Safe Driving via Expert Guided Policy Optimization

Anonymous Author(s)
Affiliation
Address
email

Abstract: When learning common skills like driving, beginners usually have experienced people or domain experts standing by to ensure the safety of the learning process. We formulate such learning scheme under the Expert-in-the-loop Reinforcement Learning (ERL) where a guardian is introduced to safeguard the exploration of the learning agent. While allowing the sufficient exploration in the uncertain environment, the guardian will intervene under dangerous situations and demonstrate the correct actions to avoid the potential accident. Thus ERL enables both exploration and expert's partial demonstration as two training data sources. Following such new setting, we develop a novel Expert Guided Policy Optimization (EGPO) method. This method integrates the guardian in the loop of reinforcement learning, which is composed of an expert policy to generate demonstration and a switch function to decide when to intervene. Particularly, constrained optimization technique is used to tackle the trivial solution that the agent deliberately behaves dangerously to deceive the expert into taking over all the time. Offline RL technique is further used to learn from the partial demonstrations generated by the expert. Safe driving experiments show that our method achieves superior training and test-time safety, outperforms baselines with a large margin in sample efficiency, and preserves the generalization capacity to unseen environments in test-time [1].

Keywords: Safe Reinforcement Learning, Human-in-the-loop, Imitation Learning

1 Introduction

Reinforcement Learning (RL) shows promising results in human-interactive applications ranging from autonomous driving [1], the power system in smart building [2], to the surgical robotics arm [3]. However, training and test time safety remains as a great concern for the real-world applications of RL. This problem draws significant attention since the agent needs to explore the environment sufficiently in order to optimize its behaviors. It might be inevitable for the agent to experience dangerous situations before it can learn how to avoid them [4], even the training algorithms are equipped with advanced techniques to reduce the probability of failures [5, 6, 7].

We human beings do not learn purely from trial-and-error exploration, for the sake of safety as well as efficiency. In daily life, when learning some common skills like driving, we usually ensure the safety by involving domain expert to safeguard our learning process. The expert not only demonstrates the correct actions but also acts as a *guardian* to allow our own safe exploration of the uncertain environment. For example as illustrated in Fig.1, when learning to drive, the student with the learner's permit can directly operate the vehicle while the instructor stands by. When a risky situation happens, the instructor takes over the vehicle to avoid the potential accident. Thus the student can learn how to handle tough situations both from the exploration and the instructor's demonstrations.

In this work, we aim to formulate such learning scheme with Expert-in-the-loop RL (ERL). As shown in the right panel of Fig.1, ERL incorporates the component of the *guardian* in the interaction between agent and environment. The guardian contains a *switch* mechanism and an *expert* policy. The switch decides to intervene the free exploration of the agent in the situations when the agent is conducting unreasonable behaviors or a potential critical failure is happening. In those cases the expert takes over the main operation and starts providing demonstrations on solving the task or avoiding dangers. Our

[1] Source code of our method and the environment will be made publicly available.

Submitted to the 5th Conference on Robot Learning (CoRL 2021). Do not distribute.

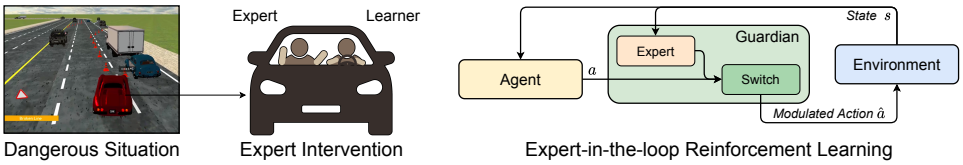

Figure 1: The expert intervenes the learner when dangerous situation happens. We model it through the Expert-in-the-loop RL scheme on the right panel where a guardian is introduced in the interaction between agent and environment.

setting of ERL extends previous works of Human-in-the-loop RL in two ways: First, the guardian inspects the exploration all the time and actively intervenes if necessary, instead of passively advising which action is good [8] or evaluating the collected trajectories after the agent rolling out [9, 10]. This feature guarantees the safe exploration in training time. Second, the guardian does not merely intervene the exploration and terminate the episode [11], instead, it demonstrates to the learning agent the correct actions under risky states to avoid accidents. Thus those demonstrations become more effective training data to the learning agent.

Following the setting of ERL, we develop a novel method called ***E**xpert **G**uided **P**olicy **O**ptimization (**EGPO**)*. EGPO addresses two challenges in ERL. First, the learning agent may abuse the guardian and consistently causes intervention so that it can exploit the high performance and safety of the expert. To tackle this issue, we impose the Lagrangian method on the policy optimization to limit the intervention frequency. Moreover, we apply the PID controller to update the Lagrangian multiplier, which substantially improves the dual optimization with off-policy RL algorithm. The second issue is the partial demonstration data collected from the guardian. Since those data is highly off-policy to the learning agent, we introduce offline RL technique into EGPO to stabilize the training with the off-policy partial demonstration. The experiments show that our method can achieve superior training safety while yields a well-performing policy with high safety in the test time. Furthermore, our method exhibits better generalization performance compared to previous methods.

As a summary, the main contributions of this work are: (1) We formulate the Expert-in-the-loop RL (ERL) framework that incorporates the guardian as a demonstrator as well as a safety guardian. (2) We develop a novel ERL method called ***E**xpert **G**uided **P**olicy **O**ptimization (**EGPO**)* with a practical implementation of guardian mechanism and learning pipeline. (3) Experiments show that our method achieves superior training and test safety, outperforms baselines with a large margin in sample efficiency, and generalizes to unseen environments in test time.

2 Related Work

Safe RL. Learning RL policy under safety constraints [12, 13, 7] becomes an important topic in the community due to the safety concern in real-world applications. Many methods based on constrained optimization have been developed, such as the trust region methods [5], Lagrangian methods [5, 6, 14], barrier methods [15, 16, 17], Lyapunov methods [18, 4, 19], *etc.*. Another direction is based on the safety critic, where an additional value estimator is learned to predict cost, apart from the primal critic estimating the discounted return [20, 7, 21]. Saunders et al. [11] propose HIRL, a scheme for safe RL requiring extra manual efforts to demonstrate and train an imitation learning decider who intervenes the endangered agent. Differently, in our work the guardian does not terminate the exploration but instead continues the trajectory with expert to demonstrate the proper actions to take. However, majority of the aforementioned methods hold the issue that only the upper bound of failure probability of the learning agent can be guaranteed theoretically, but there is no mechanism to explicitly ensure the occurrence of the critical failures. Dalal et al. [22] equip the policy network with a safety layer that can modulate the output action as an absolutely safe one, when the assumption that cost function is the linear transformation of the action is hold. The proposed EGPO utilizes the guardian to ensure safe exploration without assuming the structure of the cost function.

Learning from Demonstration. Many works consider leveraging the collected demonstrations to improve policy. Behavior Cloning (BC) [23] uses supervised learning to fit the policy function to produce the same action as the expert. GAIL [24] and SQIL [25] ask the learning agent to execute in the environment and collect trajectories to evaluate the divergence between the agent and the expert. This exposes the agent to possible dangerous states. DAgger [26] periodically queries the

expert for new demonstration and is successfully applied to extensive domains [27, 28, 29]. Similar to our idea, the Human-Gated DAgger (HG-DAgger) [28] utilizes an expert to intervene exploration and carry the agent to safe states before giving back the control. However, HG-Dagger does not impose constraints to reduce human intervention and does not utilize the data in free exploration. Recently, offline RL draws wide attention which learns policy from the dataset generated by arbitrary policies [30, 31, 32]. The main challenge of offline RL is the out-of-distribution (OOD) actions [31]. Conservative Q-Learning (CQL) [33] addresses the impact of OOD actions by learning a conservative Q-function to estimate the lower bounds of true Q values. In this work, we use the CQL technique to improve the training on the trajectories with partial demonstrations given by the guardian.

Human-in-the-loop RL. An increasing number of works focus on incorporating human into the training loop of RL. The human is responsible for evaluating the trajectories sampled by the learning agent [9, 34, 10], or being a consultant to guide which action to take when the agent requests [8]. Besides, the human can also actively monitor the training process, such as deciding whether to terminate the episode if potential danger is happening [35, 11]. Our work is derived from the Human-in-the-loop framework where the guardian plays the role of human expert to provide the feedback to the learning agent. However, for previous human-in-the-loop RL works it is much less explored about how to optimize the agent to minimize interventions and efficiently utilize the data generated by the expert, which will be addressed in this work.

3 Expert Guided Policy Optimization

Extending the setting of Human-in-the-loop RL, we frame the *Expert-in-the-loop RL* (ERL) that incorporates the guardian to ensure training safety and improve efficiency. We develop a novel method called ***E**xpert **G**uided **P**olicy **O**ptimization (**EGPO**)* to implement the guardian mechanism. We will introduce the overview and the method in detail as follows.

3.1 Overview of the Guardian Mechanism

We take learning to drive as a motivating example. Generally speaking, the student driver learns the skills of driving from the instructor through two approaches: (1) *Student learns from instructor's demonstrations*. At the early stage of training, the student observes the demonstrations given by the instructor and learns rapidly by imitating the behaviors. Besides, the student also learns how the expert tackles dangerous situations; (2) *Student in driver seat operates the vehicle in an exploratory way while the instructor serves as guardian*. The instructor only conducts takeover of the vehicle when necessary in dangerous situations, otherwise the student can explore freely. Therefore, we can see that the student learns to drive from both imitation and exploration.

Based on this motivating example, we have the framework of *Expert-in-the-loop RL* (ERL). As illustrated in Fig. 1, on top of the conventional RL scheme, we introduce the component of *guardian*, which resembles the instructor who not only provides high-quality demonstrations to accelerate the learning, but also safeguards the exploration of agent in the environment. In the proposed EGPO method, the guardian is composed of two parts: an expert and a switch function.

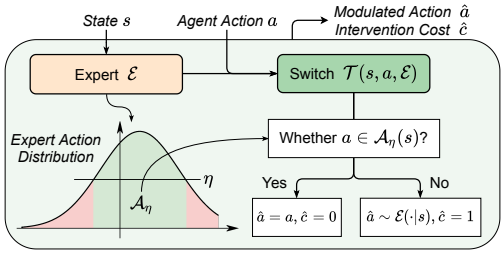

Figure 2: Flowchart of the guardian mechanism.

The **expert** policy $\mathcal{E} : a^E \sim \mathcal{E}(\cdot|s)$ can output safe and reliable actions a^E in most of the time. Besides, it can provides the probability of taking action a produced by the agent: $\mathcal{E}(a|s) \in [0, 1]$. This probability reflects the agreement of the expert on the agent's action, which serves as an indicator for intervention in the switch function. We assume the access to such well-performing expert policy. The **switch** is another part of guardian, which decides under which state and timing the expert should intervene and demonstrate the correct actions to the learning policy. As shown in Fig. 2, the switch function $\mathcal{T}$ considers the agent action as well as the expert and outputs the modulated action $\hat{a}$ fed to the environment and the intervention occurrence $\hat{c}$:

$$\mathcal{T}(s, a, \mathcal{E}) = (\hat{a}, \hat{c}) = \begin{cases} (a^E \sim \mathcal{E}(\cdot|s), 1), & \text{if } a \in \mathcal{A}_\eta(s) \\ (a, 0), & \text{otherwise,} \end{cases} \tag{1}$$

wherein η is the confidence level on the expert action probability and $\mathcal{A}_\eta(s) = \{a \in \mathcal{A} : \mathcal{E}(a|s) \geq \eta\}$ is the *safe action space*, relating to the expert policy and the confidence level.

The switch mechanism leads to the formal representation of the behavior policy:

$$\hat{\pi}(a|s) = \pi(a|s)\mathbf{1}_{a\in\mathcal{A}_\eta(s)} + \mathcal{E}(a|s)F(s), \tag{2}$$

wherein $F(s) = \int_{a'\notin\mathcal{A}_\eta(s)} \pi(a'|s)da'$ is a function which denotes the probability of choosing an action that will be rejected by the switch. Emulating how human drivers judge the risky situations, we rely on the expert's evaluation of the safety during training, instead of any external or objective criterion.

We derive the guarantee on the training safety from the introduction of guardian. We first have the assumption of expert:

Assumption 1 (Failure probability of the expert)**.** *For all state, the step-wise probability of expert producing unsafe action is bounded by a small value $\epsilon < 1$: $\mathbb{E}_{a\sim a^E(\cdot|s)} I(s,a) \leq \epsilon$, wherein $I(s,a) \in \{0,1\}$ is a boolean denotes whether next state $s' \sim \mathcal{P}(s'|s,a)$ is an unsafe state.*

We use the expected cumulative probability of failure to measure the expected risk encountered by the behavior policy: $\hat{V} = \mathbb{E}_{s_0}\hat{V}(s_0) = \mathbb{E}_{s_0,\tau\sim P(\hat{\pi})}\sum_{t=0}^{\infty}\gamma^t I(s_t,a_t)$ wherein $P(\hat{\pi})$ refers to the trajectory distribution deduced by the behavior policy. We propose the main theorem of this work:

Theorem 1 (Upper bound of training risk)**.** *The expected cumulative probability of failure $\hat{V}$ of the behavior policy $\hat{\pi}$ in EGPO is bounded by the step-wise failure probability of the expert ϵ as well as the confidence level η:*

$$\hat{V} \leq \frac{\epsilon}{1-\gamma}(1 + \frac{1}{\eta} + \frac{\gamma}{1-\gamma}K'_\eta),$$

*wherein $K'_\eta = \max_s \int_{a\in\mathcal{A}_\eta(s)} da$ is **negative correlated** to η.*

This theorem indicates that the training risk is bounded by the failure probability of expert ϵ, as well as the confidence level η. When ϵ is fixed, increasing the confidence level will shrink the upper bound of $\hat{V}$, leading to better training safety. The proof is given in the Appendix.

In the implementation, the actions from agent are firstly modulated by the guardian and the safe actions will be applied to the environment. The collected data in execution is used to update the agent. We update the learning agent with off-policy RL algorithm. Meanwhile, we also leverage a recent offline RL technique to tackle the partial demonstrations provided by the guardian and further improve the learning stability. The policy learning is presented in Sec. 3.2. Besides, since the intervention from guardian indicates the agent has done something wrong, we also optimize the policy to reduce intervention frequency through the constrained optimization in Sec. 3.3.

3.2 Learning Policy from Exploration and Partial Demonstration

The proposed EGPO method can work with most of the RL algorithms to train the safe policy since the guardian mechanism does not impose any assumption on the underlying RL method. In this work, we use an off-policy actor-critic method Soft Actor-Critic (SAC) [36] to train the agent. The method utilizes two neural networks including a Q network estimating the state-action value: Q_ϕ, and a policy network: π_θ. ϕ and θ are the parameters. The training algorithm alternates between the policy evaluation and the policy improvement in each iteration. The policy evaluation process updates the estimated Q function by minimizing the L2 norm of the entropy regularized TD error:

$$\begin{aligned} y(r_t, s_{t+1}) &= r_t + \gamma \mathop{\mathbb{E}}_{a_{t+1}\sim\hat{\pi}(\cdot|s_{t+1})} [Q_{\bar{\phi}}(s_{t+1}, a_{t+1}) - \alpha \log \pi_\theta(a_{t+1}|s_{t+1})], \\ L_Q(\phi) &= \frac{1}{2} \mathop{\mathbb{E}}_{(s_t,a_t,r_t,s_{t+1})\sim\mathcal{B}} [y(r_t, s_{t+1}) - Q_\phi(s_t, a_t)]^2. \end{aligned} \tag{3}$$

Here $\mathcal{B}$ is the replay buffer, $\bar{\phi}$ is the delayed parameters, α is a temperature parameter. On the other hand, the policy improvement objective is written as:

$$L_\pi(\theta) = -\mathop{\mathbb{E}}_{(s_t,a_t)\sim\mathcal{B}} [Q_\phi(s_t, a_t) - \alpha \log \pi_\theta(a_t|s_t)]. \tag{4}$$

Since we use a mixed safety-ensured policy $\hat{\pi}$ to explore the environment, part of the collected transitions contain the actions from the expert. This part of data comes as *partial demonstration* to improve the agent, denoted as $\mathcal{B}^{\mathcal{E}}$. We need to incorporate distributional shift technique to overcome such off-policy data issue. Many works have been proposed to overcome this problem, such as the V-trace in on-policy algorithm IMPALA [37], the advantage-weighted actor-critic [38] in off-policy algorithm, and many other offline RL methods [32, 31, 33]. To train with the off-policy data produced by the guardian, we employ the recent Conservative Q-Learning (CQL) [33], known as an effective offline RL method, into our *Learning from Partial Demonstration (LfPD)* setting, and then the objective to update Q function becomes:

$$L_Q^{\text{LfPD}}(\phi) = \beta(\underbrace{\mathop{\mathbb{E}}_{s\sim\mathcal{B}^{\mathcal{E}},a\sim\pi}[Q_\phi(s,a)]}_{\text{1st Term}} - \underbrace{\mathop{\mathbb{E}}_{s\sim\mathcal{B}^{\mathcal{E}},a\sim\mathcal{E}}[Q_\phi(s,a)]}_{\text{2nd Term}}) + \underbrace{\frac{1}{2}\mathop{\mathbb{E}}_{(s,a)\sim\mathcal{B}}[y(r_t,s_{t+1}) - Q_\phi(s_t,a_t)]^2}_{\text{3rd Term}}. \tag{5}$$

Note that the 1st Term and 2nd Term are expectations over only the partial demonstration $\mathcal{B}^{\mathcal{E}}$, instead of the whole batch $\mathcal{B}$. In the partial demonstration data, the 1st Term reduces the Q values for the actions taken by the agent, while the 2nd Term increases the Q values for the actions taken by the expert. The 3rd term is the original TD learning objective in Eq. 3. CQL reflects such an idea: be conservative to the actions sampled by the agent, and optimistic to the actions sampled by the expert. Minimizing the objective Eq. 5 can lead to a better and stabler Q function. In next session, we discuss a new issue emerged in the ERL that damages the training, and propose our solution to the problem.

3.3 Intervention Minimization via Constrained Optimization

The guardian intervenes the exploration of the agent once it behaves dangerously or inefficiently in current state. However, if no action is taken to limit intervention frequency, the learning policy is prone to heavily relying on the guide and protection of the guardian, so that the guardian would always take over. In this case, the learning policy receives high reward under the supervision of guardian but fails to finish tasks independently.

In this section, we consider the intervention minimization as a constrained optimization problem and apply the Lagrangian method into the policy improvement process. Though the SAC with the Lagrangian method has been proposed in [39], we find that directly optimizing the Lagrangian dual is highly unstable with off-policy RL algorithm. Stooke et al. [6] analyze that the update of the Lagrangian multiplier, from a perspective of control theory, is an integral control. Introducing extra proportional and derivative control to update the Lagrangian multiplier can reduce the oscillations and cost violations. In our case, we apply the PID control of Lagrangian multipliers with an offline RL algorithm.

Concretely, we first turn the problem of intervention minimization into a constrained optimization problem: $\theta^* = \arg\max_\theta \mathbb{E}_{\pi_\theta}[\sum_{t=0}\gamma^t r_t]$, s.t. $\mathbb{E}_{\pi_\theta}[\sum_{t=0}\gamma^t \hat{c}_t] \leq C$ wherein C is the intervention frequency limit in one episode. The Lagrangian dual form of the above problem becomes an unconstrained optimization problem with a penalty term:

$$\theta^* = \arg\max_\theta \min_{\lambda\geq 0} \mathop{\mathbb{E}}_{\tau\sim\hat{\pi}}\{(\sum_{t=0}\gamma^t r_t) - \lambda[(\sum_{t=0}\gamma^t \hat{c}_t) - C]\}, \tag{6}$$

where $\lambda \geq 0$ is known as the Lagrangian multiplier. The optimization over θ and λ can be conducted iteratively between policy gradient ascent and stochastic gradient descent (SGD).

We additionally introduce an intervention critic Q_ψ^C to estimate the cumulative intervention occurrence $\sum_{t'=t}\gamma^{(t-t')}\hat{c}_{t'}$. This network can be optimized by following Eq. 3 but replace the reward with the intervention occurrence. Now we can write the intervention minimization objective $L_\pi^\lambda(\theta)$ and the practical form of Eq. 6 to update the policy $L'_\pi(\theta)$ as:

$$L_\pi^\lambda(\theta) = -\mathop{\mathbb{E}}_{(s_t,a_t)\sim\mathcal{B}}[Q_\psi^C(s_t,a_t) - C], \tag{7}$$

$$L'_\pi(\theta) = L_\pi(\theta) - \lambda L_\pi^\lambda(\theta). \tag{8}$$

Conducting SGD on Eq. 8 w.r.t. the policy parameter θ can improve the return while reduce the intervention. As reported in [6] that optimizing Lagrangian multiplier exhibits oscillations and

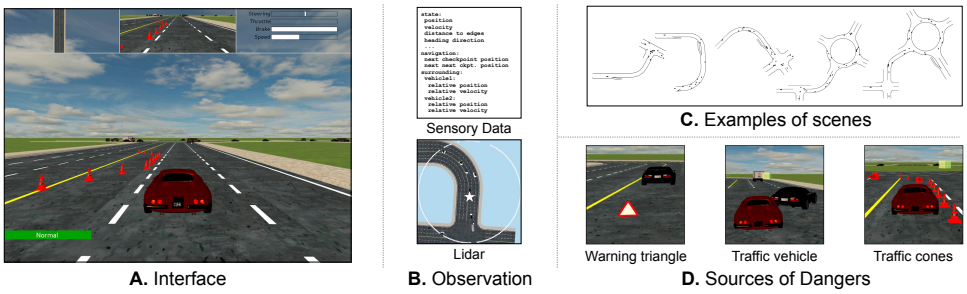

Figure 3: **A**. The interface of our environment. **B**. The observations feeding to the target vehicle. **C**. The examples of the scenes we use in training and test. **D**. The three events creating costs: crashing with warning triangle, cone or other vehicles. $+1$ cost is given once those events occur.

overshoot that destabilize the policy learning, thus we adopt a PID controller to update λ and form the responsive intervention minimization:

$$\lambda \leftarrow K_p\delta + K_i \int_{i=1}^{k} \delta di + K_d \frac{\delta}{di}, \quad \text{wherein} \quad \delta = \mathbb{E}_{\tau}[\sum_{t=0} \hat{c}_t] - C, \tag{9}$$

where we denote the training iteration as i. K_p, K_i, K_d are the hyper-parameters. Optimizing λ with Eq. 6 reduces to the proportional term in Eq. 9, while the integral and derivative terms compensate the accumulated error and overshoot in the intervention occurrence. We apply the PID controller in EGPO, as well as the baseline SAC-Lagrangian method in the experiments. Empirical results support that PID control on λ can bring more stable learning dynamics and hyperparameter robustness.

4 Experiments

4.1 Experimental Settings

Environment. We evaluate the proposed method and baselines in the autonomous driving simulator PGDrive [40]. The environment supports generating an unlimited number of scenes via the Procedural Generation. Each of the scenes includes the vehicle agent, the complex road network, the dense traffic flow, and many obstacles such as cones and warning triangles, as shown in Fig. 3**D**. The task for the agent is to steer the target vehicle with low-level signal, namely acceleration, brake and steering, to reach the predefined destination. Each collision to the traffic vehicles or obstacles yields $+1$ cost. The cumulative cost across one episode is the measurement on the safety of a policy, which is independent of the expert being used or not during training. The reward function only contains a dense driving reward and a sparse terminal reward. The dense reward is the longitudinal movement toward destination in Frenet coordinates. The sparse reward $+20$ is given when the agent arrives the destination. We build our testing benchmark based on PGDrive rather than other RL environments like the safety gym [41] or Mujoco [42] because we target on the application of autonomous driving and the generalization of the RL methods. Different to those environments, PGDrive can generate an infinite number of driving scenes which allows evaluating the generalization of different methods in the context of safe RL.

Split of training and test sets. Different from the conventional RL setting where the agent is trained and tested in the same fixed environment, we focus on evaluating the generalization through testing performance. We split the scenes into the training set and test set with 100 and 50 different scenes respectively. At the beginning of each episode, a scene in the training or test set is randomly selected. After each training iteration, we roll out the learning agent without guardian in the test environments and record the percentage of successful episodes over 30 evaluation episodes, called *success rate*. Besides, we also record the cost given by the environment and present the cumulative cost in each episode in following tables and figures.

Training expert policy. In our experiment, the expert policy is a stochastic policy trained from the Lagrangian PPO [41] with batch size as large as 160,000 and a long training time. To further improve the performance of the expert, we have reward engineering by doubling the cost and adding penalty for dangerous actions.

Table 1: The test performance of different approaches.

Category	Method	Cumulative Reward	Cumulative Cost	Success Rate
Expert	*PPO-Lag*	*392.38* $\pm$*99.47*	*1.26*$\pm$*0.57*	*0.86*$\pm$*0.05*
RL	SAC-RS	346.49 $\pm$16.51	8.68 $\pm$3.34	0.68 $\pm$0.10
	PPO-RS	294.10 $\pm$22.28	3.93 $\pm$4.19	0.41$\pm$0.09
Safe RL	SAC-Lag	333.90 $\pm$19.00	2.21 $\pm$1.08	0.65 $\pm$0.14
	PPO-Lag	288.04 $\pm$53.72	1.03 $\pm$0.34	0.43 $\pm$0.21
	CPO	194.06 $\pm$108.86	1.71 $\pm$1.02	0.21 $\pm$0.29
Offline RL	CQL	373.95 $\pm$8.89	0.24 $\pm$0.30	0.72 $\pm$0.11
IL	BC	362.18 $\pm$6.39	**0.13** $\pm$**0.17**	0.57 $\pm$0.12
	Dagger	346.16 $\pm$22.62	0.67 $\pm$0.23	0.66 $\pm$0.12
	GAIL	309.66 $\pm$12.47	0.68 $\pm$0.20	0.60 $\pm$0.07
Ours	EGPO	**388.37** $\pm$**10.01**	0.56 $\pm$0.35	**0.85** $\pm$**0.05**

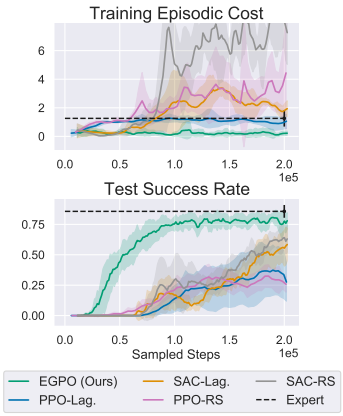

Figure 4: Comparison between our method and safe RL baselines.

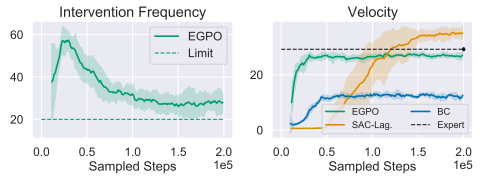

Figure 5: The learning dynamics of EGPO and baseline methods during training.

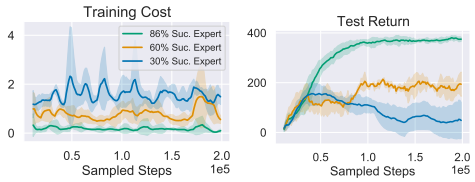

Figure 6: The curves of EGPO agents when varying the quality of experts.

Implementation details. We conduct experiments on PGDrive and aforementioned algorithms using RLLib [43], a distributed learning system which allows large-scale parallel experiments. Generally, we host 8 concurrent trials in an Nvidia GeForce RTX 2080 Ti GPU. Each trial consumes 2 CPUs with 8 parallel rollout workers. Each trial is trained over roughly $200,000$ environmental steps, which corresponds to 11 hours of individual driving experience. All experiments are repeated 5 times with different random seeds. Information about other hyper-parameters is given in Appendix.

4.2 Results

Compared to RL and Safe RL baselines. We evaluate two RL baselines PPO [44] and SAC [36], with the reward shaping (RS) method that considers negative cost as auxiliary reward. We also evaluate three safe RL methods, namely the Lagrangian version of PPO and SAC [6, 39] and CPO [5]. As shown in Fig. 4 and Table 1, EGPO shows superior training and test time safety compared to the baselines. During training, EGPO limits the occurrence of dangers, denoted by the episodic cost, to almost zero. Noticeably, EGPO achieves lower cost compared to the expert policy. EGPO also learns rapidly and results to a high test success rate.

Compared to Imitation Learning and Offline RL baselines. We use the expert to generate $250,000$ steps of transitions from training environments and use this dataset to train with Behavior Cloning (BC), GAIL [24], DAgger [26], and offline RL method CQL [33]. As shown in Table 1, EGPO yields better test time success rate compared to the imitation learning baselines. BC outperforms ours in test time safety, but we find that BC agent learns conservative behaviors resulting in the average velocity to 15.05 km/h, while EGPO runs normally in 27.52 km/h, as shown in Fig. 5.

Learning dynamics. We denote the intervention frequency by the averaged episodic intervention occurrence $\mathbb{E}\sum_t \hat{c}_t$. As illustrated in Fig. 5, at the beginning of the training, the guardian is involved more frequently to provide driving demonstrations and prevent agent from entering dangerous states. After acquiring primary driving skills, the agent is prone to choosing actions that are more acceptable by guardian and thus the takeover frequency decreases along with the increasing of training steps.

Table 2: The test performance when ablating components in EGPO.

Experiment	Cumulative Reward	Cumulative Cost	Success Rate
(a) W/ rule-based switch	339.10$\pm$11.41	0.91$\pm$0.60	0.57$\pm$0.09
(b) W/o intervention min.	38.31$\pm$3.61	1.00$\pm$0.00	0.00$\pm$0.00
(c) W/o PID in SAC-Lag.	338.80 $\pm$16.23	0.59$\pm$0.40	0.67$\pm$0.10
(d) W/o CQL loss	378.00 $\pm$6.77	0.43 $\pm$0.54	0.80 $\pm$0.08
(e) W/o environmental reward	379.91$\pm$7.87	**0.43$\pm$0.26**	0.79$\pm$0.06
EGPO	**388.37** $\pm$**10.01**	0.56 $\pm$0.35	**0.85** $\pm$**0.05**

Figure 7: Ablation study on η.

4.3 Ablation Studies

The impact of expert quality. To investigate the impact of the expert if its quality is not as good as the well-performing expert used in the main experiments, we involve two experts with 60% and 30% test success rate into the training of EGPO. Those two experts are retrieved from the intermediate checkpoints when training the expert. The result of training EGPO with the inferior experts is shown in Fig. 6. We can see that improving the expert's quality can reduce the training cost. This result also justifies empirically the Theorem 1 where the training safety is bounded by the expert safety. Besides, we find better expert leads to better EGPO agent in term of the cumulative return. We hypothesize this is because using premature policies as expert will lead the switch function to produce chaotic intervention signals that mystifies the exploration.

The impact of confidence level. The confidence level η is a hyper-parameter. As shown in Fig. 7, we find that when $\eta > 0.05$, the performance decreases as η increases. This is because higher η means less freedom of free exploration. In the extreme case where $\eta = 1.0$, all data is collected by the expert. At this time, the intervention minimization multiplier λ will goes to large value, which damages the training. When $\eta = 0.0$, the whole algorithm reduces to vanilla SAC.

Ablations of the guardian mechanism. **(a)** We adopt a rule-based switch design to validate the effectiveness of the statistical switch in Sec. 3.1. For example, the intervention happens when the distance to the nearby vehicles or to the boundary of road is too small. We find that the statistical switch performs better than rules. This is because it is hard to enumerate manual rules that cover all possible dangerous situations. **(b)** Removing the intervention minimization technique, the takeover frequency becomes extremely high and the agent learns to drive directly toward the boundary of the road. This causes consistently the out-of-the-road failures, resulting in the zero success rate and 1 episodic cost. This result shows the importance of the intervention minimization in Sec. 3.3. **(c)** We find that removing the PID controller on updating λ in intervention minimization causes a highly unstable training. It is consistent with the result in [6]. We therefore need to use PID controller to optimize λ in EGPO and SAC-Lag. **(d)** Removing CQL loss in Eq. 5 damages the performance. We find this ablation reduces the training stability. **(e)** We disable the environment reward in EGPO, so that the only supervision signal to train the policy is the intervention occurrence. This method outperforms IL baselines with a large margin, but remains lower than EGPO in the return and success rate. This suggests EGPO framework can be turned into a practical Imitation Learning method.

Human-in-the-loop experiment. To demonstrate the potential of EGPO, we conduct a human-in-the-loop experiment, where a human expert supervises the learning progress of the agent in a map. The evaluation results suggest that EGPO can achieve 90% success rate with merely 15,000 environmental steps of training, while SAC-Lag takes 185,000 steps to achieve similar results. EGPO also outperforms Behavior Cloning method which even consumes more human data in a large margin. Please refer to Appendix and supplementary video for details.

5 Conclusion

We develop a Expert Guided Policy Optimization method for the Expert-in-the-loop reinforcement learning. The method incorporates the guardian mechanism in the interaction of agent and environment to ensure safe and efficient exploration. The experiments on safe driving show that the proposed method can achieve training and test-time safety and outperform previous safe RL and imitation baselines. In future work we will explore the potential of involving human to provide feedback in the learning process.

References

[1] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah. Learning to drive in a day. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8248–8254. IEEE, 2019.

[2] K. Mason and S. Grijalva. A review of reinforcement learning for autonomous building energy management. *Computers & Electrical Engineering*, 78:300–312, 2019.

[3] F. Richter, R. K. Orosco, and M. C. Yip. Open-sourced reinforcement learning environments for surgical robotics. *arXiv preprint arXiv:1903.02090*, 2019.

[4] Y. Chow, O. Nachum, E. Duenez-Guzman, and M. Ghavamzadeh. A lyapunov-based approach to safe reinforcement learning. In *Advances in neural information processing systems*, pages 8092–8101, 2018.

[5] J. Achiam, D. Held, A. Tamar, and P. Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 22–31. JMLR. org, 2017.

[6] A. Stooke, J. Achiam, and P. Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, pages 9133–9143. PMLR, 2020.

[7] H. Bharadhwaj, A. Kumar, N. Rhinehart, S. Levine, F. Shkurti, and A. Garg. Conservative safety critics for exploration. *arXiv preprint arXiv:2010.14497*, 2020.

[8] T. Mandel, Y.-E. Liu, E. Brunskill, and Z. Popović. Where to add actions in human-in-the-loop reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[9] P. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. *arXiv preprint arXiv:1706.03741*, 2017.

[10] L. Guan, M. Verma, S. Guo, R. Zhang, and S. Kambhampati. Explanation augmented feedback in human-in-the-loop reinforcement learning. *arXiv preprint arXiv:2006.14804*, 2020.

[11] W. Saunders, G. Sastry, A. Stuhlmueller, and O. Evans. Trial without error: Towards safe reinforcement learning via human intervention. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2067–2069. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

[12] J. Garcıa and F. Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

[13] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[14] D. A. Calian, D. J. Mankowitz, T. Zahavy, Z. Xu, J. Oh, N. Levine, and T. Mann. Balancing constraints and rewards with meta-gradient d4pg. *arXiv preprint arXiv:2010.06324*, 2020.

[15] A. Taylor, A. Singletary, Y. Yue, and A. Ames. Learning for safety-critical control with control barrier functions. In *Learning for Dynamics and Control*, pages 708–717. PMLR, 2020.

[16] Y. Liu, J. Ding, and X. Liu. Ipo: Interior-point policy optimization under constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4940–4947, 2020.

[17] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3387–3395, 2019.

[18] T. J. Perkins and A. G. Barto. Lyapunov design for safe reinforcement learning. *Journal of Machine Learning Research*, 3(Dec):803–832, 2002.

[19] H. Sikchi, W. Zhou, and D. Held. Lyapunov barrier policy optimization. *NeurIPS Deep Reinforcement Learning Workshop 2020*, 2020.

[20] J. Zhang, B. Cheung, C. Finn, S. Levine, and D. Jayaraman. Cautious adaptation for reinforcement learning in safety-critical settings. In *International Conference on Machine Learning*, pages 11055–11065. PMLR, 2020.

[21] K. Srinivasan, B. Eysenbach, S. Ha, J. Tan, and C. Finn. Learning to be safe: Deep rl with a safety critic. *arXiv preprint arXiv:2010.14603*, 2020.

[22] G. Dalal, K. Dvijotham, M. Vecerik, T. Hester, C. Paduraru, and Y. Tassa. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

[23] B. Widrow. Pattern recognition and adaptive control. *IEEE Transactions on Applications and Industry*, 83(74):269–277, 1964.

[24] J. Ho and S. Ermon. Generative adversarial imitation learning. *arXiv preprint arXiv:1606.03476*, 2016.

[25] S. Reddy, A. D. Dragan, and S. Levine. Sqil: Imitation learning via reinforcement learning with sparse rewards. *arXiv preprint arXiv:1905.11108*, 2019.

[26] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

[27] S. Ross, N. Melik-Barkhudarov, K. S. Shankar, A. Wendel, D. Dey, J. A. Bagnell, and M. Hebert. Learning monocular reactive uav control in cluttered natural environments. In *2013 IEEE international conference on robotics and automation*, pages 1765–1772. IEEE, 2013.

[28] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer. Hg-dagger: Interactive imitation learning with human experts. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8077–8083. IEEE, 2019.

[29] J. Zhang and K. Cho. Query-efficient imitation learning for end-to-end autonomous driving. *arXiv preprint arXiv:1605.06450*, 2016.

[30] S. Levine, A. Kumar, G. Tucker, and J. Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

[31] S. Fujimoto, D. Meger, and D. Precup. Off-policy deep reinforcement learning without exploration. *arXiv preprint arXiv:1812.02900*, 2018.

[32] Y. Wu, G. Tucker, and O. Nachum. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*, 2019.

[33] A. Kumar, A. Zhou, G. Tucker, and S. Levine. Conservative q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*, 2020.

[34] B. Ibarz, J. Leike, T. Pohlen, G. Irving, S. Legg, and D. Amodei. Reward learning from human preferences and demonstrations in atari. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 8022–8034, 2018.

[35] D. Abel, J. Salvatier, A. Stuhlmüller, and O. Evans. Agent-agnostic human-in-the-loop reinforcement learning. *arXiv preprint arXiv:1701.04079*, 2017.

[36] 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*, pages 1861–1870, 2018.

[37] L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International Conference on Machine Learning*, pages 1407–1416. PMLR, 2018.

[38] X. B. Peng, A. Kumar, G. Zhang, and S. Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

[39] S. Ha, P. Xu, Z. Tan, S. Levine, and J. Tan. Learning to walk in the real world with minimal human effort, 2020.

[40] Q. Li, Z. Peng, Q. Zhang, C. Qiu, C. Liu, and B. Zhou. Improving the generalization of end-to-end driving through procedural generation. *arXiv preprint arXiv:2012.13681*, 2020.

[41] A. Ray, J. Achiam, and D. Amodei. Benchmarking safe exploration in deep reinforcement learning. *openai*, 2019.

[42] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *IROS*, pages 5026–5033. IEEE, 2012. ISBN 978-1-4673-1737-5. URL http://dblp.uni-trier.de/db/conf/iros/iros2012.html#TodorovET12.

[43] E. Liang, R. Liaw, R. Nishihara, P. Moritz, R. Fox, K. Goldberg, J. Gonzalez, M. Jordan, and I. Stoica. Rllib: Abstractions for distributed reinforcement learning. In *International Conference on Machine Learning*, pages 3053–3062, 2018.

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