Integrating LTL Constraints into PPO for Safe Reinforcement Learning

Maifang Zhang[1,*], Hang Yu[2,*], Qian Zuo[1,*], Cheng Wang[3], Vaishak Belle[1], and Fengxiang He[1,†]

***Abstract*— This paper proposes Proximal Policy Optimization with Linear Temporal Logic Constraints (PPO-LTL), a framework that integrates safety constraints written in LTL into PPO for safe reinforcement learning. LTL constraints offer rigorous representations of complex safety requirements, such as regulations that broadly exist in robotics, enabling systematic monitoring of safety requirements. Violations against LTL constraints are monitored by limit-deterministic Büchi automata, and then translated by a logic-to-cost mechanism into penalty signals. The signals are further employed for guiding the policy optimization via the Lagrangian scheme. Extensive experiments on the Zones and CARLA environments show that our PPO-LTL can consistently reduce safety violations, while maintaining competitive performance, against the state-of-the-art methods. The code is at `https://github.com/EVIEHub/PPO-LTL`.**

I. Introduction

Reinforcement Learning (RL) has achieved remarkable success across diverse domains, including robotics [1]. In RL, Proximal Policy Optimization (PPO) is a widely adopted on-policy method, which constrains policy updates with a clipped surrogate objective, striking a balance between exploration and stability [2]. Deploying RL in safety-critical environments remains highly challenging, where violations against safety constraints can lead to catastrophic outcomes. Safe RL has been comprehensively reviewed and addresses this challenge in the framework of constrained optimization, where the agent seeks to maximize reward whilst bounding cumulative safety costs [3]. Within this family, constrained PPO via the Lagrangian scheme (PPO-Lagrangian) has emerged as a major approach [4].

Despite these advances, a critical limitation remains: the constraints need to be written in analytic inequalities of the agent's state and action. This is not compatible with a large family of abstract safety constraints, such as regulations that broadly exist in robotics. For example, the British Highway Code regulates driving behavior [5], which is difficult to translate to the aforementioned inequalities. This calls for machine-computable and principled safety specifications within the RL training process.

To address this issue, we propose Proximal Policy Optimization with Linear Temporal Logic Constraints (PPO-LTL), a novel method that represents abstract constraints in LTL [6], [7], and embeds these constraints into PPO agents. LTL provides a rigorous and machine-verifiable tool to encode temporal properties, such as "always avoid unsafe states," "eventually reach a goal,", and regulatory rules, like "stop at a red light until it turns green," as logic specifications.

We design a logic-to-cost mechanism that systematically translates violations of temporal logic constraints into cost signals that guide policy learning. This mechanism can be instantiated in a wide range of environments, serving as a plug-and-play solution. Each LTL specification is first compiled into an ω-automaton, typically a limit-deterministic Büchi automaton (LDBA) [8], [9]. The automaton encodes the satisfaction conditions of the temporal property by defining a finite set of states and labeled transitions based on atomic propositions. During execution, it evolves synchronously with the agent-environment interaction, effectively acting as a runtime monitor that checks whether the agent's trajectory satisfies the specification.

When a violation occurs, the monitor emits a cost signal to reflect the severity of the violation, determined by a set of pre-defined weights associated with different safety rules. These violation costs are aggregated over time and integrated into the safe reinforcement learning framework. The resulting signals are then combined with policy optimization via the Lagrangian scheme, allowing the agent to optimize task performance while ensuring that safety requirements are satisfied. Unlike handcrafted penalties, this approach provides a principled, generalizable, and modular way to encode high-level safety requirements into the learning process.

We provide a rigorous theoretical analysis of our approach. We formulate PPO-LTL as an inexact projected primal–dual method driven by biased stochastic gradient oracles. Specifically, we view two usual components in PPO, clipped surrogate objective and finite-epoch minibatch updates, as mechanisms that yield biased stochastic approximations to the true Lagrangian gradient. We then prove an ergodic stationarity guarantee for the projected primal-dual dynamics that underpin PPO-LTL. This result shows that despite the biased and noisy gradient estimates, our algorithm consistently converges to a neighborhood of a stationary point. Practically, it means PPO-LTL can stably reduce constraint violations without relying on exact gradient evaluations, highlighting its robustness in challenging settings such as autonomous driving.

We conduct comprehensive experiments to evaluate PPO-LTL across diverse benchmark environments, including ZonesEnv (continuous control with logical regions) [10] and CARLA (autonomous driving simulator) [11]. Our method

*Equal Contribution.

†Corresponding author. Email: F.He@ed.ac.uk.

[1]M. Zhang, Q. Zuo, V. Belle, and F. He were with the School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, Scotland.

[2]H. Yu was with the School of Computer Science, Faculty of Engineering, University of Sydney, Darlington NSW 2008, Australia.

[3]C. Wang was with the School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh EH14 4AS, Scotland.

is compared against PPO [2], as the baseline, and a range of state-of-the-art safe RL methods, including TIRL-PPO, TIRL-SAC [12], PPO-Mask, PPO-Shielding [13], [14], and PPO-Lagrangian. The empirical results demonstrate consistent reductions in safety violations while maintaining competitive task performance. Extensive ablation study and sensitivity analysis further verify our contributions.

II. Related Work

Shielding. Shielding approaches enforce safety by preempting unsafe actions online using verified policies or model-checking over abstract models [13], [14]. This yields binary-safe behavior with strong formal guarantees but can restrict exploration and lead to non-stationary data distributions. In contrast, soft-integration methods map violations into cost signals through a logic-to-cost mechanism, providing dense feedback that is compatible with gradient-based optimization and constrained MDP solvers [4], [15]. Using LTL monitors to emit per-rule costs further enables modular handling of multiple constraints, compositional reasoning across specifications, and straightforward scalability to large rule sets [7], [16]. Compared with runtime action filtering strategies, PPO-LTL incorporates temporal logic constraints directly within the policy optimization loop, providing dense feedback signals that support learning under complex temporal requirements.

Logic tools in RL. Beyond LTL, other formal logics have also been explored in reinforcement learning. Signal Temporal Logic (STL) [17] extends temporal reasoning to real-valued signals, while deontic logics [18] express normative concepts like obligations and permissions. These approaches offer richer expressiveness but often come with higher computational costs and task-specific design complexity, limiting their practicality. Probabilistic Logic Programming (PLP) [19] has also been introduced to model uncertainty in logical reasoning, allowing agents to encode probabilistic constraints. In contrast, LTL achieves a balance: it is lightweight, easy to compile into automata, and sufficient to capture temporal safety properties [20]. Recent work [21] applies LTL in a model-based driving framework via product MDPs and linear programming for policy synthesis, while our method focuses on learning-based constrained optimization rather than model-based synthesis.

III. Preliminaries

Markov Decision Process (MDP). We consider a discounted constrained MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, c, \mu, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, respectively, $P(s'|s,a)$ represents the transition probability of moving from state s to state s' given action a, $r : (s,a) \to [0, R_{\max}]$ is the reward function, $c : (s,a) \to [0, C_{\max}]$ is the aggregated cost $c(s,a) = \sum_{i=1}^{K} w_i c^{(i)}(s,a)$ where w_i are fixed weights for $i \in [K]$, μ is the initial-state distribution, and $\gamma \in (0,1)$ is the discount factor controlling how much the agent values future rewards. A policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ is a mapping from the state space to the space of probability distribution over the action space and $\pi(a \mid s)$ denotes the probability of selecting action a at state s. Let $\theta \in \Theta \subset \mathbb{R}^p$ denote the parameters of a stochastic policy $\pi_\theta(\cdot|s)$. For a given policy π_θ, define the discounted occupancy measure over $\mathcal{S} \times \mathcal{A}$ as $d_\theta^\gamma = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \mathrm{P}_{\pi_\theta}(s_t = s, a_t = a|\mu)$. We define reward value function and cost value function as,

$$J_R(\theta) = \mathbb{E}_{(s,a)\sim d_\theta^\gamma}\left[r(s,a)\right], \; J_C(\theta) = \mathbb{E}_{(s,a)\sim d_\theta^\gamma}\left[c(s,a)\right].$$

The constrained optimization objective is as follows,

$$\max_{\theta\in\Theta} J_R(\theta) \quad \text{s.t.} \quad J_C(\theta) \le d,$$

where $d = \sum_{i=1}^{K} w_i d_i$ is the aggregated safety budget. The corresponding Lagrangian function is defined as $\mathcal{L}(\theta, \lambda) = J_R(\theta) - \lambda(J_C(\theta) - d)$, where $\lambda \in [0, \Lambda]$ is the Lagrangian multiplier with $\Lambda < \infty$.

Proximal Policy Optimization (PPO). Policy gradient methods directly optimize the policy but could suffer from instability if updates are too large [22]. PPO addresses this issue by clipping updates, effectively putting a "safety belt" on learning. The clipped surrogate objective $L^{\text{PPO}}(\pi)$ is defined as follows,

$$\mathbb{E}_t\Big[\min\left(\rho_t(\pi)A_t,\ \text{clip}(\rho_t(\pi), 1-\epsilon, 1+\epsilon)A_t\right)\Big],$$

where $\rho_t(\pi) = \frac{\pi(a_t|s_t)}{\pi_{\text{old}}(a_t|s_t)}$ is the importance ratio, A_t is the advantage estimate (measuring how good an action is compared to the average), and ϵ is a clipping parameter that prevents drastic updates. In practice, PPO optimizes this objective using multiple epochs of mini-batch stochastic gradient updates over collected rollouts.

Linear Temporal Logic (LTL). LTL is a formal language used to specify temporal properties over infinite sequences of system states. A formula in LTL, called a specification, defines a desired temporal behavior of the system. An LTL formula is constructed from a finite set of atomic propositions $\mathcal{P}$, which represent fundamental facts about the environment (e.g., "collision occurred" or "goal reached"). Each proposition is a Boolean statement about the system state that is either true or false, such as whether the vehicle is currently in a safe zone or whether a traffic light is green. These propositions are combined using Boolean connectives ($\neg$, $\wedge$, $\vee$) and temporal operators such as $\mathbf{G}$ (always), $\mathbf{F}$ (eventually), $\mathbf{X}$ (next), and $\mathbf{U}$ (until) to express more complex requirements over time. The semantics of an LTL formula specify the precise conditions under which a specification is considered satisfied. They are defined over a trajectory $\sigma = s_0 s_1 s_2 \ldots$, where $\sigma, t \models \varphi$ indicates that the specification φ holds at time t — in other words, the system's behavior up to time t conforms to the property described by φ. For example, $\mathbf{G}\neg\texttt{collision}$ requires that collisions never occur, while $\mathbf{F}\,\texttt{goal}$ means the goal state must eventually be reached.

IV. PPO-LTL

A. Constraints as LTL Specifications

In real-world applications, many safety requirements depend simultaneously on the environment's state (e.g., position, distance to obstacles) and the temporal structure of

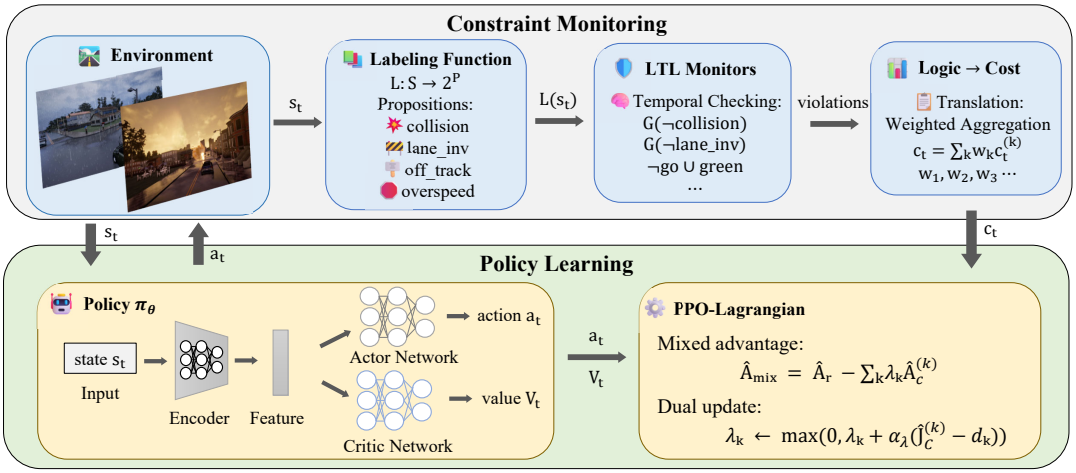

Fig. 1: PPO-LTL: environment states are labeled with atomic propositions, monitored by LTL checkers to generate constraint costs, which are integrated with task rewards for policy optimization.

events (e.g., their order and timing) [3], [4], [16], [23]. For example, requirements like "reach the goal eventually after visiting a checkpoint" cannot be simply encoded as scalar penalties. They require temporal reasoning over sequences of states [24], [20]. To formally represent such requirements, we define them as LTL specifications. Each specification is a logical formula describing desired temporal behavior over system trajectories and serves as a formal requirement that the agent's policy should satisfy. A pilot work has been in the literature that represents traffic laws as LTL specifications. LTL combines temporal operators with Boolean connectives to specify complex behaviors, e.g., always avoiding collisions while eventually reaching a destination ($\mathbf{G}\neg\texttt{collision} \wedge \mathbf{F}\,\texttt{destination}$), or requiring that entering an intersection is eventually followed by a green light ($\mathbf{G}(\texttt{intersection} \rightarrow \mathbf{F}\,\texttt{green})$). This expressiveness enables compact modeling of multi-stage driving rules in autonomous driving scenarios.

B. Logic-to-Cost Mechanism

Büchi Automata and LDBA. Each LTL specification can be translated into a Büchi automaton (BA), a state-transition structure that monitors whether the agent's event sequence satisfies the temporal rule. The automaton reads the interaction trajectory, and satisfaction is achieved when designated accepting states are visited infinitely often during execution. For reinforcement learning, we adopt a simplified variant called the limit-deterministic Büchi automaton (LDBA), which provides more predictable checking and improved computational efficiency. The LDBA enables high-level temporal logic to be evaluated step by step during training, forming the basis for converting symbolic rules into numerical signals that guide policy learning.

Monitors. During training, temporal specifications are checked by runtime monitors that evolve synchronously with the environment. Each monitor observes the trajectory and determines whether a specification ϕ_i is satisfied. When a violation-related transition is detected, the monitor emits a nonnegative cost signal $c_t^{(i)}$, whose magnitude is determined by a rule-specific weight reflecting its relative importance. Multiple monitors may generate costs simultaneously, and all violation costs are aggregated to guide policy optimization in the CMDP framework: $c_t = \sum_{i=1}^{K} c_t^{(i)}$. Safety-critical rules contribute larger costs, while goal-oriented requirements maintain elevated costs until the condition is satisfied.

Reach-Avoid Decomposition. To further simplify policy optimization, each compiled automaton can be decomposed into a sequence of reach-avoid subtasks [24], [25]. Each subtask consists of (1) a reach condition: the state or event that must eventually occur, and (2) an avoid condition: the event that must never happen. For example, $\mathbf{F}(\texttt{goal}) \wedge \mathbf{G}\neg\texttt{collision}$ is decomposed into two subtasks: "always avoid collisions" and "eventually reach the goal." This transformation reduces policy search complexity while preserving the original temporal semantics.

Logic-to-Cost Mechanism. The final environment feedback is therefore given by: $(s_{t+1}, r_t, c_t, \texttt{info})$, where r_t encodes task performance, c_t represents aggregated constraint costs, and $\texttt{info}$ provides diagnostic information for each rule. This runtime logic-to-cost conversion is domain-agnostic and can be applied across diverse environments, guiding reinforcement learning toward policies that satisfy temporal logic constraints while optimizing performance.

C. The Langragian Scheme in PPO-LTL

Given the per-step violation costs produced by the logic-to-cost mechanism, PPO-LTL incorporates them directly into the policy optimization process. We solve the constrained optimization problem using a primal-dual approach, where constraint information influences policy updates through a mixed advantage signal $\hat{A}_{\text{mix}} = \hat{A}_r - \sum_{k=1}^{K} \lambda_k \hat{A}_c^{(k)}$, where $\hat{A}_r$ and $\hat{A}_c^{(k)}$ are generalized advantage estimates for reward and cost, respectively. After each PPO update, the multipliers are updated via projected gradient ascent:

$$\lambda_k \leftarrow \max\left(0,\ \lambda_k + \alpha_\lambda\big(\hat{J}_C^{(k)} - d_k\big)\right).$$

When costs exceed their pre-defined limits, λ_k increases, strengthening the penalty applied to violations. Conversely, when the costs remain within acceptable bounds, λ_k decreases or stays constant, enabling the policy to continue improving task performance.

V. Theoretical Guarantee

In this section, we analyze the convergence properties of PPO-LTL. We formulate the learning process within a Product MDP framework: the state space is treated as an augmented space $\mathcal{S} = \mathcal{S}_{env} \times \mathcal{Q}$, where $\mathcal{S}_{env}$ is the environment state, and $\mathcal{Q}$ is the LDBA state. The temporally dependent LTL cost becomes strictly Markovian, allowing defining the cost function as $c(s,a) = \sum_{k=1}^{K} w_k c^{(k)}(s,a)$. Correspondingly, PPO-LTL generates an iterate sequence $\{(\theta_t, \lambda_t)\}_{t\geq 0}$ as follows:

$$\theta_{t+1} = \Pi_\Theta(\theta_t + \alpha \hat{g}_t), \quad \lambda_{t+1} = \Pi_{[0,\Lambda]}(\lambda_t + \beta \hat{u}_t), \quad (1)$$

where Π denotes Euclidean projection, $\alpha,\ \beta > 0$ are the learning rates, $\hat{g}_t$ is a stochastic ascent direction for the primal objective, and $\hat{u}_t$ is a stochastic estimate of the constraint residual $\hat{J}_C(\theta_t) - d$. We further abstract PPO-LTL in an inexact projected primal-dual framework with biased stochastic gradient oracles: the clipped surrogate optimization and finite-epoch minibatch updates in PPO are modeled as producing biased stochastic estimates of the true Lagrangian gradient. For given learning rates $\alpha, \beta > 0$, define the primal and dual gradient mappings as follows:

$$\mathcal{G}(\theta,\lambda) := \frac{1}{\alpha}\left(\Pi_\Theta\left(\theta + \alpha \nabla_\theta \mathcal{L}(\theta,\lambda)\right) - \theta\right),$$
$$\mathcal{H}(\theta,\lambda) := \frac{1}{\beta}\left(\Pi_{[0,\Lambda]}\left(\lambda + \beta(J_C(\theta) - d)\right) - \lambda\right),$$

quantifying first-order stationarity of this problem under projection.

The theory relies on two mild assumptions on the primal domain and stochastic gradient, given in Sec VII, following usual practice [26], [2], [27]. We prove the following theorem.

Theorem 1: Conditioned on Assumptions 1 and 2 and the learning rates $0 < \alpha \leq 1/(4L_{\mathcal{L}})$ and $\beta > 0$, let $\{\theta_t, \lambda_t\}_{t\geq 0}$ be defined by (1). Then, for all $T \geq 1$,

$$\begin{aligned}
&\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\mathcal{G}(\theta_t,\lambda_t)\|] \\
\leq &\sqrt{\frac{2(\Delta_{\mathcal{L}} + 2\Lambda U_{\max})}{\alpha T}} + O\left(\sqrt{\sigma_\theta^2 + a^2 + \alpha}\right) \\
&+ O\left(\left(G_{\max}^2 + \sigma_\theta^2 + a^2\right)^{1/4}\right), \\
&\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\mathcal{H}(\theta_t,\lambda_t)\|] \\
\leq &\sqrt{\frac{\Delta_{\mathcal{L}}}{\beta T}} + O\left(\sqrt{U_{\max}^2 + \sigma_\lambda^2 + b^2 + \alpha^2 + \frac{G_{\max}^2}{\beta}}\right) \\
&+ O\left(\frac{\alpha}{\sqrt{\beta}}\sqrt{G_{\max}^2 + \sigma_\theta^2 + a^2}\right),
\end{aligned}$$

TABLE I: Results in ZonesEnv.

Method	Reward	Hit Wall Rate	λ
PPO	17.95 ± 0.84	3.7 ± 2.1%	-
PPO-Mask	10.14 ± 2.82	6.0 ± 2.0%	-
PPO-Shielding	15.92 ± 0.55	12.0 ± 2.0%	-
PPO-Lagrangian	23.23 ± 1.51	6.0 ± 2.0%	0.00
PPO-LTL-A	17.86 ± 0.73	4.3 ± 3.5%	0.0048
PPO-LTL-B	18.61 ± 0.67	4.7 ± 3.1%	0.0018

where $\Delta_{\mathcal{L}} = \sup_{\Theta\times[0,\Lambda]} \mathcal{L}(\theta,\lambda) - \inf_{\Theta\times[0,\Lambda]} \mathcal{L}(\theta,\lambda) < \infty$, $U_{\max} = \sup_{\theta\in\Theta} |J_C(\theta) - d|$, $G_{\max} = \sup_{(\theta,\lambda)} \|\nabla_\theta \mathcal{L}(\theta,\lambda)\|$, and $O(\cdot)$ hides problem-dependent constants independent of T.

A detailed proof is given in Section VII. Theorem 1 establishes an ergodic stationarity guarantee for the projected primal-dual dynamics underlying PPO-LTL. It demonstrates that despite the biased and noisy gradient estimates inherent to PPO (e.g., due to clipping and minibatch sampling, captured by the variance and bias terms σ and a, b), the algorithm reliably converges to a neighborhood of the stationary point. In practice, this implies that PPO-LTL can stably minimize constraint violations without requiring exact gradient computation, confirming its robustness in complex environments like autonomous driving.

VI. Experiments

We conduct extensive experiments on ZonesEnv and CARLA, which fully support our algorithm.

Baselines and Comparison Methods. PPO is used as an unconstrained baseline. TIRL-PPO and TIRL-SAC are included as standard Safe RL baselines to evaluate the performance of alternative constrained optimization techniques. PPO-Mask is a heuristic safety filter that preemptively overrides imminent unsafe actions with predefined safe fallbacks (e.g., hard braking). We include this to illustrate the limitations (e.g., deadlocks and over-conservatism) of purely reactive, rule-based interventions. PPO-Shielding is our main competitor in the experiments. PPO-Lagrangian is included as a standard constrained RL method. For fair comparison, all methods use a CNN backbone, consisting of 6 convolutional layers with ReLU activations, followed by policy and value networks with 2-layer MLPs containing [500, 300] units each.

Environments and Implementation. We evaluate on two environments using a 256-dimensional CustomMultiInputExtractor [28] and 3 random seeds. (1) **ZonesEnv:** A Safety Gymnasium [10] grid-world where a point robot navigates colored zones representing atomic propositions (e.g., "avoid blue until green"). Wall collisions yield penalties, and LTL violations generate costs. Models are trained for 200k steps. (2) **CARLA:** An autonomous driving simulator [11] (v0.9.13, Town02). Observations include semantic segmentation, ego-states, and the one-hot LDBA state. LTL monitors map events to generate deterministic costs for PPO-Lagrangian updates. Models are trained for 100k steps.

TABLE II: Results in CARLA. Arrows (↑, ↓) indicate the direction where values are strictly better.

Methods	Collision Rate ↓	Routes ↑	Distance	Speed	Length	Col. Num	Cent. Dev	Cost	λ
PPO	0.262 ± 0.115	0.013 ± 0.008	6.58	0.45	4030	8.0	0.77	-	-
TIRL-PPO	0.173 ± 0.129	0.083 ± 0.111	6.37	0.07	9460	7.3	0.87	-	-
TIRL-SAC	0.336 ± 0.063	0.027 ± 0.011	6.92	0.92	926	57.3	0.12	-	-
PPO-Mask	0.408 ± 0.058	0.010 ± 0.012	2.38	1.06	1338	38.0	0.37	-	-
PPO-Shielding	0.267 ± 0.056	0.072 ± 0.036	18.87	10.63	93	164.3	0.52	-	-
PPO-Lagrangian	0.233 ± 0.089	0.077 ± 0.051	7.78	0.96	6616	103.0	0.60	0.00	0.01
PPO-LTL-A	**0.143 ± 0.110**	0.077 ± 0.107	5.96	3.57	703	154.3	0.52	0.25	0.03
PPO-LTL-B	0.170 ± 0.148	**0.236 ± 0.037**	12.79	1.48	4859	10.0	0.73	0.08	0.00

TABLE III: Ablation studies and sensitivity analyses in CARLA.

Configuration	Collision Rate ↓	Routes ↑	Distance	Speed	Length	Col. Num	Cent. Dev	Cost	λ
Ablation of LTL Components									
No collision	**0.159 ± 0.027**	0.009 ± 0.003	2.71	0.68	4123	11.0	0.64	0.07	0.00
No off-track	0.275 ± 0.129	0.032 ± 0.044	3.29	0.06	10501	5.7	0.62	0.07	0.00
No lane invasion	0.207 ± 0.046	**0.024 ± 0.009**	6.17	1.98	14323	21.7	0.57	0.06	0.01
Sensitivity of Cost Limit (cl) and Dual Learning Rate (α_λ)									
cl = 0.001	0.229 ± 0.153	0.033 ± 0.035	5.19	1.10	4749	21.0	0.78	0.06	0.01
cl = 0.05	0.273 ± 0.199	0.020 ± 0.023	3.04	4.65	907	228.7	0.56	0.22	0.71
cl = 0.5	0.395 ± 0.130	0.045 ± 0.059	4.86	0.09	6614	11.7	0.91	0.01	0.00
α_λ = 0.00001	**0.160 ± 0.059**	0.062 ± 0.063	8.67	0.89	2957	11.7	0.95	0.01	0.00
α_λ = 0.0001	0.243 ± 0.081	0.053 ± 0.033	8.98	0.54	3029	9.7	0.63	0.07	0.00
α_λ = 0.01	0.233 ± 0.097	**0.099 ± 0.065**	8.28	2.05	3776	8.3	0.86	0.06	0.03
Constraint Strictness & Mixed Formulations									
Relaxed high	0.258 ± 0.067	0.042 ± 0.040	4.51	0.09	3380	13.3	0.62	0.01	0.00
Relaxed moderate	0.205 ± 0.100	0.056 ± 0.053	3.79	1.50	2787	53.7	0.53	0.01	0.00
Collision focused	0.233 ± 0.206	0.010 ± 0.008	0.07	0.22	3020	15.7	0.39	0.00	0.00
Ultra loose	0.393 ± 0.110	0.141 ± 0.098	8.81	1.04	3996	18.0	0.77	0.00	0.00
Mixed light	0.282 ± 0.076	0.042 ± 0.056	3.37	0.64	4218	18.0	0.90	0.01	0.00
Mixed light loose	0.242 ± 0.018	0.041 ± 0.033	2.86	0.26	5728	7.0	0.48	0.01	0.00
Mixed medium	0.229 ± 0.131	0.044 ± 0.029	5.62	3.69	3248	50.7	0.81	0.06	0.00
Collision-Only Variants Without Temporal LTL									
Col-only (cl=0.5)	0.243 ± 0.079	0.033 ± 0.043	4.93	0.52	4155	11.3	1.04	0.00	0.00
Col-only (cl=0.1)	0.248 ± 0.228	0.028 ± 0.030	6.10	0.40	3573	14.3	0.90	0.00	0.00
Col-only mid	0.346 ± 0.175	0.158 ± 0.200	9.07	2.07	1584	52.0	0.86	0.01	0.00
Col-only loose	0.193 ± 0.081	**0.269 ± 0.297**	12.66	0.93	9531	9.3	0.56	0.00	0.00

Evaluation Metrics. We assess both task performance and constraint satisfaction. Safety metrics include violation rates for each constraint type (collision, off-track, lane invasion, heading, weaving, overspeed, steering jerk), computed as the ratio of violation events to total steps (e.g., VR $= N_{\text{vio}}/N_{\text{total}}$) [4]. Task metrics include route completion rate (RCR), average speed $\bar{v}$, and total distance traveled D_{total}, where RCR $= d_{\text{completed}}/d_{\text{target}}$. Constraint dynamics are measured by the average episodic cost $\hat{J}_C = \frac{1}{N}\sum_i C_i$, the final Lagrange multiplier λ^*, and the convergence behavior over training.

Comparison Results in ZonesEnv. Table I presents performance across 3 seeds. PPO-LTL-A prioritizes stability, while PPO-LTL-B slightly relaxes constraints. Baselines exhibit distinct flaws: heuristic PPO-Mask severely restricts exploration 10.14 reward, while PPO-Shielding struggles with continuous dynamics, yielding the highest hit-wall rate 12.0%. Although PPO-Lagrangian achieves the highest apparent reward 23.23, this is deceptive; lacking LTL memory, it ignores temporal rules and incurs a massive unshown violation cost of 56.98. Standard PPO maintains a low hit-wall rate 3.7% but cannot enforce complex temporal specifications. In contrast, both PPO-LTL variants provide well-balanced policies. They significantly outperform Mask and Shielding in valid rewards while strictly adhering to LTL constraints with competitive hit-wall rates 4.3% and 4.7%.

Comparison Results in CARLA. Table II compares PPO-LTL-A (strict cost limit 0.02) and PPO-LTL-B (relaxed limit 0.1) against baselines. PPO fails to balance safety and liveness. Standard Safe RL baselines exhibit severe pathologies: TIRL-PPO suffers from the freezing robot problem (near-zero speed despite 9460-step survival), while TIRL-SAC fails to converge safely (0.336 collision rate). Furthermore, PPO-Shielding shows a reckless driving pattern: despite deceptive high speeds, it crashes rapidly (93-step length, 164.3 collisions) with minimal route completion (0.072). Conversely, PPO-Mask's sudden stops cause conservative deadlocks (2.38 distance) and ironically higher collisions (0.408). PPO-Lagrangian's lack of temporal foresight limits progress (7.78 distance). In contrast, PPO-LTL balances

TABLE IV: Sensitivity analysis of cost limit and Lagrangian learning rate (α_λ) in ZonesEnv for PPO-LTL.

Cost Limit	α_λ	Reward ↑	Hit Wall Rate ↓	λ
0.03	0.008	18.04 ± 1.30	6.7 ± 1.2%	0.0072
0.05	0.010	17.86 ± 0.73	**4.3 ± 3.5%**	0.0048
0.07	0.015	**18.61 ± 0.67**	4.7 ± 3.1%	0.0018
0.10	0.020	18.35 ± 1.42	5.7 ± 2.1%	0.0002

proactive safety and task liveness, avoiding both over-conservatism and reckless speed. PPO-LTL-A achieves the lowest collision rate (0.143, a 45% reduction over standard PPO). PPO-LTL-B achieves the highest route completion (0.236) and maintains long, stable episodes.

Ablation Study and Sensitivity Analysis. Table III evaluates diverse constraint configurations, constraint contributions through systematic removal, and hyperparameter robustness on the CARLA environment. The results verify the necessity of carefully balancing LTL constraints: simplistic or overly relaxed bounds can easily induce naive speed or reckless driving, while appropriately tuned temporal logic ensures safe and effective task execution. Furthermore, removing individual constraints confirms that multi-component LTL constraints are essential for balanced driving performance. Table IV presents a sensitivity analysis of PPO-LTL across varying cost limits and Lagrangian learning rates on ZonesEnv. Across a threefold range of cost limits, the framework maintains stable behavior, demonstrating that the constraint mechanism provides interpretable and consistent control over policy behavior.

Computational Costs. PPO-LTL incurs negligible overhead compared to standard PPO. Over 3 random seeds, training PPO-LTL (vs. PPO) took 235.3s ± 0.9s (vs. 226.3s ± 0.5s) for 200k steps in ZonesEnv, and 2557.3s ± 97.8s (vs. 2536.3s ± 39.0s) for 100k steps in CARLA. This confirms that LTL monitoring and Lagrangian dual updates introduce minimal computational burden, maintaining practical efficiency for real-world applications.

VII. Proof

Assumption 1: The primal domain Θ is compact and convex, which models the bounded iterates induced by trust-region regularization, clipping, or explicit projection in (1). The value functions $J_R(\theta)$ and $J_C(\theta)$ are continuously differentiable on Θ, and their gradients are Lipschitz:

$$\|\nabla_\theta J_R(\theta) - \nabla_\theta J_R(\theta')\| \le L_R\|\theta - \theta'\|$$
$$\|\nabla_\theta J_C(\theta) - \nabla_\theta J_C(\theta')\| \le L_C\|\theta - \theta'\|.$$

Thus, for any $\lambda \in [0, \Lambda]$, the Lagrangian $\mathcal{L}(\theta, \lambda)$ is $L_\mathcal{L}$-smooth in θ with $L_\mathcal{L} = L_R + \Lambda L_C$.

Assumption 2: Let $\mathcal{F}_t$ denote the σ-field generated by all randomness up to the beginning of t. The estimate $\hat{g}_t$ satisfies

$$\mathbb{E}[\hat{g}_t|\mathcal{F}_t] = \nabla_\theta\mathcal{L}(\theta_t, \lambda_t) + a_t,$$

where $\|a_t\| \le a$, and

$$\mathbb{E}[\|\hat{g}_t - \mathbb{E}[\hat{g}_t|\mathcal{F}_t]\|^2|\mathcal{F}_t] \le \sigma_\theta^2.$$

The dual signal $\hat{u}_t$ satisfies $\mathbb{E}[\hat{u}_t|\mathcal{F}_t] = (J_C(\theta_t) - d) + b_t$ where $\|b_t\| \le b$, and has bounded variance $\mathbb{E}[(\hat{u}_t - \mathbb{E}[\hat{u}_t|\mathcal{F}_t])^2|\mathcal{F}_t] \le \sigma_\lambda^2$. The bias terms a_t and b_t capture approximation effects including clipping mismatch, minibatch updates and estimation errors.

Proof. For a fixed t, we define the true gradient $\tilde{\theta}_{t+1} := \Pi_\Theta(\theta_t + \alpha\nabla_\theta\mathcal{L}(\theta_t, \lambda_t))$. Thus, it holds $\tilde{\theta}_{t+1} - \theta_t = \alpha\mathcal{G}(\theta_t, \lambda_t)$. By Assumption 1, we have

$$\mathcal{L}(\tilde{\theta}_{t+1}, \lambda_t) \ge \mathcal{L}(\theta_t, \lambda_t) + \langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \tilde{\theta}_{t+1} - \theta_t\rangle - \frac{L_\mathcal{L}}{2}\left\|\tilde{\theta}_{t+1} - \theta_t\right\|^2.$$

Since for all θ, $\langle\tilde{\theta}_{t+1} - (\theta_t + \alpha\nabla_\theta\mathcal{L}(\theta_t, \lambda_t)), \theta_t - \tilde{\theta}_{t+1}\rangle \ge 0$, choosing $\theta = \theta_t$ gives

$$\langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \tilde{\theta}_{t+1} - \theta_t\rangle \ge \frac{1}{\alpha}\|\tilde{\theta}_{t+1} - \theta_t\|^2.$$

Then it yields

$$\mathcal{L}(\tilde{\theta}_{t+1}, \lambda_t) - \mathcal{L}(\theta_t, \lambda_t) \ge \left(\frac{1}{\alpha} - \frac{L_\mathcal{L}}{2}\right)\|\tilde{\theta}_{t+1} - \theta_t\|^2,$$
$$= \alpha\left(1 - \frac{\alpha L_\mathcal{L}}{2}\right)\|\mathcal{G}(\theta_t, \lambda_t)\|^2.$$

Under $\alpha < \frac{1}{4L_\mathcal{L}}$, we have $1 - \frac{\alpha L_\mathcal{L}}{2} \ge \frac{7}{8}$, and hence

$$\mathcal{L}(\tilde{\theta}_{t+1}, \lambda_t) - \mathcal{L}(\theta_t, \lambda_t) \ge \frac{7}{8}\alpha\left\|\mathcal{G}(\theta_t, \lambda_t)\right\|^2. \quad (2)$$

Now let $\delta_t := \hat{g}_t - \nabla_\theta\mathcal{L}(\theta_t, \lambda_t)$ denote the gap between $\hat{g}_t$ and the gradient of $\mathcal{L}(\theta_t, \lambda_t)$. Thus, it holds $\|\theta_{t+1} - \tilde{\theta}_{t+1}\| \le \alpha\|\delta_t\|$. Since $\theta_{t+1} = \Pi_\Theta(\theta_t + \alpha(\nabla_\theta\mathcal{L}(\theta_t, \lambda_t) + \delta_t))$, for all $\theta \in \Theta$, we have $\langle\theta_{t+1} - (\theta_t + \alpha(\nabla_\theta\mathcal{L}(\theta_t, \lambda_t) + \delta_t)), \theta - \theta_{t+1}\rangle \ge 0$. Choose $\theta = \tilde{\theta}_{t+1}$, then,

$$\langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t) + \delta_t, \theta_{t+1} - \tilde{\theta}_{t+1}\rangle \ge \frac{1}{\alpha}\langle\theta_{t+1} - \theta_t, \theta_{t+1} - \tilde{\theta}_{t+1}\rangle. \quad (3)$$

Similarly, from the definition of $\tilde{\theta}_{t+1}$,

$$\langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \tilde{\theta}_{t+1} - \theta_{t+1}\rangle \ge \frac{1}{\alpha}\langle\tilde{\theta}_{t+1} - \theta_t, \tilde{\theta}_{t+1} - \theta_{t+1}\rangle. \quad (4)$$

We decompose

$$\langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \theta_{t+1} - \theta_t\rangle = \langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \tilde{\theta}_{t+1} - \theta_t\rangle + \langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \theta_{t+1} - \tilde{\theta}_{t+1}\rangle.$$

For the second term, rearranging (3) yields $\langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \theta_{t+1} - \tilde{\theta}_{t+1}\rangle \ge \frac{1}{\alpha}\langle\theta_{t+1} - \theta_t, \theta_{t+1} - \tilde{\theta}_{t+1}\rangle - \langle\delta_t, \theta_{t+1} - \tilde{\theta}_{t+1}\rangle$. Combining these terms and using the geometric identity $\langle x, x - y\rangle = \frac{1}{2}(\|x\|^2 + \|x - y\|^2 - \|y\|^2)$ yields

$$\langle\nabla_\theta\mathcal{L}(\theta_t, \lambda_t), \theta_{t+1} - \theta_t\rangle \ge \frac{1}{2\alpha}\|\tilde{\theta}_{t+1} - \theta_t\|^2 - \langle\delta_t, \theta_{t+1} - \tilde{\theta}_{t+1}\rangle \ge \frac{\alpha}{2}\|\mathcal{G}(\theta_t, \lambda_t)\|^2 - \alpha\|\delta_t\|^2. \quad (5)$$

By $L_\mathcal{L}$-smoothness again, for any $x, y \in \Theta$,

$$\mathcal{L}(x, \lambda_t) \ge \mathcal{L}(y, \lambda_t) + \langle\nabla_\theta\mathcal{L}(y, \lambda_t), x - y\rangle - \frac{L_\mathcal{L}}{2}\|x - y\|^2.$$

Choosing $x = \theta_{t+1}$ and $y = \theta_t$, we have

$$\mathcal{L}(\theta_{t+1}, \lambda_t) \geq \mathcal{L}(\theta_t, \lambda_t) + \langle \nabla_\theta \mathcal{L}(\theta_t, \lambda_t), \theta_{t+1} - \theta_t \rangle - \frac{L_\mathcal{L}}{2} \|\theta_{t+1} - \theta_t\|^2.$$

Using (5) and $\|\theta_{t+1} - \theta_t\| \leq \alpha(\|\nabla_\theta \mathcal{L}(\theta_t, \lambda_t)\| + \|\delta_t\|)$, we obtain

$$\mathcal{L}(\theta_{t+1}, \lambda_t) - \mathcal{L}(\theta_t, \lambda_t) \geq \frac{\alpha}{2} \|\mathcal{G}(\theta_t, \lambda_t)\|^2 - \alpha \|\delta_t\|^2 - L_\mathcal{L} \alpha^2 \left(\|\nabla_\theta \mathcal{L}(\theta_t, \lambda_t)\|^2 + \|\delta_t\|^2 \right).$$

Since $\Theta \times [0, \Lambda]$ is compact and $\nabla_\theta \mathcal{L}$ is continuous, there exists $G_{\max} < \infty$ such that $\|\nabla_\theta \mathcal{L}\| \leq G_{\max}$. Thus, it yields

$$\mathcal{L}(\theta_{t+1}, \lambda_t) - \mathcal{L}(\theta_t, \lambda_t) \geq \frac{\alpha}{2} \|\mathcal{G}(\theta_t, \lambda_t)\|^2 - C_1 \alpha \|\delta_t\|^2 - C_2 \alpha^2,$$

where C_1, $C_2 > 0$ are constants independent of T. Rearranging and summing it from $t = 0$ to $T - 1$ and taking expectations gives

$$\frac{\alpha}{2} \sum_{t=0}^{T-1} \mathbb{E}\left[\|\mathcal{G}(\theta_t, \lambda_t)\|^2 \right] \leq \mathbb{E}\left[\sum_{t=0}^{T-1} (\mathcal{L}(\theta_{t+1}, \lambda_t) - \mathcal{L}(\theta_t, \lambda_t)) \right] + C_1 \alpha \sum_{t=0}^{T-1} \mathbb{E}\, \|\delta_t\|^2 + C_2 T \alpha^2.$$

Moreover,

$$\sum_{t=0}^{T-1} (\mathcal{L}(\theta_{t+1}, \lambda_t) - \mathcal{L}(\theta_t, \lambda_t)) = \mathcal{L}(\theta_T, \lambda_{T-1}) - \mathcal{L}(\theta_0, \lambda_0) + \sum_{t=1}^{T-1} (\mathcal{L}(\theta_t, \lambda_{t-1}) - \mathcal{L}(\theta_t, \lambda_t)).$$

We define the true dual ascent direction $u_t := J_C(\theta_t) - d$, get the corresponding projection $\tilde{\lambda}_{t+1} := \Pi_{[0,\Lambda]}(\lambda_t + \beta u_t)$, and recall $\mathcal{L}(\theta, \lambda) = J_R(\theta) - \lambda(J_C(\theta) - d)$, the second term becomes $\mathcal{L}(\theta_t, \lambda_{t-1}) - \mathcal{L}(\theta_t, \lambda_t) = (\lambda_t - \lambda_{t-1}) u_t$. Define $S_T := \sum_{t=1}^{T-1} (\lambda_t - \lambda_{t-1}) u_t$ and it yields

$$S_T = \lambda_{T-1} u_{T-1} - \lambda_0 u_1 - \sum_{t=1}^{T-2} \lambda_t \left(u_{t+1} - u_t \right).$$

Since $J_C(\theta)$ is bounded on Θ, there exists $U_{\max} < \infty$ such that $|u_t| \leq U_{\max}$ for all t. Therefore, $|\lambda_{T-1} u_{T-1} - \lambda_0 u_1| \leq 2\Lambda U_{\max}$. Since Θ is compact and $\nabla_\theta J_C(\theta)$ is continuous, there exists $G_C < \infty$ such that $|u_{t+1} - u_t| = |J_C(\theta_{t+1}) - J_C(\theta_t)| \leq G_C \|\theta_{t+1} - \theta_t\|$. Hence, Cauchy-Schwarz and Jensen's inequalities give

$$\left| \sum_{t=1}^{T-2} \lambda_t (u_{t+1} - u_t) \right| \leq \Lambda G_C \sum_{t=1}^{T-2} \|\theta_{t+1} - \theta_t\|, \leq \sqrt{T} \Lambda G_C \left(\sum_{t=1}^{T-2} \|\theta_{t+1} - \theta_t\|^2 \right)^{1/2}.$$

Since $\|\theta_{t+1} - \theta_t\| \leq \alpha \left(\|\nabla_\theta \mathcal{L}(\theta_t, \lambda_t)\| + \|\delta_t\| \right)$, it holds $\mathbb{E}\left[\|\theta_{t+1} - \theta_t\|^2 \right] \leq 2\alpha^2 \left(G_{\max}^2 + \sigma_\theta^2 + a^2 \right)$. Combining the above bounds yields

$$E[|S_T|] \leq 2\Lambda U_{\max} + \alpha T \Lambda G_C \sqrt{2(G_{\max}^2 + \sigma_\theta^2 + a^2)}.$$

Combining $\mathbb{E}[\mathcal{L}(\theta_T, \lambda_{T-1}) - \mathcal{L}(\theta_0, \lambda_0)] \leq \Delta_\mathcal{L}$, we obtain

$$\frac{\alpha}{2} \sum_{t=0}^{T-1} \mathbb{E}[\|\mathcal{G}(\theta_t, \lambda_t)\|^2] \leq \Delta_\mathcal{L} + C_1 \alpha T (\sigma_\theta^2 + a^2) + C_2 T \alpha^2 + 2\Lambda U_{\max} + \alpha T \Lambda G_C \sqrt{2(G_{\max}^2 + \sigma_\theta^2 + a^2)}.$$

Therefore,

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[\|\mathcal{G}(\theta_t, \lambda_t)\|^2] \leq \frac{2(\Delta_\mathcal{L} + 2\Lambda U_{\max})}{\alpha T} + 2C_1(\sigma_\theta^2 + a^2) + 2C_2 \alpha + 2\Lambda G_C \sqrt{2(G_{\max}^2 + \sigma_\theta^2 + a^2)}.$$

By Jensen's inequality and the elementary bound, we get the result. Since $\tilde{\lambda}_{t+1} = \Pi_{[0,\Lambda]}(\lambda_t + \beta u_t)$ and $\lambda_{t+1} = \Pi_{[0,\Lambda]}(\lambda_t + \beta \hat{u}_t)$, we have $\tilde{\lambda}_{t+1} - \lambda_t = \beta \mathcal{H}(\theta_t, \lambda_t)$. We let $\varepsilon_t := \hat{u}_t - u_t$ be the dual estimation error. For any fixed θ_t, we have

$$\mathcal{L}(\theta_t, \lambda_{t+1}) - \mathcal{L}(\theta_t, \lambda_t) = -(\tilde{\lambda}_{t+1} - \lambda_t) u_t - (\lambda_{t+1} - \tilde{\lambda}_{t+1}) u_t. \tag{6}$$

Since for all $\lambda \in [0, \Lambda]$, $\langle \tilde{\lambda}_{t+1} - (\lambda_t + \beta u_t), \lambda - \tilde{\lambda}_{t+1} \rangle \geq 0$, taking $\lambda = \lambda_t$ gives

$$(\tilde{\lambda}_{t+1} - \lambda_t) u_t \geq \frac{1}{\beta} (\tilde{\lambda}_{t+1} - \lambda_t)^2, \geq \beta \|\mathcal{H}(\theta_t, \lambda_t)\|^2. \tag{7}$$

Since

$$|\lambda_{t+1} - \tilde{\lambda}_{t+1}| = |\Pi_{[0,\Lambda]}(\lambda_t + \beta(u_t + \varepsilon_t)) - \Pi_{[0,\Lambda]}(\lambda_t + \beta u_t)| \leq \beta |\varepsilon_t|,$$

it yields

$$-(\lambda_{t+1} - \tilde{\lambda}_{t+1}) u_t \leq \beta |\varepsilon_t| |u_t|. \tag{8}$$

Combining (6), (7) and (8), we obtain

$$\mathcal{L}(\theta_t, \lambda_{t+1}) - \mathcal{L}(\theta_t, \lambda_t) \leq -\beta \|\mathcal{H}(\theta_t, \lambda_t)\|^2 + \beta |\varepsilon_t| |u_t|, \leq -\beta \|\mathcal{H}(\theta_t, \lambda_t)\|^2 + \frac{\beta}{2} \varepsilon_t^2 + \frac{\beta}{2} u_t^2.$$

Moreover, $|\lambda_{t+1} - \lambda_t| \leq \beta(|u_t| + |\varepsilon_t|)$ and

$$-(\lambda_{t+1} - \lambda_t) u_{t+1} = -(\lambda_{t+1} - \lambda_t) u_t - (\lambda_{t+1} - \lambda_t)(u_{t+1} - u_t)$$

for all t. Therefore, apply Young's inequality and it yields

$$|(\lambda_{t+1} - \lambda_t)(u_{t+1} - u_t)| \leq \beta(|u_t| + |\varepsilon_t|) G_C \|\theta_{t+1} - \theta_t\|, \leq \frac{\beta}{4}(u_t^2 + \varepsilon_t^2) + 2\beta G_C^2 \|\theta_{t+1} - \theta_t\|^2.$$

By Assumption 2, $\mathbb{E}[\varepsilon_t^2 | \mathcal{F}_t] \leq \sigma_\lambda^2 + b^2$. Combining all terms together and using

$$\mathcal{L}(\theta_{t+1}, \lambda_{t+1}) - \mathcal{L}(\theta_{t+1}, \lambda_t) = -(\lambda_{t+1} - \lambda_t) u_{t+1},$$

we obtain

$$\mathbb{E}[\mathcal{L}(\theta_{t+1}, \lambda_{t+1}) - \mathcal{L}(\theta_{t+1}, \lambda_t)] \leq -\beta \mathbb{E}[\|\mathcal{H}(\theta_t, \lambda_t)\|^2] + \beta C_3 (U_{\max}^2 + \sigma_\lambda^2 + b^2) + \beta C_4 \alpha^2,$$

where C_3, C_4 are constants independent of T. Summing the decomposition over $t = 0$ to $T - 1$ yields

$$\sum_{t=0}^{T-1} \mathbb{E}[\mathcal{L}(\theta_{t+1}, \lambda_{t+1}) - \mathcal{L}(\theta_t, \lambda_t)] = \mathbb{E}[\mathcal{L}(\theta_T, \lambda_T) - \mathcal{L}(\theta_0, \lambda_0)] \geq -\Delta_{\mathcal{L}}.$$

Since

$$(\mathcal{L}(\theta_{t+1}, \lambda_{t+1}) - \mathcal{L}(\theta_t, \lambda_t)) = \underbrace{(\mathcal{L}(\theta_{t+1}, \lambda_t) - \mathcal{L}(\theta_t, \lambda_t))}_{:=A_t} + \underbrace{(\mathcal{L}(\theta_{t+1}, \lambda_{t+1}) - \mathcal{L}(\theta_{t+1}, \lambda_t))}_{:=B_t},$$

we sum it over $t = 0, \ldots, T - 1$ gives

$$\sum_{t=0}^{T-1} \mathbb{E}[B_t] \geq -\Delta_{\mathcal{L}} - \sum_{t=0}^{T-1} \mathbb{E}[A_t].$$

By $L_{\mathcal{L}}$-smoothness, we have

$$A_t \leq G_{\max} \|\theta_{t+1} - \theta_t\| + \frac{L_{\mathcal{L}}}{2} \|\theta_{t+1} - \theta_t\|^2.$$

Taking expectation and applying Young's inequality yields $\mathbb{E}[A_t] \leq \frac{G_{\max}^2}{2} + \frac{1+L_{\mathcal{L}}}{2} \mathbb{E}\|\theta_{t+1} - \theta_t\|^2$. Thus,

$$\sum_{t=0}^{T-1} \mathbb{E}[A_t] \leq T \frac{G_{\max}^2}{2} + \frac{1 + L_{\mathcal{L}}}{2} \sum_{t=0}^{T-1} \mathbb{E}\|\theta_{t+1} - \theta_t\|^2.$$

Using $\mathbb{E}\left[\|\theta_{t+1} - \theta_t\|^2\right] \leq 2\alpha^2 \left(G_{\max}^2 + \sigma_\theta^2 + a^2\right)$, we have

$$\sum_{t=0}^{T-1} \mathbb{E}[A_t] \leq \frac{G_{\max}^2}{2} T + (1 + L_{\mathcal{L}}) T \alpha^2 \left(G_{\max}^2 + \sigma_\theta^2 + a^2\right).$$

Collecting terms dividing by βT, it yields

$$\begin{aligned} &\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[\|\mathcal{H}(\theta_t, \lambda_t)\|^2] \\ \leq &\frac{\Delta_{\mathcal{L}}}{\beta T} + C_3(U_{\max}^2 + \sigma_\lambda^2 + b^2) + C_4 \alpha^2 + \frac{G_{\max}^2}{2\beta} \\ &+ \frac{\alpha^2}{\beta} \left((1 + L_{\mathcal{L}})(G_{\max}^2 + \sigma_\theta^2 + a^2)\right). \end{aligned}$$

Finally, applying Jensen's inequality and the elementary bound completes the proof. □

VIII. Conclusion

This paper introduces a PPO-LTL framework that augments PPO with safety specifications expressed in LTL through the Lagrangian scheme, providing a precise way to encode complex safety requirements, such as regulatory rules. Theoretical guarantee on convergence is provided. Experiments in the Zones and CARLA environments are in full support of our method.

References

[1] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[2] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[3] J. García and F. Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, no. 42, pp. 1437–1480, 2015.

[4] A. Ray, J. Achiam, and D. Amodei, "Benchmarking safe exploration in deep reinforcement learning," in *Safe Machine Learning Workshop at NeurIPS*, 2019.

[5] C. Tennant, C. Neels, G. Parkhurst, P. Jones, S. Mirza, and J. Stilgoe, "Code, culture, and concrete: Self-driving vehicles and the rules of the road," *Frontiers in Sustainable Cities*, vol. 3, p. 710478, 2021.

[6] A. Pnueli, "The temporal logic of programs," *18th Annual Symposium on Foundations of Computer Science (SFCS 1977)*, pp. 46–57, 1977.

[7] C. Baier and J.-P. Katoen, *Principles of Model Checking*. MIT press, 2008.

[8] M. Y. Vardi, "Automata-theoretic model checking revisited," in *Verification: Theory and Practice*. Springer, 1996, pp. 137–150.

[9] S. Sickert, J. Esparza, B. Finkbeiner, and M. Leucker, "Limit-deterministic büchi automata for linear temporal logic," in *Computer Aided Verification*. Springer, 2016, pp. 312–332.

[10] J. Ji, J. Zhou, B. Zhang, J. Dai, X. Pan, R. Sun, W. Huang, Y. Geng, M. Liu, and Y. Yang, "Omnisafe: An infrastructure for accelerating safe reinforcement learning research," *Journal of Machine Learning Research*, vol. 25, no. 285, pp. 1–6, 2024.

[11] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[12] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. Pmlr, 2018, pp. 1861–1870.

[13] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, and U. Topcu, "Safe reinforcement learning via shielding," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[14] N. Jansen, B. Könighofer, S. Junges, A. Serban, and R. Bloem, "Safe reinforcement learning using probabilistic shields," in *31st International Conference on Concurrency Theory (CONCUR 2020)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020, pp. 3–1.

[15] E. Altman, *Constrained Markov Decision Processes*. CRC Press, 2021.

[16] A. Camacho, R. T. Icarte, T. Q. Klassen, R. A. Valenzano, and S. A. McIlraith, "Ltl and beyond: Formal languages for reward function specification in reinforcement learning," in *IJCAI*, vol. 19, 2019, pp. 6065–6073.

[17] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *International symposium on formal techniques in real-time and fault-tolerant systems*. Springer, 2004, pp. 152–166.

[18] P. Y. Chan, X. Li, Y. Lu, Y. Lin, and A. Bundy, "Formalise regulations for autonomous vehicles with right-open temporal deontic defeasible logic," in *International Conference on Innovative Techniques and Applications of Artificial Intelligence*. Springer, 2024, pp. 194–207.

[19] L. De Raedt and A. Kimmig, "Probabilistic (logic) programming concepts," *Machine Learning*, vol. 100, no. 1, pp. 5–47, 2015.

[20] J. Chen, S. E. Li, and M. Tomizuka, "Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5068–5078, 2021.

[21] S. Qi, Z. Zhang, Z. Sun, and S. Haesaert, "Risk-aware autonomous driving with linear temporal logic specifications," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025, pp. 14 877–14 883.

[22] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA: MIT Press, 2018. [Online]. Available: http://incompleteideas.net/book/the-book-2nd.html

[23] Y. Li, F. He, M. Xue, *et al.*, "Temporal logic guided safe reinforcement learning," in *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, 2023.

[24] P. Vaezipoor, A. C. Li, R. A. T. Icarte, and S. A. Mcilraith, "Ltl2action: Generalizing ltl instructions for multi-task rl," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10 497–10 508.
[25] M. Jackermeier and A. Abate, "Deepltl: Learning to efficiently satisfy complex ltl specifications for multi-task rl," in *The Thirteenth International Conference on Learning Representations*, 2025.
[26] J. Baxter and P. L. Bartlett, "Infinite-horizon policy-gradient estimation," *journal of artificial intelligence research*, vol. 15, pp. 319–350, 2001.
[27] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *International conference on machine learning*. Pmlr, 2017, pp. 22–31.
[28] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," in *Journal of Machine Learning Research*, vol. 22, 2021, pp. 1–8.