{U}$, and an output reward function $\delta_r : \mathcal{U} \times \text{AP} \rightarrow [\mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}]$. At each environment step, a labeling function outputs a logical *event* $\ell \in 2^{\text{AP}}$. The transition function δ_u updates the RM state based on the observed event, and the output reward function δ_r returns the corresponding RM reward. Thus, the reward at time t

can depend on the full history merely through the current RM state $u_t = \delta_u^*(u_0, \ell_0, \ell_1, \dots, \ell_t)$, where δ_u^* denotes the standard extension of δ_u to event strings.

C. MDP-RM and Product MDP

A Markov decision process with a reward machine (MDP-RM) is a tuple $\mathcal{R}_{\mathcal{M}} = (\mathcal{M}/r, \mathcal{R}, L)$ where $\mathcal{M}$ and $\mathcal{R}$ are defined as above, and L is a labeling function as $L : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow 2^{\text{AP}}$. The labeling function assigns to each (s, a, s') a subset of atomic propositions, representing the high-level events which are true at this transition.

An MDP-RM $\mathcal{R}_{\mathcal{M}}$ can be equivalently seen as a product MDP $\mathcal{M}' = (\mathcal{S}', \mathcal{A}', \mathcal{P}', \mu'_0, \gamma', r')$ where $\mathcal{S}' = \mathcal{S} \times \mathcal{U}$, $\mathcal{A}' = \mathcal{A}$, $\mathcal{P}'(s', u' | s, u, a) = \mathcal{P}(s' | s, a) \mathbf{1}(u' = \delta_u(u, L(s')))$, $\gamma' = \gamma$, μ'_0 is the initial distribution over $\mathcal{S} \times \mathcal{U}$ with $\mu'_0(s, u) = \mu_0(s) \mathbf{1}(u = u_I)$ and $r'(s, u, a, s', u') = \delta_r(u, L(s'))(s, a, s')$. At each time step, the agent observes the product state $s^\times = (s, u)$, and executes an action $a \in \mathcal{A}$. The MDP state then transitions to $s' \sim \mathcal{P}(\cdot | s, a)$, and the RM node transitions to $u' = \delta_u(u, L(s'))$. As a result, the product MDP state transitions to $s^{\times, \prime} = (s', u')$. A policy $\pi : \mathcal{S}' \rightarrow \Delta(\mathcal{A}')$ of $\mathcal{M}'$ maps joint states to a distribution over actions.

Note that only the product state space is constructed explicitly; the remaining components of the product MDP, such as the product transition function, are not computed explicitly.

D. Proximal Policy Optimization (PPO)

After constructing the product MDP, our goal is to find a policy that maximizes the regular infinite horizon RL objective. We define the RL objective as:

$$J(\pi, r) = \mathbb{E}_{\substack{a_t \sim \pi(s_t^\times) \\ s_{t+1}^\times \sim \mathcal{P}'(\cdot | s_t^\times, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t r'(s_t^\times, a_t, s_{t+1}^\times) \right]. \quad (\text{II.1})$$

Our goal is to find a policy that maximizes the objective as:

$$\pi^* = \arg \max_{\pi} J(\pi; r) \quad (\text{II.2})$$

The optimization of the autonomous agent's control policy is carried out using Proximal Policy Optimization (PPO) in $\mathcal{M}'$. PPO is a widely adopted first-order deep reinforcement learning algorithm that promotes stable and reliable policy improvement by limiting overly large policy updates, thereby balancing learning efficiency and robustness.

We parameterize the stochastic policy, termed the Actor, utilizing a fully connected Multi-Layer Perceptron (MLP) defined by weights θ , denoted as $\pi_\theta(s_t^\times)$. Since our framework employs a discrete action space, the Actor MLP maps the high-dimensional joint state vector to categorical probabilities using a softmax function, from which the discrete control actions are stochastically sampled. Concurrently, a Critic MLP network, parameterized by ϕ , approximates the state-value function baseline $V_\phi(s_t^\times)$.

III. METHODOLOGY

In this section, we present our proposed labeling function and reward machine design used for incorporating non-Markovian traffic rules into RL policies. For our formulation throughout this section, we consider the state vector as:

$$\mathbf{s} = [x_e, y_e, \mathbf{x}^\top, \mathbf{y}^\top, v_e, v_s, \psi_e, \boldsymbol{\psi}^\top, h, d_{\text{stop}}, \mathbf{t}_{\text{arrival}}^\top, \mathbf{t}_{\text{wait}}^\top]^\top, \quad (\text{III.1})$$

where (x_e, y_e) and $(\mathbf{x}, \mathbf{y})$ are the position coordinates of the ego vehicle and all the surrounding vehicles respectively. The longitudinal velocity of the ego vehicle and the lead vehicle are v_e and v_s respectively. The heading angles of the ego vehicle and all of the surrounding vehicles are ψ_e and $\boldsymbol{\psi}$ respectively. The headway of the ego vehicle is h . The distance of the ego vehicle to the stop sign is d_{stop} . The time vectors for registered arrivals and waiting at the intersection for all vehicles are denoted as $\mathbf{t}_{\text{arrival}}$ and $\mathbf{t}_{\text{wait}}$, respectively.

A. Labeling Function

The labeling function $L : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow 2^{\text{AP}}$ is designed to capture the high-level events (i.e., atomic propositions) that are relevant to the traffic rules being enforced. Since the focus of this work is on stop-and-go behavior at stop signs and interactions at unsignalized intersections, the atomic propositions are designed to encode events related to safety, stopping compliance, and right-of-way.

a) *Safety and stop enforcement*: We formulate the longitudinal safety for the ego vehicle via the Responsibility-Sensitive Safety (RSS) Model [27]. For autonomous driving to be free from an accident caused by an autonomous vehicle, the longitudinal safety headway is determined as a function of v_e and v_s as follows:

$$d_{\text{RSS}}(v_e, v_s) = v_e \rho + \frac{1}{2} a_{\text{max,acc}}^e (\rho)^2 + \frac{(v_e + a_{\text{max,acc}}^e \rho)^2}{2|a_{\text{conf,dec}}^e|} - \frac{v_s^2}{2|a_{\text{max,dec}}^s|},$$

where ρ is the response (i.e., reaction) time of the ego vehicle, $a_{\text{max,acc}}^e$ and $a_{\text{conf,dec}}^e$ are the maximum allowable acceleration and the maximum comfortable deceleration of the ego vehicle, respectively. $a_{\text{max,dec}}^s$ is the maximum allowable deceleration of the lead vehicle. By this formulation, the longitudinal car-following safety on the ego vehicle is met as long as:

$$h \geq d_{\text{RSS}}(v_e, v_s).$$

Since both safety and stopping compliance depend on distances to a preceding object (the lead vehicle for safety and the stop sign for stop compliance), we can define distance to this object and its velocity, $(d_{\text{in}}, v_{\text{in}})$ in a piecewise manner:

$$(d_{\text{in}}, v_{\text{in}}) = \begin{cases} (h, v_s) & \text{if } (d_{\text{reg}} \leq d_{\text{stop}}) \wedge (h < d_{\text{stop}}) \\ (d_{\text{stop}}, 0) & \text{if } (d_{\text{reg}} \leq d_{\text{stop}}) \wedge (h \geq d_{\text{stop}}) \\ (d_{\text{stop}}, 0) & \text{if } (0 \leq d_{\text{stop}} < d_{\text{reg}}) \\ (h, v_s) & \text{if } (d_{\text{stop}} < 0) \end{cases},$$

where $d_{\text{reg}} = 5\text{m}$ is defined and set as the registration threshold for the stop sign at the intersection. As a result, we denote the combined compliance of the longitudinal car-following safety and stop enforcement as,

$$\text{comp} := (d_{\text{in}} \geq d_{\text{RSS}}(v_e, v_{\text{in}})) \wedge (|v_e| \geq 0.1).$$

We also denote the deviation from the combined compliance as,

$$\delta_{\text{comp}} := d_{\text{in}} - d_{\text{RSS}}(v_e, v_{\text{in}}).$$

b) *Intersection passing priority*: For each vehicle i arriving at the intersection, we define its arrival time $t_{\text{arrival},i}$ as the time when it gets within 5m from the stop sign, and its waiting time $t_{\text{wait},i}$ as the time elapsed since it stopped (i.e., $|v_i| \leq 0.1$) after its arrival. Vehicle priority at the intersection is determined on a first-come first-served basis. Let $\mathcal{V}_{\text{wait}}$ denote the set of vehicles which have arrived and are waiting at the intersection. The priority [5] to pass the intersection is granted for a vehicle i if the following three conditions hold.

- 1) It has the longest waiting time, i.e.,

$$\bigwedge_{j \in \mathcal{V}_{\text{wait}} \setminus \{i\}} t_{\text{wait},i} \geq t_{\text{wait},j} + \epsilon, \quad (\text{III.2})$$

with ϵ being a tolerance (normally equal to one simulation timestep).

- 2) It has the earliest arrival time if waiting times are equal, i.e.,

$$\bigwedge_{\substack{|t_{\text{wait},i} - t_{\text{wait},j}| < \epsilon, \\ j \in \mathcal{V}_{\text{wait}} \setminus \{i\}}} t_{\text{arrival},i} \leq t_{\text{arrival},j} - \epsilon. \quad (\text{III.3})$$

- 3) It is on the right of or is oncoming towards the other left-turning vehicle if both waiting and arrival times are equal. Let $\text{dir}_{ij} = (\Delta \mathbf{p}_{ij}^\top \hat{\mathbf{e}}_i > 0) \vee (\psi_i - \psi_j > \frac{3}{4}\pi)$, where $\Delta \mathbf{p}_{ij} = [x_j - x_i \ y_j - y_i]^\top$ is the relative position between vehicles i and j , and $\hat{\mathbf{e}}_i = [\cos \psi_i \ \sin \psi_i]^\top$ is the heading vector of vehicle i . Then,

$$\bigwedge_{\substack{(|t_{\text{wait},i} - t_{\text{wait},j}| < \epsilon) \wedge (|t_{\text{arrival},i} - t_{\text{arrival},j}| < \epsilon) \\ j \in \mathcal{V}_{\text{wait}} \setminus \{i\}}} \text{dir}_{ij}. \quad (\text{III.4})$$

Formally, the right of way (row) is defined as:

$$\text{row} \iff (\text{III.2}) \vee (\text{III.3}) \vee (\text{III.4}).$$

Overall, the labeling function returns the following labels when corresponding conditions are met:

- X: True if a collision occurs on the ego vehicle.
- comp: True if the combined compliance of the longitudinal car-following safety and stop enforcement is met.
- S: True if the ego vehicle achieves registered stop at the intersection: $(0 \leq d_{\text{stop}} \leq d_{\text{reg}}) \wedge (|v_e| < 0.1) \wedge (d'_{\text{stop}} \geq 0)$.
- P: True if $(d'_{\text{stop}} < 0) \wedge (d_{\text{stop}} \geq 0)$. This label is returned at the exact time step when the stop is passed,

noting that primed variables denote the value at the next time step.

- row: True if the ego vehicle has the right of way.
- G: True if the ego vehicle reaches 20m past the intersection.

B. Reward Machine Design

We propose a reward machine design for unsignalized intersection as shown in Fig. 1. This reward machine is composed of the states:

- u_0 denoting that the ego vehicle maintains the combined compliance of car-following safety and stop enforcement; and is being rewarded for higher ego vehicle speed for improved traffic throughput efficiency,
- u_1 denoting that the ego vehicle fails the combined compliance and initiates to recover it,
- u_2 denoting that the ego vehicle comes to a full stop at the intersection and registered for its priority to pass,
- u_3 denoting that the ego vehicle is granted with the priority and initiates to pass,
- u_{-1} denoting the terminal state, reached in the case of a crash, failing to stop, or successful completion.

The reward function is specified in a piecewise manner, where the constant components are shown in Fig. 1, and the variable components associated with safety compliance and stop enforcement are as defined below:

$$\delta_r(u_0, \text{comp})(s, a, s') = \text{sat}\left(1 - \frac{\delta_{\text{comp}}}{2}\right) + \mathbb{I}(d_{\text{stop}} \geq d_{\text{reg}}) \cdot F(d'_{\text{stop}}, d_{\text{stop}}) \quad (\text{III.5})$$

$$\delta_r(u_1, \neg \text{comp})(s, a, s') = \frac{1}{5} \delta_{\text{comp}}, \quad (\text{III.6})$$

where $\text{sat}(1 - \frac{\delta_{\text{comp}}}{2})$ is the saturation function that encodes safety with respect to the RSS model, $\mathbb{I}(d_{\text{stop}} \geq d_{\text{reg}})$ is the indicator function, and $F(d'_{\text{stop}}, d_{\text{stop}})$ encourages progress towards the stop sign. Specifically,

$$F(d'_{\text{stop}}, d_{\text{stop}}) = -\frac{\gamma}{2} d'_{\text{stop}} + \frac{1}{2} d_{\text{stop}}.$$

IV. SIMULATION STUDIES

To validate the efficacy of our RM design, we conducted simulation studies on a stop-sign scenario using highway-env [28], and unsignalized intersection using traffic simulation package SUMO [29]. In both cases, the simulations provide the continuous state, from which we extract the corresponding labels to update the reward machine state. The product state is constructed from the continuous state and reward machine state, accordingly, and used in PPO. Our code is available on GitHub¹.

A. Simulation setup

a) Toy case: Stopping at a stop sign: In this simulation study (using highway-env [28]), we define a single-lane driving scenario with a stop sign, as shown in Fig. 2. This

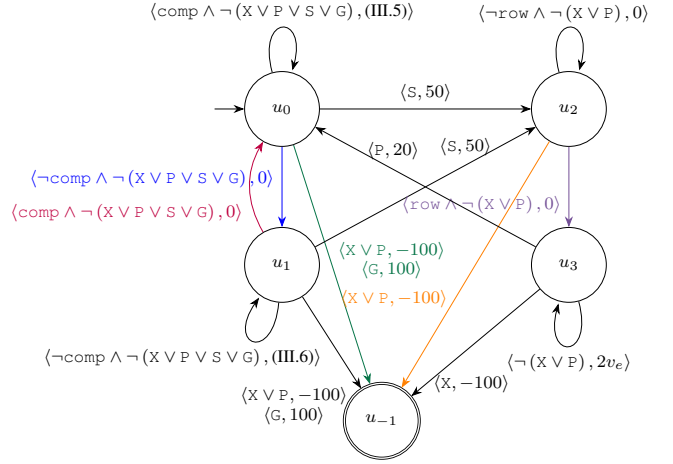


Fig. 1. Reward machine design for unsignalized intersection with first-come-first-serve rule compliance. For each transition, the first entry in brackets represents the condition, and the second entry represents the reward. For some transitions, rewards are specified by equation numbers for brevity. This design can also be used for scenarios free from right-of-way constraints, such as maintain safety and stopping at the stop sign, by setting $\text{row} = \text{True}$ to bypass the yielding state transitions.

is a toy case in which the ego vehicle is required to stop at the stop sign, but the surrounding vehicles are not bounded by this enforcement and maintain their default car-following manner. The stop sign is at 150m ahead of the initial position of the ego vehicle, and the goal is set as 20m beyond the stop sign. Each vehicle has a uniform length of 5m.

The ego vehicle operates with a discrete action space $\mathcal{A} = \{0, 1, 2\}$, mapping to target speed adjustments of $\{0, 15, 30\}$ m/s which is then realized via a low-level proportional controller. Surrounding vehicles are governed by the Intelligent Driver Model (IDM) [30]. For all vehicles, the vehicle dynamics are governed by the Kinematic Bicycle Model [31].

The simulation runs at a timestep of 0.067s (15Hz) with a policy update period of 0.2s. The ego vehicle has a speed range of $[0, 30]$ m/s, is initialized with a speed of 20 m/s, and a random headway sampled uniformly from $[26.2, 34.9]$ m. The maximum allowable acceleration for the ego vehicle is set as $a_{\text{max,acc}} = 5 \text{ m/s}^2$. Both the ego and surrounding vehicles have uniform maximum comfortable deceleration and the maximum allowable deceleration set as $a_{\text{comf,dec}} = -5 \text{ m/s}^2$ and $a_{\text{max,dec}} = -6 \text{ m/s}^2$, respectively. The RSS response time ρ is set as 0.2s.

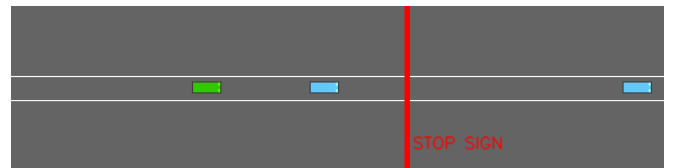


Fig. 2. The toy case for stopping at a stop sign, built by modifying the highway-env simulator [28]. Ego vehicle is shown in green, and the surrounding vehicles are shown in blue.

¹<https://tinyurl.com/4vjjrwn5>

b) Unsignalized intersection: In this simulation study, we construct a four-way unsignalized intersection using SUMO [29]. Each vehicle as a uniform length of 5m. The geometry of this intersection environment comprises four 100-meter single-lane edges converging on a 14.4×14.4 m² intersection, with stop signs 7.2m from the center. The ego vehicle operates with a discrete action space $\mathcal{A} = \{0, 1, 2\}$, mapping to acceleration commands for “decelerating” ($a_{\text{cmd.}} = -5$ m/s²), “maintaining speed” ($a_{\text{cmd.}} = 0$ m/s²), and “accelerating” ($a_{\text{cmd.}} = 5$ m/s²). Surrounding vehicles are governed by the Krauss stochastic car-following model [32]. Enforced stopping at the intersection and right-of-way conventions (defined as (III.2)–(III.4)) are encoded on all surrounding vehicles by default in SUMO.

Eight vehicles including the ego are initially-spawned in the layout as shown in Fig. 3, with the ego vehicle colored in green. The maximum comfortable deceleration and the maximum allowable deceleration are identical for all vehicles, i.e., $a_{\text{comf,dec}} = a_{\text{max,dec}} = a_{\text{dec}}$. All information about the vehicle setups are shown in Table I. Note that the initial position of the ego vehicle is randomized in the range of [69, 87]m to simulate random intersection arrival orders. The simulation runs at a timestep of 0.1s with a policy update period of 0.2s. The RSS response time ρ is set as 0.2s.

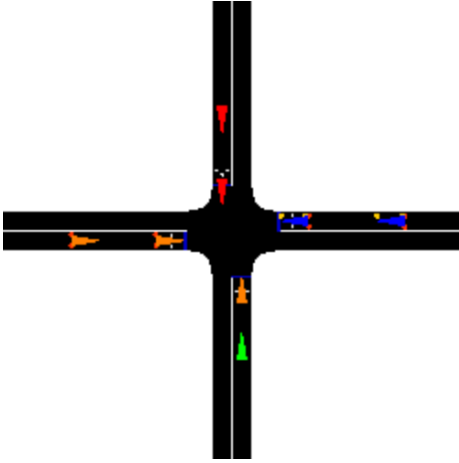


Fig. 3. The unsignalized intersection environment using traffic simulation package SUMO. Ego vehicle is shown in green.

c) Baseline and training hyperparameters: To evaluate the performance of our developed framework, we compare it against a baseline approach (B1), employing a standard reinforcement learning agent trained with a reward function R_{total} using an internal memory flag $b_{\text{stop}} \in \{0, 1\}$ with $b_{\text{stop}} = 0$ as not stopped and $b_{\text{stop}} = 1$ as stopped. The reward function is given by:

$$R_{\text{total}} = R_{\text{viol}} + (1 - b_{\text{stop}})R_{\text{appr}} + b_{\text{stop}}R_{\text{depart}} + R_{\text{task}} \quad (\text{IV.1})$$

Specifically, the agent receives a severe terminal penalty $R_{\text{viol}} = -100$ for safety violations, such as colliding or failing to stop at the stop sign. Before stopping, an approach reward R_{appr} based on speed of the ego vehicle encourages

TABLE I
VEHICLE PARAMETERS, INITIAL CONDITIONS (v_0 , $d_{\text{stop},0}$), AND LANE SETUPS (THE INITIAL LANE AS “S” AND THE TARGET LANE AS “G”) FOR THE FOUR-WAY UNSIGNALIZED INTERSECTION STUDY. (UNITS: DISTANCE: M; SPEED: M/S; ACCELERATION: M/S².)

ID	v_0	$d_{\text{stop},0}$	$a_{\text{max,acc}}$	a_{dec}	v_{max}	S-G
Red 1	10	75	3	−5	20	N-S
Red 2	10	55				
Orange 1	9	90	2.6	−4.5	20	W-E
Orange 2	9	70				
Orange 3	9	60	2.6	−4.5	20	S-N
Blue 1	8	85				
Blue 2	8	65	2	−4	15	E-N
Green (ego)	10	[69, 87]				
			5	−5	20	S-N

safe car-following and smooth deceleration as the vehicle nears the intersection. Upon successfully stopping at the correct location, the agent earns a sparse task bonus $R_{\text{task}} = 20$, which triggers the internal “stopped” flag. Finally, a departure reward R_{depart} based on the speed of the ego vehicle incentivizes continued safe car-following as the agent leaves the stop sign and proceeds toward the goal.

As another baseline approach (B2), we employ an LSTM-PPO agent [33], [34]. The baseline uses the same continuous traffic states (III.1), action space, and scalar reward components, while augmenting the policy with recurrent memory. Specifically, the LSTM compresses the state history into a hidden state, which is then used by the PPO actor-critic networks for action selection and value estimation.

Similar to (IV.1), the reward function used for this baseline is an addition of: $R_{\text{viol}} = -100$, $R_{\text{task}} = 50$, a RSS-based reward term for safety car-following as (III.5)–(III.6), a reward term for proceeding to pass the intersection with $2v_e$ conditioned on b_{stop} and row , and a terminal reward $R_{\text{goal}} = 100$ for reaching the goal.

As a result, B2 serves as a stronger scalar-reward baseline: it tests whether recurrent memory alone can recover the temporal stop–yield–proceed hierarchy.

B. Evaluation Metrics

To evaluate the performance of our developed framework, we introduce five metrics:

Safety measures the number of safe episodes (i.e. without collisions):

$$M_{\text{safe}} = \left(1 - \frac{n_{\text{collision}}}{n_{\text{total}}}\right) \times 100\%,$$

where $n_{\text{collision}}$ is the number of episodes in which the ego vehicle encounters a collision and n_{total} is the total number of test episodes.

Stop-sign compliance measures the number of episodes in which the agent stopped at the stop sign, defined as

$$M_{\text{stop}} = \frac{n_{\text{stopped}}}{n_{\text{total}}} \times 100\%,$$

where n_{stopped} is the number of episodes in which the ego vehicle makes a valid stop registered for the intersection.

Yielding compliance measures the number of episodes with yielding compliance at the intersection before being granted with the priority to pass the intersection, defined as

$$M_{\text{yield}} = \left(1 - \frac{n_{\text{viol}}}{n_{\text{total}}}\right) \times 100\%,$$

where n_{viol} is the number of episodes in which the agent enters the intersection without its priority granted.

Completion measures the number of successfully completed episodes, defined as

$$M_{\text{goal}} = \frac{n_{\text{goal}}}{n_{\text{total}}} \times 100\%,$$

where n_{goal} is the number of episodes with the agent fully stops at the stop sign and then proceed to reach the goal.

Stopping precision measures the average distance from where the agent stopped at the stop sign among all test episodes, that is,

$$D_{\text{stop}} = \frac{1}{n_{\text{stopped}}} \sum_{i=1}^{n_{\text{stopped}}} d_{\text{stop},i}(t_{\text{stop},i}),$$

where $t_{\text{stop},i}$ represents the instance at which the agent comes to a full stop (i.e., $|v_e(t_{\text{stop}})| \leq 0.1$ m/s) during i th test episode. Note that the stopping precision is only considered among episodes with stop-sign compliance.

V. RESULTS AND DISCUSSION

For both simulation studies, we train the policies using PPO with an MLP architecture, employing a learning rate of 3×10^{-4} , batch size of 64, discount factor of 0.99, exponential decay coefficient in Generalized Advantage Estimation of 0.95, and clip range of 0.2. We report results from two evaluation protocols: in single-seed tests, each policy is trained for 122880 environment steps using a single random seed, after which we evaluate its performance over $n_{\text{total}} = 100$ test episodes and report the average return; in multi-seed tests, we assess training stability and robustness by repeating the experiment 10 times with different random seeds, training policies for an increasing number of environment steps (up to 122880) and evaluating performance at multiple points during training.

A. Toy case: Stopping at a stop sign

The single-seed test results are shown in Table II. The proposed RM is successful at handling the tasks of enforced stopping at the stop sign and subsequently proceeding. Baselines B1 and B2 maintain safety, but they fail to learn to first stop at the stop sign, then proceed to reach the goal: the ego vehicle is observed to stop further away ($> d_{\text{reg}}$) from the stop sign, and remains stationary instead of proceeding to pass the stop sign.

The multi-seed test results are shown in Fig. 4. The curves demonstrate that the proposed agent eventually learns all objectives, sequentially mastering stop-sign compliance and goal accomplishment while achieving a final stopping

TABLE II
PERFORMANCE COMPARISON BETWEEN THE PROPOSED REWARD MACHINE DESIGN AND TWO BASELINES FOR THE STOP-SIGN SCENARIO.

	M_{safe}	M_{stop}	M_{goal}	D_{stop}
B1	100%	0%	0%	-
B2	100%	0%	0%	-
RM	100%	100%	100%	0.05m

precision of $D_{\text{stop}} = 0.05\text{m}$. In contrast, baselines B1 and B2 fail to progress beyond basic safety metrics, resulting in 0% stop-sign compliance and 0% task completion. Because their performance lines remain entirely flat, they are omitted from the figures.

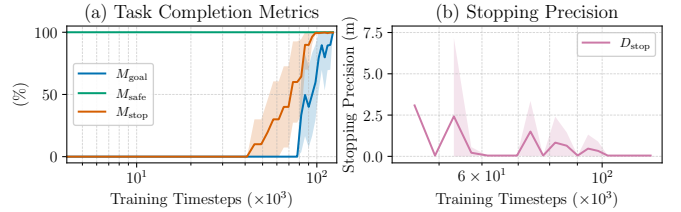


Fig. 4. Training curves on the toy case with the reward machine architecture. (a) Evolution of task completion metrics with increasing training timesteps: safety M_{safe} , enforced-stop compliance M_{stop} , and goal completion M_{goal} . (b) Stopping distance D_{stop} , quantifying the precision of the learned stopping behavior. The solid curves correspond to the mean and the shaded region to the 95% confidence intervals over the 100 test episodes. Subfigure (b) is truncated to start with 45056 training timesteps when the stop-enforcement objective begins to be learned.

B. Unsignalized intersection

The single-seed results are shown in Table III. In this unsignalized intersection environment, the proposed reward machine successfully balances all concurrent tasks: maintaining safety, executing a mandatory stop, yielding to traffic priority, and safely clearing the intersection.

In contrast, the baselines struggle significantly with this multi-stage rule sequence. Baseline B1 fails to progress beyond basic collision avoidance, while B2 yields poor performance across all compliance and goal completion metrics alongside an inaccurate average stopping distance. This underscores that relying on implicit recurrent memory alone is insufficient for agents to reliably infer latent FCFS priorities and the structured stop–yield–proceed phases.

TABLE III
PERFORMANCE COMPARISON BETWEEN THE PROPOSED REWARD MACHINE DESIGN AND TWO BASELINES FOR UNSIGNALIZED INTERSECTION.

	M_{safe}	M_{stop}	M_{yield}	M_{goal}	D_{stop}
B1	100%	0%	0%	0%	-
B2	58%	13%	47%	5%	4.41m
RM	100%	97%	96%	96%	2.29m

The multi-seed training curves for the unsignalized intersection are shown in Fig. 5 and Fig. 6. For our proposed method, all task completion metrics sharply improve alongside a steady refinement in stopping precision. Ultimately, all four performance metrics stably converge near 100% by the end of training. In contrast, baseline B2 struggles to balance these concurrent objectives. It fails to master the stop-sign and goal completion tasks entirely, achieving only modest safety improvements at the expense of partial yielding compliance.

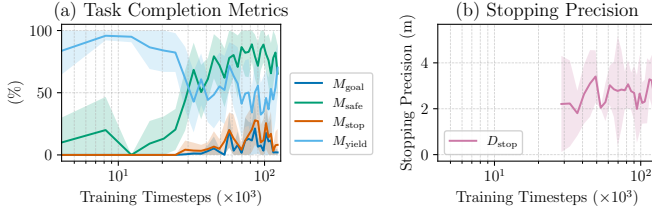


Fig. 5. Training curves on unsignalized intersection with baseline B2. (a) Evolution of task completion metrics with increasing training timesteps: safety M_{safe} , stop-sign compliance M_{stop} , yielding compliance M_{yield} , and goal completion M_{goal} . (b) Stopping distance D_{stop} , quantifying the precision of the learned stopping behavior. The solid curves correspond to the mean and the shaded region to the 95% confidence intervals over 100 test episodes.

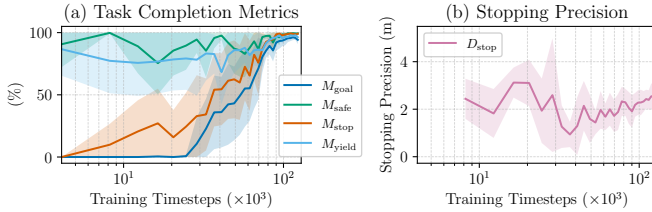


Fig. 6. Training curves on unsignalized intersection with the proposed reward machine design. (a) Evolution of task completion metrics with increasing training timesteps: safety M_{safe} , stop-sign compliance M_{stop} , yielding compliance M_{yield} , and goal completion M_{goal} . (b) Stopping distance D_{stop} , quantifying the precision of the learned stopping behavior. The solid curves correspond to the mean and the shaded region to the 95% confidence intervals over 100 test episodes.

C. Sensitivity to the RSS response time and registration threshold

A sensitivity study over the RSS response time ρ and the registration threshold d_{reg} comparing to the original setting ($\rho = 0.2\text{s}$, $d_{\text{reg}} = 5\text{m}$) is shown in Table IV. The RM policy remains effective across all tested variations, suggesting that the performance is not tied to a single hand-picked constant. Increasing ρ adds the conservativeness of the RSS distance; while both $\rho = 0.3\text{s}$ and $\rho = 0.5\text{s}$ preserve perfect safety in these runs, their compliance and completion rates vary with the induced stopping behavior. Varying d_{reg} from aligning its numerical value with the uniform vehicle length leads to slightly lower compliance rates and better stopping precision, with $d_{\text{reg}} = 3\text{m}$ yielding the best stopping precision and $d_{\text{reg}} = 7\text{m}$ producing the worst compliance rates.

TABLE IV
SENSITIVITY OF THE PROPOSED REWARD MACHINE DESIGN TO THE RSS RESPONSE TIME AND REGISTRATION THRESHOLD FOR THE UNSIGNALIZED INTERSECTION SCENARIO.

(ρ, d_{reg})	M_{safe}	M_{stop}	M_{yield}	M_{goal}	D_{stop}
(0.2, 5)	100%	97%	96%	96%	2.29m
(0.3, 5)	100%	92%	87%	85%	3.02m
(0.5, 5)	100%	100%	99%	99%	2.47m
(0.2, 3)	100%	95%	95%	95%	0.42m
(0.2, 7)	98%	95%	95%	93%	1.97m

VI. CONCLUSION

In this work, we addressed the problem of non-Markovian traffic rule compliance in self-driving applications, where the agent must track temporal histories to determine the correct behavior. We conduct simulation studies on a toy case requiring stopping at a stop sign [28], and on an all-way unsignalized intersection scenario using the traffic simulation package SUMO [29]. We validate the performance of our proposed RM framework in terms of rule compliance, safety, and stopping precision. In contrast to two baselines, our proposed approach achieves nearly a 100% rate of rule compliance in both experiments.

Among the challenges in this work is the careful design of the labeling function, the reward machine, and the associated reward function to ensure reliable performance. This becomes even harder in more complex traffic scenarios, such as lane merges with ambiguous right-of-way. This design burden motivates the need for scaling methods that directly learn reward machines from expert demonstrations [35], [36], thereby reducing manual design effort. Another limitation of the proposed framework is its reliance on access to the exact states of the ego vehicle and surrounding agents. In practical autonomous driving systems, such quantities are typically estimated through perception and state estimation modules and are therefore subject to noise, delay, and partial observability. A promising direction for future work is the development of robust variants of the framework that can operate reliably under state uncertainty.

REFERENCES

- [1] Waymo, “Autonomously navigating the real world: Lessons from the PG&E outage,” <https://waymo.com/blog/2025/12/autonomously-navigating-the-real-world/>, December 2025, accessed: 2026-02-28.
- [2] National Highway Traffic Safety Administration (NHTSA), “Traffic Safety Facts 2022: A Compilation of Motor Vehicle Traffic Crash Data,” U.S. Department of Transportation, Washington, D.C., Tech. Rep. DOT HS 813 656, 2023, available from the National Center for Statistics and Analysis.
- [3] R. C. Coakley and E. Stollof, “Intersection safety needs identification report,” *Prepared for Federal Highway Administration (DTFH61-05-D-00026)*, 2009.
- [4] S. Rahmani, Z. G. Xu, S. C. Calvert, and B. van Arem, “Automated vehicles at unsignalized intersections: Safety and efficiency implications of mixed human and automated traffic,” *Transportation Research Record*, vol. 0, no. 0, p. 03611981251370343, 2025.

- [5] Michigan Legislature, "Michigan vehicle code (excerpt) act 300 of 1949, section 257.649: Right of way; rules; violation as civil infraction," Michigan Compiled Laws (MCL), 1949, accessed online.
- [6] K. Mokhtari and A. R. Wagner, "Safe deep q-network for autonomous vehicles at unsignalized intersection," *arXiv preprint arXiv:2106.04561*, 2021.
- [7] J. Chen, V. Sugumaran, and P. Qu, "Connected and automated vehicle control at unsignalized intersection based on deep reinforcement learning in vehicle-to-infrastructure environment," *International Journal of Distributed Sensor Networks*, vol. 18, no. 7, p. 15501329221114060, 2022.
- [8] P. Haritz, D. Wanke, and T. Liebig, "Enhancing safety for autonomous agents in partly concealed urban traffic environments through representation-based shielding," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, 2024, pp. 1758–1763.
- [9] C. Zhang, K. Kacem, G. Hinz, and A. Knoll, "Safe and rule-aware deep reinforcement learning for autonomous driving at intersections," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 2708–2715.
- [10] S. Khaitan and J. M. Dolan, "State dropout-based curriculum reinforcement learning for self-driving at unsignalized intersections," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 12 219–12 224, 2022.
- [11] M. Al-Sharman, R. Dempster, M. A. Daoud, M. Nasr, D. Rayside, and W. Melek, "Self-learned autonomous driving at unsignalized intersections: A hierarchical reinforced learning approach for feasible decision-making," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 12 345–12 356, 2023.
- [12] B. Peng, M. F. Keskin, B. Kulcsár, and H. Wymeersch, "Connected autonomous vehicles for improving mixed traffic efficiency in unsignalized intersections with deep reinforcement learning," *Communications in Transportation Research*, vol. 1, p. 100017, 2021.
- [13] C. Spatharis and K. Blekas, "Multiagent reinforcement learning for autonomous driving in traffic zones with unsignalized intersections," *Journal of Intelligent Transportation Systems*, vol. 28, no. 1, pp. 103–119, 2024.
- [14] J. Feng, X. Yuan, Z. Li, X. Pan, and K. Liu, "Right-of-way based multi-agent deep reinforcement learning for collaborative decision-making at unsignalized intersection," *Expert Systems with Applications*, vol. 299, p. 130051, 2026.
- [15] A. Abouelazm, J. Michel, and J. M. Zöllner, "A review of reward functions for reinforcement learning in the context of autonomous driving," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, 2024, pp. 156–163.
- [16] S. Maierhofer, A.-K. Rettinger, E. C. Mayer, and M. Althoff, "Formalization of interstate traffic rules in temporal logic," in *2020 IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 752–759.
- [17] P. Nilsson, O. Hussien, A. Balkan, Y. Chen, A. D. Ames, J. W. Grizzle, N. Ozay, H. Peng, and P. Tabuada, "Correct-by-construction adaptive cruise control: Two approaches," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 4, pp. 1294–1307, 2016.
- [18] K. M. Arfvidsson, F. J. Jiang, K. H. Johansson, and J. Mårtensson, "Ensuring safety at intelligent intersections: Temporal logic meets reachability analysis," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, 2024, pp. 292–298.
- [19] R. Karagulle, N. Aréchiga, A. Best, J. DeCastro, and N. Ozay, "A safe preference learning approach for personalization with applications to autonomous vehicles," *IEEE Robotics and Automation Letters*, vol. 9, no. 5, pp. 4226–4233, 2024.
- [20] S. Sadraddini and C. Belta, "Robust temporal logic model predictive control," in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2015, pp. 772–779.
- [21] R. Karagulle, M. A. Valdez Calderon, and N. Ozay, "4c: Custom-and-correct-by-construction controller synthesis using multi-modal human feedback," *IFAC-PapersOnLine*, vol. 58, no. 30, pp. 254–259, 2024, 5th IFAC Workshop on Cyber-Physical Human Systems.
- [22] D. Sadigh, E. S. Kim, S. Coogan, S. S. Sastry, and S. A. Seshia, "A learning based approach to control synthesis of markov decision processes for linear temporal logic specifications," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 1091–1096.
- [23] R. T. Icarte, T. Klassen, R. Valenzano, and S. McIlraith, "Using reward machines for high-level task specification and decomposition in reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2107–2116.
- [24] C. Yang, M. Littman, and M. Carbin, "On the (in) tractability of reinforcement learning for ltl objectives," *arXiv preprint arXiv:2111.12679*, 2021.
- [25] H. Venkataraman, D. Aksaray, and P. Seiler, "Tractable reinforcement learning of signal temporal logic objectives," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 308–317.
- [26] Z. Cui, Y. Wang, N. Bian, and H. Chen, "Reward machine reinforcement learning for autonomous highway driving: A unified framework for safety and performance," in *2023 7th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, 2023, pp. 1–6.
- [27] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," *arXiv preprint arXiv:1708.06374*, 2017.
- [28] E. Leurent, "An environment for autonomous driving decision-making," <https://github.com/eleurent/highway-env>, 2018.
- [29] P. Alvarez Lopez, A. Banse, M. Behrisch, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, R. Nippold, and P. Wagner, "Simulation of urban mobility (sumo)," Nov. 2025. [Online]. Available: <https://doi.org/10.5281/zenodo.17594250>
- [30] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Phys. Rev. E*, vol. 62, pp. 1805–1824, Aug 2000.
- [31] P. Polack, F. Alché, B. d'Andréa Novel, and A. de La Fortelle, "The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?" in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 812–818.
- [32] S. Krauss, "Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics," Ph.D. dissertation, German Aerospace Center (DLR), 1998.
- [33] M. J. Hausknecht and P. Stone, "Deep recurrent q-learning for partially observable mdps," in *AAAI fall symposia*, vol. 45, 2015, p. 141.
- [34] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: reliable reinforcement learning implementations," *J. Mach. Learn. Res.*, vol. 22, no. 1, Jan. 2021.
- [35] M. L. Shehab, A. Aspeel, and N. Ozay, "Learning reward machines from partially observed policies," *Transactions on Machine Learning Research (TMLR)*, 2025.
- [36] —, "Active reward machine inference from raw state trajectories," in *17th World Symposium on the Algorithmic Foundations of Robotics (WAFR)*, 2026.