GUARDED POLICY OPTIMIZATION WITH IMPERFECT ONLINE DEMONSTRATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Teacher-Student Framework (TSF) is a reinforcement learning setting where a teacher agent or human expert guards the training of a student agent by intervening and providing online demonstrations. Assuming the teacher policy is optimal, it has the perfect timing and capability to intervene the control of the student agent, providing safety guarantee and exploration guidance. Nevertheless, in many real-world settings it is expensive or even impossible to obtain a well-performing teacher policy. In this work we relax the assumption of a well-performing teacher and develop a new method that can incorporate arbitrary teacher policies with modest or inferior performance. We instantiate an off-policy Reinforcement Learning algorithm, termed **T**eacher-**S**tudent **S**hared **C**ontrol (**TS2C**), which incorporates teacher intervention based on trajectory-based value estimation. Theoretical analysis validates that the proposed TS2C algorithm attains efficient exploration and lower-bound safety guarantee without being affected by the teacher's own performance. Experiments on autonomous driving simulation show that our method can exploit teacher policies at any performance level and maintain a low training cost. Moreover, the student policy excels the imperfect teacher policy in terms of higher accumulated reward in held-out testing environments.

1 INTRODUCTION

In Reinforcement Learning (RL), the Teacher-Student Framework (TSF) (Zimmer et al., 2014; Kelly et al., 2019) incorporates well-performing neural controllers or human experts as teacher policies in the learning process of autonomous agents. The teacher policy either provides safety guarantee (Peng et al., 2021) or facilitates efficient exploration (Torrey & Taylor, 2013). At each step, the teacher chooses to intervene the free exploration of the student according to a specific intervention function. Online data collected from both the teacher and the student policy will be saved into the replay buffer and exploited with off-policy RL algorithms.

The majority of RL methods in TSF assume the availability of a well-performing teacher policy (Spencer et al., 2020; Torrey & Taylor, 2013), thus the behaviors of student will be largely constrained. The teacher intervention is triggered when the student acts differently from the teacher (Peng et al., 2021) or when the teacher finds the current state worth exploring (Chisari et al., 2021). This is similar to imitation learning where the performance of learned agent is significantly affected by the quality of demonstrations (Kumar et al., 2020; Fujimoto et al., 2019). Thus with current TSF methods if the teacher is incapable of providing high-quality demonstrations, the student will be misguided and its final performance will be upper-bounded by the performance of the teacher. However, it is time-consuming or impossible to obtain a well-performing teacher in many real-world applications such as object manipulation with robot arms (Yu et al., 2020a) and autonomous driving (Li et al., 2022). As a result, current TSF methods will behave poorly in these settings.

In reality, the coach of Usain Bolt does not necessarily need to run faster than Usain Bolt. There could be possible learning schemes where a student can outperform the teacher while retaining a lower-bound safety guarantee from the teacher. In this work we develop a new guarded policy optimization method called **T**eacher-**S**tudent **S**hared **C**ontrol (**TS2C**). It follows the setting of a teacher policy and a learning student policy, but relaxes the requirement of the teacher policy's high-quality demonstrations. A new intervention mechanism is designed: Rather than triggering intervention

based on the similarity of actions of teacher and student, the intervention is now determined by a trajectory-based value estimator. The student is allowed to conduct actions that deviate from the teacher's, as long as their expected return is promising. By relaxing the intervention criterion from step-wise action similarity to trajectory-based value, the student has the freedom to act differently when the teacher fails to provide correct demonstration and thus has the potential to outperform sub-optimal teacher. We conduct theoretical analysis and show that in previous TSF methods the return of the shared control policy is upper bounded by the performance of the teacher policy. In contrast, TS2C is not limited by the suboptimal teacher in upper-bound performance, while still retaining a lower-bound performance and safety guarantee.

Extensive experiments show that under the newly proposed method, the teacher-student shared control can learn efficiently and safely under the different levels of teachers while other TSF algorithms are largely bounded by the teacher's performance. Furthermore, the student policies trained under the proposed TS2C substantially outperform baseline methods in terms of higher efficiency and lower cost, supporting our theoretical analyses.

2 Background

2.1 Preliminaries

We consider an infinite-horizon Markov decision process (MDP), defined by the tuple $M = \langle \mathcal{S}, \mathcal{A}, P, R, \gamma, d_0 \rangle$ consisting of a finite state space $\mathcal{S}$, a finite action space $\mathcal{A}$, the state transition probability distribution $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$, the reward function $R : \mathcal{S} \times \mathcal{A} \to [R_{\min}, R_{\max}]$, the discount factor $\gamma \in (0, 1)$ and the initial state distribution $d_0 : \mathcal{S} \to [0, 1]$. Unless otherwise stated, π denotes a stochastic policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$. The state-action value and state value functions of π are defined as $Q^\pi(s, a) = \mathbb{E}_{s_0=s, a_0=a, a_t \sim \pi(\cdot|s_t), s_{t+1} \sim p(\cdot|s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)\right]$ and $V^\pi(s) = \mathbb{E}_{a \sim \pi(\cdot|s)} Q^\pi(s, a)$. The optimal policy is expected to maximize the accumulated return $J(\pi) = \mathbb{E}_{s \sim d_0} V^\pi(s)$.

The Teacher-Student Framework (TSF) models the shared control system as the combination of a teacher policy π_t which is pretrained and fixed, and a student policy π_s to be learned. The actual actions applied to the agent are deduced from a mixed policy of π_t and π_s, where π_t starts generating action when intervention happens. The details of the intervention mechanism are described in Sec. 3.2. The goal of TSF is to improve the training efficiency and safety of π_s with the involvement of π_t. The discrepancy between π_t and π_s on state s, termed as policy discrepancy, is the L_1-norm of output difference: $\|\pi_t(\cdot|s) - \pi_s(\cdot|s)\|_1 = \int_{\mathcal{A}} |\pi_t(a|s) - \pi_s(a|s)| \, \mathrm{d}a$. We define the discounted state distribution under policy π as $d_\pi(s) = (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s; \pi, d_0)$, where $\Pr^\pi(s_t = s; \pi, d_0)$ is the state visitation probability. The state distribution discrepancy is defined as the difference in L_1-norm of discounted state distributions under two policies: $\|d_{\pi_t} - d_{\pi_s}\|_1 = \int_{\mathcal{S}} |d_{\pi_t}(s) - d_{\pi_s}(s)| \, \mathrm{d}s$.

2.2 Related Work

Teacher-student Learning The idea of transferring knowledge from a teacher policy to a student policy has been explored in reinforcement learning (Zimmer et al., 2014). It improves the learning efficiency of the student policy by leveraging pretrained teacher policy, usually done by adding auxiliary loss to encourage the student policy to be close to the teacher policy (Schmitt et al., 2018; Traoré et al., 2019). Though our method follows teacher-student transfer framework, an optimal teacher is not a necessity. Other attempts to relax the need of well-performing teacher models include student-student transfer (Lin et al., 2017; Lai et al., 2020), in which heterogeneous agents exchange knowledge through mutual regularisation (Zhao & Hospedales, 2021). During training, agents are fully controlled by either the student (Traoré et al., 2019; Schmitt et al., 2018) or the teacher policy (Rusu et al., 2016), while our method follows teacher-student shared control where a mixed policy controls the agent.

Learning from Demonstrations Another way to exploit the teacher policy is to collect static demonstration data from it (Widrow, 1964). The learning agent will then regard them as optimal transitions to utilize. If the data is provided without reward signals, agent can learn by imitating the teacher's policy distribution (Ly & Akhloufi, 2020) or trajectory distribution (Ho & Ermon, 2016; Xu et al., 2019). It can also learn a parameterized reward function with inverse reinforcement learn-

ing (Abbeel & Ng, 2004). With additional reward signals, agents can perform Bellman updates pessimistically, as most offline reinforcement learning algorithms do (Levine et al., 2020). The conservative Bellman update can be performed either by restricting the overestimation of Q-function learning (Fujimoto et al., 2019; Kumar et al., 2020) or by involving model-based uncertainty estimation (Yu et al., 2020b; Chen et al., 2021b). In contrast to the offline learning from demonstration, in this work we focus on the online deployment of teacher policies with teacher-student shared control and show its superiority in reducing the state distributional shift.

Reinforcement Learning with Shared Control The shared control in RL enables both the expert and the learning agent to generate online samples in the environment (Ross et al., 2011; Kelly et al., 2019). The switch between two policies are often determined by the expert, either through the manual intervention of human participators (Abel et al., 2017; Chisari et al., 2021) or by referring to the policy distribution of a parameterized expert (Peng et al., 2021). (da Silva et al., 2020) proposes to ask for expert advice when the learner's action has high estimated uncertainty. The framework is empirically justified and only works for agents with discrete action spaces, while we build our algorithm based on theoretical analysis and investigate continuous action space in this paper. To exploit samples collected with shared control, (Ross et al., 2011; Kelly et al., 2019) compute behavior cloning loss on samples where the expert policy is in control and discard those generated by the learner. Other algorithms (Mandlekar et al., 2020; Chisari et al., 2021) assign positive reward labels on expert samples and compute policy gradient loss based on the pseudo reward. All these methods assume the expert policy to be optimal, while our proposed algorithm can make use of a suboptimal expert policy. There are also researches focusing on provable safety guarantee with shared control (Peng et al., 2021; Wagener et al., 2021), while we provide an additional lower-bound guarantee of the accumulated reward for our method.

3 Guarded Policy Optimization with Online Demonstrations

Fig. 1 shows an overview of our proposed method. In addition to conventional single-agent RL setting, we include a teacher policy π_t in the training loop. The term "teacher" only indicates that the role of this policy is to help the student training. No assumption on the optimality of the teacher is needed. The teacher policy is first used to do warmup rollouts and train a value estimator. During the training of the student policy, both π_s and π_t receive current state s from the environment. They propose actions a_s and a_t, and then a value-based intervention function $\mathcal{T}(s)$ determines which action should be taken and applied to the environment. The student policy is then updated with data collected through such shared control.

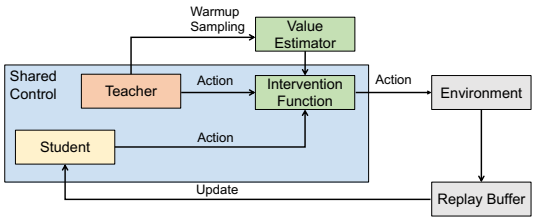

Figure 1: Overview of the proposed teacher-student shared control method. Both student and teacher policies are in the training loop and shared control according to the intervention function.

We first give a theoretical analysis on the general setting of reinforcement learning with shared control in Sec. 3.1. We then discuss the properties of different forms of intervention function $\mathcal{T}$ in Sec. 3.2. Based on these analyses, we propose a new algorithm for teacher-student shared control in Sec. 3.3. All the proofs in this section are included in Appendix A.1.

3.1 Analysis on Shared Control

In reinforcement learning with shared control, the teacher policy and the student policy act together and become a mixed behavior policy π_b. The intervention function $\mathcal{T}(s)$ determines which policy is in charge. Let $\mathcal{T}(s) = 1$ denotes the teacher policy π_t takes control and $\mathcal{T}(s) = 0$ means otherwise. Then π_b can be represented as $\pi_b(\cdot|s) = \mathcal{T}(s)\pi_t(\cdot|s) + (1 - \mathcal{T}(s))\pi_s(\cdot|s)$.

One issue with shared control is that the student policy π_s is trained with samples collected by the behavior policy π_b, whose action distribution is not always aligned with π_s. A large state distribution discrepancy between two policies $\|d_{\pi_b} - d_{\pi_s}\|_1$ can cause distributional shift and ruin the training. A similar problem exists in behavior cloning (BC), though in BC no shared control is involved

and π_s learns from samples all collected by the teacher policy π_t. To analyze the state distribution discrepancy in BC, we utilize a useful lemma from (Achiam et al., 2017).

Lemma 3.1. *The state distribution discrepancy between the teacher policy π_t and the student policy π_s is bounded by their expected policy discrepancy:*

$$\|d_{\pi_t} - d_{\pi_s}\|_1 \leqslant \frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_t}} \|\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 . \tag{1}$$

We apply the lemma to the setting of the shared control and derive a bound for $\|d_{\pi_b} - d_{\pi_s}\|_1$.

Theorem 3.2. *For any behavior policy π_b deduced by a teacher policy π_t, a student policy π_s and an intervention function $\mathcal{T}(s)$, the state distribution discrepancy between π_b and π_s is bounded by*

$$\|d_{\pi_b} - d_{\pi_s}\|_1 \leqslant \frac{\beta\gamma}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_b}} \|\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 , \tag{2}$$

where $\beta = \frac{\mathbb{E}_{s\sim d_{\pi_b}}\left[\mathcal{T}(s)\|\pi_t(\cdot|s)-\pi_s(\cdot|s)\|_1\right]}{\mathbb{E}_{s\sim d_{\pi_b}}\|\pi_t(\cdot|s)-\pi_s(\cdot|s)\|_1} \in [0,1]$ is the expected intervention rate weighed by policy discrepancy.

Both Eq. 1 and Eq. 2 bound the state distribution discrepancy by the difference in per-state policy distributions, but the upper bound with shared control is squeezed by the intervention rate β. In practical algorithms, β can be minimized to reduce the state distribution discrepancy and thus relieve the performance drop during test time. Based on Thm. 3.2, we further prove in Appendix A.1 that under the setting of shared control, the accumulated returns of behavior policy $J(\pi_b)$ and student policy $J(\pi_s)$ can be similarly related. The analysis in this section does not assume a certain form of the intervention function $\mathcal{T}(s)$. Our analysis provides the insight on the feasibility and efficiency of all previous algorithms with expert-in-the-loop shared control, including (Kelly et al., 2019; Peng et al., 2021; Chisari et al., 2021). In the following section, we will examine different forms of intervention functions and investigate their properties and performance bounds, especially with imperfect online demonstrations.

3.2 Learning from Imperfect Demonstrations

A straightforward idea to design the intervention function is to intervene when student acts differently to teacher. This idea resembles imitation learning (IL) methods where student action is optimized based on behavior cloning loss against teacher action at each step. We model such process with the action-based intervention function $\mathcal{T}_{\text{action}}(s)$:

$$\mathcal{T}_{\text{action}}(s) = \begin{cases} 1 & \text{if } \mathbb{E}_{a\sim\pi_t(\cdot|s)}[\log \pi_s(a \mid s)] < \varepsilon, \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

wherein ϵ is a predefined parameter. A similar intervention function is used in (Peng et al., 2021), where the student's action is replaced by the teacher's if it has low probability under the teacher's policy distribution. To measure the effectiveness of a certain form of intervention function, we examine the return of the behavior policy $J(\pi_b)$. With $\mathcal{T}_{\text{action}}(s)$ defined in Eq. 3 we can bound $J(\pi_b)$ with the following theorem.

Theorem 3.3. *With the action-based intervention function $\mathcal{T}_{action}(s)$, the return of the behavior policy $J(\pi_b)$ is lower and upper bounded by*

$$J(\pi_t)+\frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon} \geqslant J(\pi_b) \geqslant J(\pi_t) - \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon}, \tag{4}$$

where $H = \mathbb{E}_{s\sim d_{\pi_b}}\mathcal{H}(\pi_t(\cdot|s))$ is the average entropy of the teacher policy during shared control.

The theorem shows that $J(\pi_b)$ can be lower bounded by the return of the teacher policy π_t and extra term related to the entropy of the teacher policy. It implies that action-based intervention function $\mathcal{T}_{\text{action}}$ is indeed helpful in providing training data with high return.

A drawback of action-based intervention function is the strong assumption on the optimal teacher, which is not always accessible. If we turn to employ a suboptimal teacher, the behavior policy would be burdened due to the upper bound in Eq. 4. We illustrate this phenomenon with the example in Fig. 2. In the figure a slow vehicle in gray is driving in front of the ego-vehicle in blue, which is under shared control. The student policy is aggressive and would like to overtake the gray vehicle to reach the destination faster, while the teacher intends to follow the vehicle conservatively. Therefore, π_s and π_b will propose different actions in the current state, leading to $\mathcal{T}_{\text{action}} = 1$ according to Eq. 3. The mixed policy with shared control will always choose to follow the front vehicle and the agent can never accomplish a successful overtake.

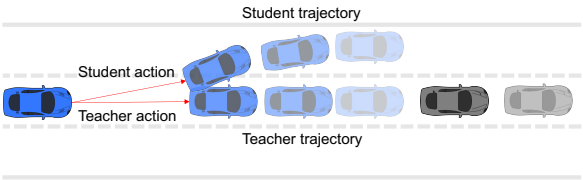

Figure 2: The ego vehicle is the blue one on the left, following the gray vehicle on the right. The upper trajectory is proposed by the student to overtake and the lower trajectory is proposed by the teacher to keep following.

To empower the student to outperform a suboptimal teacher policy, we investigate a new form of intervention function that encapsulates the long-term value estimation into the decision of intervention, designed as follows:

$$\mathcal{T}_{\text{value}}(s) = \begin{cases} 1 & \text{if } V^{\pi_t}(s) - \mathbb{E}_{a\sim\pi_s(\cdot|s)} Q^{\pi_t}(s,a) > \varepsilon, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

By using this intervention function, the teacher allows student's action if the teacher can not perform significantly better than student. $\mathcal{T}_{\text{value}}$ no longer expects the student to imitate the teacher policy step-by-step. Instead, it makes decision on the basis of long-term episodic reward. Taking trajectories in Fig. 2 again as examples, if the overtake behavior has high return, the student policy will be preferable to $\mathcal{T}_{\text{value}}$. Then the student control will not be intervened by the conservative teacher. So with the value-based intervention function, the agent's exploration ability will not be limited by a suboptimal teacher. Nevertheless, the lower-bound performance guarantee of the behavior policy π_b still holds, shown as follows.

Theorem 3.4. *With the value-based intervention function $\mathcal{T}_{value}(s)$ defined in Eq. 5, the return of the behavior policy π_b is lower bounded by*

$$J(\pi_b) \geqslant J(\pi_t) - \frac{(1-\beta)\varepsilon}{1-\gamma}. \tag{6}$$

In safety-critical scenarios, the step-wise training cost $c(s,a)$, *i.e.*, the penalty on the safety violation during training, can be regarded as negative reward. We define $\hat{r}(s,a) = r(s,a) - \eta c(s,a)$ as the combined reward, where η is the weighting hyperparameter. $\hat{V}, \hat{Q}$ and $\hat{\mathcal{T}}_{\text{value}}$ are similarly defined by substituting r with $\hat{r}$ in the original definition. Then we have the following corollary related to expected cumulative training cost, defined by $C(\pi) = \mathbb{E}_{s_0\sim d_0, a_t\sim\pi(\cdot|s_t), s_{t+1}\sim p(\cdot|s_t,a_t)} \left[\sum_{t=0}^{\infty} \gamma^t c(s_t,a_t)\right]$.

Corollary 3.5. *With safety-critical value-based intervention function $\hat{\mathcal{T}}_{value}(s)$, the expected cumulative training cost of the behavior policy π_b is upper bounded by*

$$C(\pi_b) \leqslant C(\pi_t) + \frac{(1-\beta)\epsilon}{\eta(1-\gamma)} + \frac{1}{\eta}\left[J(\pi_b) - J(\pi_t)\right]. \tag{7}$$

In Eq. 7 the upper bound of behavior policy's training cost consists of three terms: the cost of teacher policy, the threshold in intervention ϵ multiplied by coefficients and the superiority of π_b over π_t in cumulative reward. The first two terms are similar to those in Eq. 6 and the third term means a trade-off between training safety and efficient exploration, which can be adjusted by hyperparameter η.

Comparing the lower bound performance guarantee of action-based and value-based intervention function (Eq. 4 and Eq. 6), the performance gap between π_b and π_t can both be bounded with respect to the threshold for intervention ε and the discount factor γ. The difference is that the

performance gap when using $\mathcal{T}_{\text{action}}$ is in an order of $O(\frac{1}{(1-\gamma)^2})$ while the gap with $\mathcal{T}_{\text{value}}$ is in an order of $O(\frac{1}{1-\gamma})$. It implies that in theory value-based intervention leads to better lower-bound performance guarantee. In terms of training safety guarantee, value-based intervention function $\mathcal{T}_{\text{value}}$ has better safety guarantee by providing a tighter safety bound with the order of $O(\frac{1}{1-\gamma})$, in contrast to $O(\frac{1}{(1-\gamma)^2})$ of action-based intervention function (see Theorem 1 in (Peng et al., 2021)). We show in the Sec. 4.3 that the theoretical advances of $\mathcal{T}_{\text{value}}$ in training safety and efficiency can both be verified empirically.

3.3 Implementation

Justified by the aforementioned advantages of the value-based intervention function, we propose a practical algorithm called **T**eacher-**S**tudent **S**hared **C**ontrol (**TS2C**). Its workflow is listed in Appendix B. We first train parametric Q-function with data collected by the teacher policy π_t during the warmup period. Gaussian noise is added to the teacher's policy distribution to increase the state coverage during warmup. With limited training data the Q-function may fail to provide accurate estimation when encountering previously unseen states. We employ the idea of ensembling Q-networks (Chen et al., 2021a). A set of ensembled Q networks $\mathbf{Q}^\phi$ with the same architecture and different initialization weights are built and trained with the same data. To learn $\mathbf{Q}^\phi$ we follow the standard procedure in (Chen et al., 2021a) and optimize the following loss:

$$L(\phi) = \mathbb{E}_{s,a\sim\mathcal{D}}\left[y - \text{Mean}\left[\mathbf{Q}^\phi(s,a)\right]\right]^2, \tag{8}$$

where $y = \mathbb{E}_{s'\sim\mathcal{D},a'\sim\pi_t(\cdot|s')+\mathcal{N}(0,\sigma)}\left[r + \gamma\text{Mean}\left[\mathbf{Q}^\phi(s',a')\right]\right]$ is the Bellman target and $\mathcal{D}$ is the replay buffer storing sequences $\{(s,a,r,s')\}$. Teacher will intervene when $\mathcal{T}_{\text{value}}$ returns 1 or the output variance of ensembled Q-networks surpasses the threshold, which means the agent is exploring unknown regions and requires guidance. We also use $\mathbf{Q}^\phi$ to compute the state-value functions in Eq. 5, leading to the following practical intervention function:

$$\mathcal{T}_{\text{TS2C}}(s) = \begin{cases} 1 & \text{if Mean}\left[\mathbb{E}_{a\sim\pi_t(\cdot|s)}\mathbf{Q}^\phi(s,a) - \mathbb{E}_{a\sim\pi_s(\cdot|s)}\mathbf{Q}^\phi(s,a)\right] > \varepsilon_1 \\ & \text{or Var}\left[\mathbb{E}_{a\sim\pi_s(\cdot|s)}\mathbf{Q}^\phi(s,a)\right] > \varepsilon_2, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

In Eq. 2 it is shown that the distributional shift and the performance gap to oracle can be reduced with smaller β, i.e., less teacher intervention. Therefore, we minimize the amount of teacher intervention via adding negative reward to the transitions one step before the teacher intervention. Incorporating intervention minimization, we use the following loss function to update the student's Q-network parameterized by ψ:

$$L(\psi) = \mathbb{E}_{s,a\sim\mathcal{D}}\left[\left(y' - Q^\psi(s,a)\right)^2\right], \tag{10}$$

where $y' = \mathbb{E}_{s'\sim\mathcal{D},a'\sim\pi_b(\cdot|s')}\left[r - \lambda\mathcal{T}_{\text{TS2C}}(s') + \gamma Q^\psi(s',a') - \alpha\log\pi_b(a'|s')\right]$ is the soft Bellman target with intervention minimization. λ is the hyperparameter controlling the intervention minimization. α is the coefficient for maximum-entropy learning updated in the same way as Soft Actor Critic (SAC) (Haarnoja et al., 2018). To update the student's policy network parameterized by θ, we apply the objective used in SAC as:

$$L(\theta) = \mathbb{E}_{s\sim\mathcal{D}}\left[\mathbb{E}_{a\sim\pi^\theta(\cdot|s)}\left[\alpha\log\left(\pi^\theta(a\mid s)\right) - Q^\psi(s,a)\right]\right]. \tag{11}$$

4 Experiments

We conduct experiments to investigate the following questions: (1) Can agents trained with TS2C achieve super-teacher performance with suboptimal teacher policies while outperforming methods with other intervention functions? (2) Can algorithms with shared control provide safety guarantee and improve training efficiency in complex environments, compared with algorithms without it? (3) Is TS2C robust in teacher policies trained with different algorithms? To answer questions (1)(2), we conduct preliminary training with the PPO algorithm (Schulman et al., 2017) and save checkpoints on different timesteps. Policies in different stages of PPO training are used as teacher policies in TS2C and other algorithms with shared control. With regard to question (3), we use agents trained with PPO (Schulman et al., 2017), SAC (Haarnoja et al., 2018) and Behavior Cloning as the teacher policies from different sources.

Figure 3: Comparison between our method TS2C and other algorithms with shared control. The "Importance" in the legends refers to the Importance Advising algorithm. The involved teacher policies have high, medium and low performance respectively.

4.1 Environment Setup

We conduct the main experiments on the lightweight driving simulator MetaDrive (Li et al., 2022). One concern with teacher-student shared control is that the student may simply record teacher's actions and overfit the training environment. The MetaDrive driving simulator can generate an unlimited number of scenes with various road network and traffic flow for evaluating the generalizability. We choose 100 scenes for training and 50 held-out scenes for testing. Examples of the traffic scenes from MetaDrive are shown in Appendix C. To make a thorough analysis, we also make experiments in the MuJoCo simulator (Todorov et al., 2012). The results are shown in Appendix D.2.

In MetaDrive, the objective is to drive the ego vehicle to the destination without dangerous behaviors such as crashing into other vehicles. The reward function consists of the dense reward proportional to the vehicle speed and the driving distance, and the terminal +20 reward when the ego vehicle reaches the destination. Training cost is increased by 1 when the ego vehicle crashes or drives out of the road. We mainly focus on the average episodic **test reward** evaluated on the held-out test scenes for a fair comparison between different methods. Higher test reward means higher generalizable performance in unseen environment. We also compare the **training cost**. Lower training cost reflects a safer training process which is essential if the algorithm is to be applied in real-world environments.

4.2 Baselines and Implementation Details

Two sets of algorithms are selected as baselines to compare with. One includes traditional RL and IL algorithms without shared control. By comparing with these methods we can demonstrate how RL with shared control improves the efficiency and safety of training. Another set contains previous algorithms with expert and learner shared control, including Importance Advising (Torrey & Taylor, 2013) and EGPO (Peng et al., 2021). The original Importance Advising uses an intervention function based on the range of the Q-function: $I(s) = \max_{a \in A} Q_{\mathcal{D}(s,a)} - \min_{a \in A} Q_{\mathcal{D}(s,a)}$, where $Q_{\mathcal{D}}$ is the Q-table of the teacher policy. Such Q-table is not applicable in the Metadrive simulator with continuous state and action spaces, so several approximations are made. The EGPO algorithm uses an intervention function similar to the action-based intervention function introduced in section 3.2. More details on the experiment setup are in Appendix C.

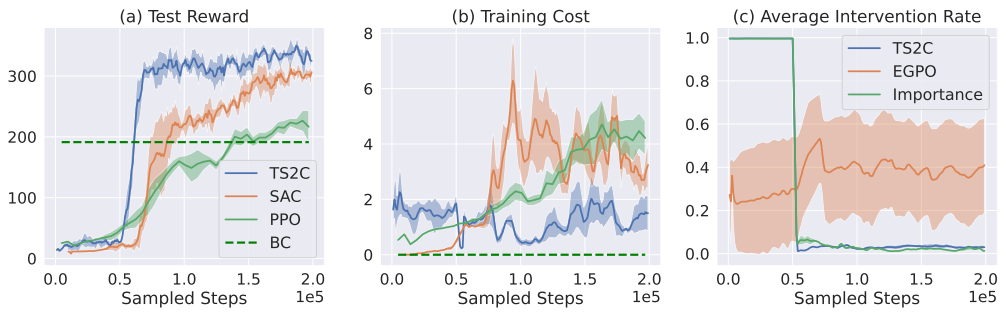

Figure 4: Figures (a) and (b) shows the comparison of efficiency and safety between TS2C and baseline algorithms without shared control. Figure (c) shows the comparison of the average intervention rate between TS2C and two baseline algorithms with shared control.

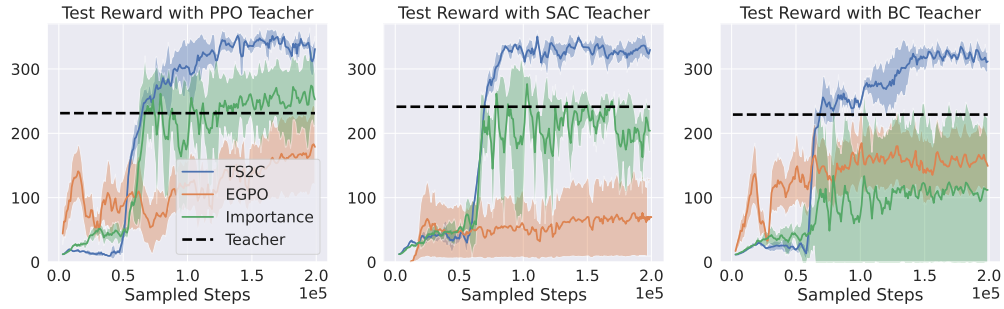

Figure 5: Comparison between our method TS2C and other algorithms with shared control. The teacher policies are trained by PPO, SAC, and behavior cloning respectively.

4.3 Results

Super-teacher performance and better safety guarantee The training result with three different levels of teacher policy can be seen in Fig. 3. The first row shows that the performance of TS2C is not limited by the imperfect teacher policies. It converges within 200k steps, independent of different performances of the teacher. EGPO and Importance Advicing is clearly bounded by teacher-medium and teacher-low, performing much worse than TS2C with imperfect teachers. The second row of Fig. 3 shows TS2C has lower training cost than both algorithms. Compared to EGPO and Importance Advising, the intervention mechanism in TS2C is better-designed and leads to better behaviors.

Better performance with shared control The result of comparing TS2C with baseline methods without shared control can be seen in Fig. 4(a)(b). We use the teacher policy with medium level of performance to train the student in TS2C. It achieves better performance and lower training cost than baseline algorithms SAC, PPO and BC. As an outlier, the behavior cloning algorithm does not involve online sampling in the training process, so it has zero training cost. We also compare TS2C trained from different levels of the teacher policy with the baseline algorithms without shared control and place the results in Appendix D.1.

Exploiting teacher policies from different sources The result of training with different sources of teacher policy can be seen in Fig. 5. The figure shows that TS2C can make use of teacher policies from different sources and achieve super-teacher performance consistently. The Importance Advising algorithm has bounded performance in accordance with the teacher policy. EGPO can only utilize the teacher policy trained from PPO and can not be properly guided by the SAC teacher. This is because the employed SAC teacher is less stochastic than the PPO policy. Student's actions have less likelihood in teacher's action distribution and are less tolerated by the action-based intervention function in EGPO, leading to large intervention rate and consequently large distributional shift. Our proposed TS2C algorithm does not access teacher internal action distribution and instead intervenes based on the state-action values of teacher policy, so it is robust to the stochasticity of teacher policy.

Ablation Studies We also conduct ablation studies and show the results in Fig. 6. We find the intervention cost and ensembled value networks are important to the algorithm's performance, while

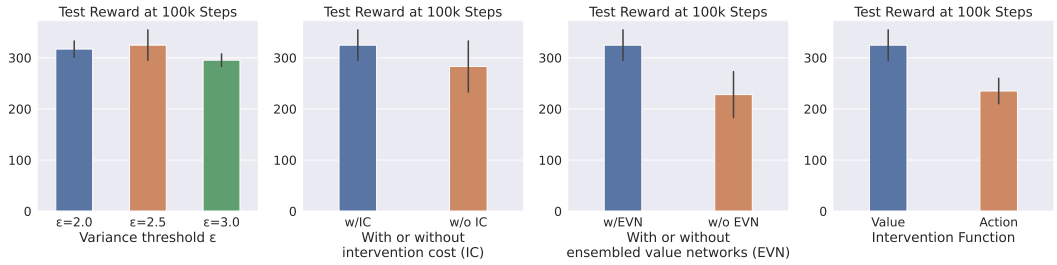

Figure 6: Ablation Studies for different variance thresholds, the intervention cost, ensembled value networks and different intervention functions.

different variance thresholds in the intervention function has little influence. Also, TS2C with action-based intervention function behaves poorly in accordance with the theoretical analysis in Section 3.2.

4.4 Effects of Intervention Functions

We further investigate the intervention behaviors under different intervention functions. As shown in Fig. 4(c), the average intervention rate $\mathbb{E}_{s\sim d_{\pi_b}}\mathcal{T}(s)$ of TS2C drops quickly as soon as the student policy takes control. The teacher policy only intervenes during a very few states where it can propose actions with higher value than the student. The intervention rate of EGPO remains high due to the action-based intervention function: the teacher intervenes whenever the student act differently.

We also show different outcomes of action-based and value-based intervention functions with screenshots in the MetaDrive simulator. In Fig. 7 the ego vehicle happens to drive behind a traffic vehicle which is in an orange trajectory. With action-based intervention the teacher takes control and keeps following the front vehicle, as shown in the green trajectory. In contrast, with the value-based intervention the student policy proposes to turn left and overtake the front vehicle as in the blue trajectory. Such action has higher return and therefore is tolerable by T_{TS2C}, leading to a better agent trajectory.

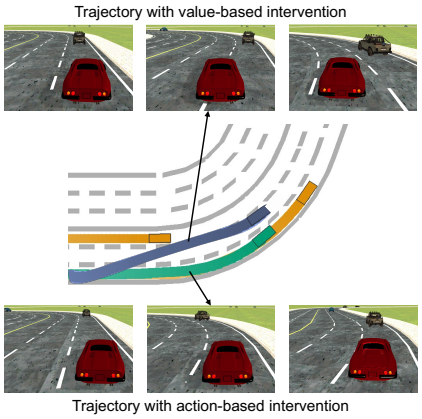

Figure 7: Visualization of trajectories resulting from different intervention mechanisms. The trajectories of traffic vehicles are marked orange. Action-based intervention drives in the lower green trajectory following the front vehicle, while the trajectory with value-based intervention is the upper navy blue one overtaking the front vehicle.

5 Conclusion and Discussion

In this work, we conduct theoretic analysis on the shared control mechanism in the Teacher-Student Framework. It is found that while RL with shared control has better properties than some imitation learning methods, using an action-based intervention function limits the performance of the student policy. We then propose TS2C, a value-based shared control learning scheme for online policy optimization with imperfect teachers. We provide the theoretic guarantees on its exploration ability and safety. Experiments show that the proposed TS2C method achieves consistent performance independent to the teacher policy being used. Our work brings progress and potential impact to relevant topics such as active learning, human-in-the-loop methods, and safety-critical applications.

Limitations. This work assumes the agent can access environmental reward, and defines the intervention function based on value estimations. This limitation could be handled by considering reward-free settings and employing unsupervised skill discovery in the setting of shared control.

REFERENCES

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 1, 2004.

David Abel, John Salvatier, Andreas Stuhlmüller, and Owain Evans. Agent-agnostic human-in-the-loop reinforcement learning. *arXiv preprint arXiv:1701.04079*, 2017.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, pp. 22–31. PMLR, 2017.

Xinyue Chen, Che Wang, Zijian Zhou, and Keith W. Ross. Randomized ensembled double q-learning: Learning fast without a model. In *International Conference on Learning Representations*, 2021a.

Xiong-Hui Chen, Yang Yu, Qingyang Li, Fan-Ming Luo, Zhiwei Qin, Wenjie Shang, and Jieping Ye. Offline model-based adaptable policy learning. *Advances in Neural Information Processing Systems*, 34, 2021b.

Eugenio Chisari, Tim Welschehold, Joschka Boedecker, Wolfram Burgard, and Abhinav Valada. Correct me if i am wrong: Interactive learning for robotic manipulation. *arXiv preprint arXiv:2110.03316*, 2021.

Felipe Leno da Silva, Pablo Hernandez-Leal, Bilal Kartal, and Matthew Taylor. Uncertainty-aware action advising for deep reinforcement learning agents. In *AAAI*, 2020.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pp. 2052–2062. PMLR, 2019.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. PMLR, 2018.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *NIPS*, pp. 4565–4573, 2016.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *NeurIPS*, pp. 12498–12509, 2019.

Michael Kelly, Chelsea Sidrane, Katherine Driggs-Campbell, and Mykel J Kochenderfer. Hg-dagger: Interactive imitation learning with human experts. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8077–8083. IEEE, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*, 2020.

Kwei-Herng Lai, Daochen Zha, Yuening Li, and Xia Hu. Dual policy distillation. In *IJCAI*, 2020.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Quanyi Li, Zhenghao Peng, Lan Feng, Qihang Zhang, Zhenghai Xue, and Bolei Zhou. Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning. *IEEE transactions on pattern analysis and machine intelligence*, 2022.

Eric Liang, Richard Liaw, Robert Nishihara, Philipp Moritz, Roy Fox, Ken Goldberg, Joseph Gonzalez, Michael Jordan, and Ion Stoica. Rllib: Abstractions for distributed reinforcement learning. In *International Conference on Machine Learning*, pp. 3053–3062. PMLR, 2018.

Kaixiang Lin, Shu Wang, and Jiayu Zhou. Collaborative deep reinforcement learning. *arXiv preprint arXiv:1702.05796*, 2017.

Abdoulaye O Ly and Moulay Akhloufi. Learning to drive by imitation: An overview of deep behavior cloning methods. *IEEE Transactions on Intelligent Vehicles*, 6(2):195–209, 2020.

Ajay Mandlekar, Danfei Xu, Roberto Martín-Martín, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Human-in-the-loop imitation learning using remote teleoperation. *arXiv preprint arXiv:2012.06733*, 2020.

Zhenghao Peng, Quanyi Li, Chunxiao Liu, and Bolei Zhou. Safe driving via expert guided policy optimization. In *5th Annual Conference on Robot Learning*, 2021.

Stéphane Ross, Geoffrey Gordon, and Drew 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*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Andrei A Rusu, Sergio Gomez Colmenarejo, Çaglar Gülçehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. In *ICLR (Poster)*, 2016.

Simon Schmitt, Jonathan J Hudson, Augustin Zidek, Simon Osindero, Carl Doersch, Wojciech M Czarnecki, Joel Z Leibo, Heinrich Kuttler, Andrew Zisserman, Karen Simonyan, et al. Kickstarting deep reinforcement learning. *arXiv preprint arXiv:1803.03835*, 2018.

John Schulman, Sergey Levine, Pieter Abbeel, Michael I. Jordan, and Philipp Moritz. Trust region policy optimization. In *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 1889–1897. JMLR.org, 2015.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Jonathan Spencer, Sanjiban Choudhury, Matthew Barnes, Matthew Schmittle, Mung Chiang, Peter Ramadge, and Siddhartha Srinivasa. Learning from interventions: Human-robot interaction as both explicit and implicit feedback. In *Robotics: Science and Systems (RSS)*, 2020.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *IROS*, pp. 5026–5033. IEEE, 2012.

Lisa Torrey and Matthew Taylor. Teaching on a budget: Agents advising agents in reinforcement learning. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems*, AAMAS '13, pp. 1053–1060. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

René Traoré, Hugo Caselles-Dupré, Timothée Lesort, Te Sun, Guanghang Cai, David Filliat, and Natalia Díaz-Rodríguez. Discorl: Continual reinforcement learning via policy distillation. In *NeurIPS workshop on Deep Reinforcement Learning*, 2019.

Nolan Wagener, Byron Boots, and Ching-An Cheng. Safe reinforcement learning using advantage-based intervention. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10630–10640. PMLR, 2021.

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

Tian Xu, Ziniu Li, and Yang Yu. On value discrepancy of imitation learning. *arXiv preprint arXiv:1911.07027*, 2019.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning*, pp. 1094–1100. PMLR, 2020a.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems*, 33:14129–14142, 2020b.

Chenyang Zhao and Timothy Hospedales. Robust domain randomised reinforcement learning through peer-to-peer distillation. In *Proceedings of The 13th Asian Conference on Machine Learning*, volume 157 of *Proceedings of Machine Learning Research*, pp. 1237–1252. PMLR, 2021.

Matthieu Zimmer, Paolo Viappiani, and Paul Weng. Teacher-student framework: a reinforcement learning approach. In *AAMAS Workshop Autonomous Robots and Multirobot Systems*, 2014.

A THEOREMS IN TS2C

A.1 DETAILED PROOF

We start the proof with the restatement of Lem. 3.1 in Sec. 3.1.

Lemma A.1 (Lemma 4.1 in (Xu et al., 2019))**.**

$$\|d_\pi - d_{\pi'}\|_1 \leqslant \frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d_\pi}\|\pi(\cdot \mid s) - \pi'(\cdot \mid s)\|_1 . \tag{12}$$

Thm 3.2 can be derived by substituting π and π' in Lem A.1 with π_b and π_s.

Theorem A.2 (Restatement of Thm. 3.2)**.** *For any behavior policy π_b deduced by a teacher policy π_t, a student policy π_s and a intervention function $\mathcal{T}(s)$, the state distribution discrepancy between π_b and π_s is bounded by policy discrepancy and intervention rate:*

$$\|d_{\pi_b} - d_{\pi_s}\|_1 \leqslant \frac{\beta\gamma}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_b}}\|\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 , \tag{13}$$

where $\beta = \frac{\mathbb{E}_{s\sim d_{\pi_b}}\|\mathcal{T}(s)[\pi_t(\cdot|s)-\pi_s(\cdot|s)]\|_1}{\mathbb{E}_{s\sim d_{\pi_b}}\|\pi_t(\cdot|s)-\pi_s(\cdot|s)\|_1}$ is the weighted expected intervention rate.

Proof.

$$\begin{aligned}
\|d_{\pi_b} - d_{\pi_s}\|_1 &\leqslant \frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_b}}\|\pi_b(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 \\
&= \frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_b}}\|\mathcal{T}(s)\pi_t(\cdot \mid s) + (1-\mathcal{T}(s))\pi_s(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 \\
&= \frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_b}}\|\mathcal{T}(s)\left[\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\right]\|_1 \\
&= \frac{\beta\gamma}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_b}}\|\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 .
\end{aligned} \tag{14}$$

□

Based on Thm. 3.2, we further prove that under the setting of shared control, the performance gap of π_s to the optimal policy π^* can be bounded by the gap between the teacher policy π_t and π^*, together with the teacher-student policy difference. Therefore, training with the trajectory collected with mixed policy π_b is to optimize an upper bound of the student's suboptimality. The following lemma is helpful in doing this.

Lemma A.3.

$$|J(\pi) - J(\pi')| \leqslant \frac{R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_\pi}\|\pi(\cdot \mid s) - \pi'(\cdot \mid s)\|_1 \tag{15}$$

Proof. It is a direct combination of Lemma 4.2 and Lemma 4.3 in (Xu et al., 2019). □

Theorem A.4. *For any behavior policy π_b consisting of a teacher policy π_t, a student policy π_s and a intervention function $\mathcal{T}(s)$, the suboptimality of the student policy is bounded by*

$$|J(\pi^*) - J(\pi_s)| \leqslant \frac{\beta R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim\pi_b}\|\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 + |J(\pi^*) - J(\pi_b)| , \tag{16}$$

Proof.

$$
\begin{aligned}
|J(\pi_b) - J(\pi_s)| &\leqslant \frac{R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \|\pi_b(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 \\
&= \frac{R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \|\mathcal{T}(s)\pi_t(\cdot \mid s) + (1-\mathcal{T}(s))\pi_s(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 \\
&= \frac{R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \|\mathcal{T}(s)\left[\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\right]\|_1 \\
&= \frac{\beta R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim \pi_b} \|\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 . \\
|J(\pi^*) - J(\pi_s)| &\leqslant |J(\pi_b) - J(\pi_s)| + |J(\pi^*) - J(\pi_b)| \\
&\leqslant \frac{\beta R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim \pi_b} \|\pi_t(\cdot \mid s) - \pi_s(\cdot \mid s)\|_1 + |J(\pi^*) - J(\pi_b)| .
\end{aligned}
\tag{17}
$$

□

Theorem A.5 (Restatement of Thm. 3.3)**.** *With the action distributional intervention function $\mathcal{T}_{action}(s)$, the return of the behavior policy $J(\pi_b)$ is lower and upper bounded by*

$$
J(\pi_t) + \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon} \geqslant J(\pi_b) \geqslant J(\pi_t) - \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon} \tag{18}
$$

where $R_{\max} = \max_{s,a} r(s,a)$ is the maximal possible reward, $H = \mathbb{E}_{s\sim d_{\pi_b}}\mathcal{H}(\pi^t(\cdot|s))$ is the average entropy of the teacher policy during shared control.

Proof.

$$
\begin{aligned}
|J(\pi_b) - J(\pi_t)| &\leqslant \frac{R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \|\pi_b(\cdot \mid s) - \pi_t(\cdot \mid s)\|_1 \\
&= \frac{R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \|\mathcal{T}(s)\pi_t(\cdot \mid s) + (1-\mathcal{T}(s))\pi_s(\cdot \mid s) - \pi_t(\cdot \mid s)\|_1 \\
&= \frac{(1-\beta)R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \|\pi_s(\cdot \mid s) - \pi_t(\cdot \mid s)\|_1 \\
&\leqslant \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \sqrt{\mathrm{D_{KL}}(\pi_t(\cdot|s)\|\pi_s(\cdot|s))} \\
&= \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \sqrt{\mathbb{E}_{a\sim\pi_t(\cdot|s)}\left[\log \pi_t(a|s) - \log \pi_s(a|s)\right]} \\
&= \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\mathbb{E}_{s\sim d_{\pi_b}} \sqrt{\mathcal{H}(\pi^t(\cdot|s) - \varepsilon} \\
&\leqslant \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon}.
\end{aligned}
\tag{19}
$$

Therefore, we obtain

$$
\begin{aligned}
\frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon} \geqslant J(\pi_b) - J(\pi_t) \geqslant -\frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon} \\
J(\pi_t) + \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon} \geqslant J(\pi_b) \geqslant J(\pi_t) - \frac{\sqrt{2}(1-\beta)R_{\max}}{(1-\gamma)^2}\sqrt{H-\varepsilon},
\end{aligned}
\tag{20}
$$

which concludes the proof. □

To prove Thm. 3.4, we introduce a useful lemma from (Schulman et al., 2015).

Lemma A.6.

$$
J(\pi) = J(\pi') + \mathbb{E}_{s_t,a_t\sim\tau_\pi}\left[\sum_{t=0}^{\infty}\gamma^t A_{\pi'}(s_t,a_t)\right] \tag{21}
$$

Theorem A.7 (Restatement of Thm. 3.4)**.** *With the value-based intervention function $\mathcal{T}_{value}(s)$ defined in Eq. 5, the return of the behavior policy π_b is lower bounded by*

$$J(\pi_b) \geqslant J(\pi_t) - \frac{\varepsilon}{1-\gamma}. \tag{22}$$

Proof.

$$\begin{aligned}
J(\pi_b) - J(\pi_t) &= \mathbb{E}_{s_n, a_n \sim \tau_{\pi_b}} \left[\sum_{n=0}^{\infty} \gamma^n A_t(s_n, a_n) \right] \\
&= \mathbb{E}_{s_n, a_n \sim \tau_{\pi_b}} \left[\sum_{n=0}^{\infty} \gamma^n \left[Q_t(s_n, a_n) - V_t(s_n) \right] \right] \\
&= \mathbb{E}_{s_n \sim \tau_{\pi_b}} \left[\sum_{n=0}^{\infty} \gamma^n \left[\mathbb{E}_{a \sim \pi_b(\cdot|s_n)} Q_t(s_n, a) - V_t(s_n) \right] \right] \\
&= \mathbb{E}_{s_n \sim \tau_{\pi_b}} \left[\sum_{n=0}^{\infty} \gamma^n \left[\mathcal{T}(s_n) \mathbb{E}_{a \sim \pi_t(\cdot|s_n)} Q_t(s_n, a) + (1 - \mathcal{T}(s_n)) \mathbb{E}_{a \sim \pi_s(\cdot|s_n)} Q_t(s_n, a) - V_t(s_n) \right] \right] \\
&= \mathbb{E}_{s_n \sim \tau_{\pi_b}} \left[\sum_{n=0}^{\infty} \gamma^n \left[(1 - \mathcal{T}(s_n)) \left[\mathbb{E}_{a \sim \pi_s(\cdot|s_n)} Q_t(s_n, a) - V_t(s_n) \right] \right] \right] \\
&= (1-\beta) \mathbb{E}_{s_n \sim \tau_{\pi_b}} \left[\sum_{n=0}^{\infty} \gamma^n \left[\mathbb{E}_{a \sim \pi_s(\cdot|s_n)} Q_t(s_n, a) - V_t(s_n) \right] \right] \\
&\geqslant -(1-\beta) \mathbb{E}_{s_n \sim \tau_{\pi_b}} \left[\sum_{n=0}^{\infty} \gamma^n \varepsilon \right] \\
&= -\frac{(1-\beta)\varepsilon}{1-\gamma},
\end{aligned} \tag{23}$$

which concludes the proof. □

Then we prove the corollary related to safety-critical scenarios.

Corollary A.8 (Restatement of Cor. 3.5)**.** *With safety-critical value-based intervention function $\hat{\mathcal{T}}_{value}(s)$, the expected cumulative training cost of the behavior policy π_b is upper bounded by*

$$C(\pi_b) \leqslant C(\pi_t) + \frac{(1-\beta)\epsilon}{\eta(1-\gamma)} + \frac{1}{\eta} \left[J(\pi_b) - J(\pi_t) \right]. \tag{24}$$

Proof. We define expected return under policy π with combined reward $\hat{r}$ as $\hat{J}(\pi)$, therefore

$$\begin{aligned}
\hat{J}(\pi) &= \mathbb{E}_{s_0 \sim d_0, a_t \sim \pi(\cdot|s_t), s_{t+1} \sim p(\cdot|s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t \hat{r}(s_t, a_t) \right] \\
&= \mathbb{E}_{s_0 \sim d_0, a_t \sim \pi(\cdot|s_t), s_{t+1} \sim p(\cdot|s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t \left[r(s_t, a_t) + \eta c(s_t, a_t) \right] \right] \\
&= J(\pi) + \eta C(\pi)
\end{aligned} \tag{25}$$

According to Thm. A.7, under $\hat{T}_{\text{value}}(s)$ we have

$$\hat{J}(\pi_b) \geqslant \hat{J}(\pi_t) - \frac{\varepsilon}{1-\gamma}. \tag{26}$$

Eq. 24 can be immediately proved by combining Eq. 25 and Eq. 26. □

Algorithm 1 The workflow of TS2C during training

Input: Warmup steps W; Scale of warmup noise σ; Training steps N; Teacher policy π_t.
Initialize student policy π_s^θ, a set of parameterized Q-function for teacher policy $\mathbf{Q}^\phi$, parameterized Q-function for student policy Q^ψ and the replay buffer D.
for $i = 1$ **to** W **do**
  Observe state s_i and sample $a_i \sim \pi_t(\cdot|s) + \mathcal{N}(0, \sigma)$.
  Step the environment with a_i and store the tuple (s_i, a_i, r_i, s_{i+1}) to D.
  Update ϕ with Temporal-Difference loss in Eq. 8.
end for
for $i = 1$ **to** N **do**
  Observe state s_i and sample $a_t \sim \pi_t(\cdot|s_i)$, $a_s \sim \pi_s^\theta(\cdot|s_i)$.
  Compute $\mathcal{T}_{\text{ts2c}}(s_i)$ with Eq. 9, behavior policy $\pi_b(\cdot|s_i)$ and a_b.
  Step the environment with a_b and store the tuple $(s_i, a_b, r_i, s_{i+1}, \mathcal{T}_{\text{value}}(s_{i+1}))$ to D.
  Update ψ in the student Q-function with the loss in Eq. 10.
  Update θ in the student policy with the loss in Eq. 11.
end for

A.2 Discussions on the Results

Thm. 3.3 has a squared planning horizon $\frac{1}{(1-\gamma)^2}$ in the discrepancy term. This is in accordance with many previous works (Thm. 1 in (Xu et al., 2019), Thm. 4.1 in (Janner et al., 2019) and Thm. 1 in (Schulman et al., 2015)), which include $(1-\gamma)^2$ in the denominator when it comes to differences of the cumulative return, given the difference in the action distribution. The order of $\frac{1}{1-\gamma}$ in Thm. 3.3 is tight, which dominates the gap in accumulated return. Nevertheless, the other constant terms, e.g. $R_{\max}$ and the average entropy, can be tighter given some additional assumptions. We did not derive a tighter bound since the derivation will not be related to the main contribution of this paper, which is the new type of intervention function. Thm. 3.3 and Thm. 3.4 in their current forms are enough to demonstrate that the value-based intervention function has the advantage of providing more efficient exploration and better safety guarantee compared with action-based intervention function.

B The Algorithm

The workflow of TS2C during training is show in Alg. 1.

C Additional Experiment Demonstrations

C.1 Demonstrations of Driving Scenarios

The demonstrations of several driving scenarios are shown in Fig. 8. We provide a demonstrative video showing the agent behavior trained with PPO and our TS2C algorithm in the supplementary materials.

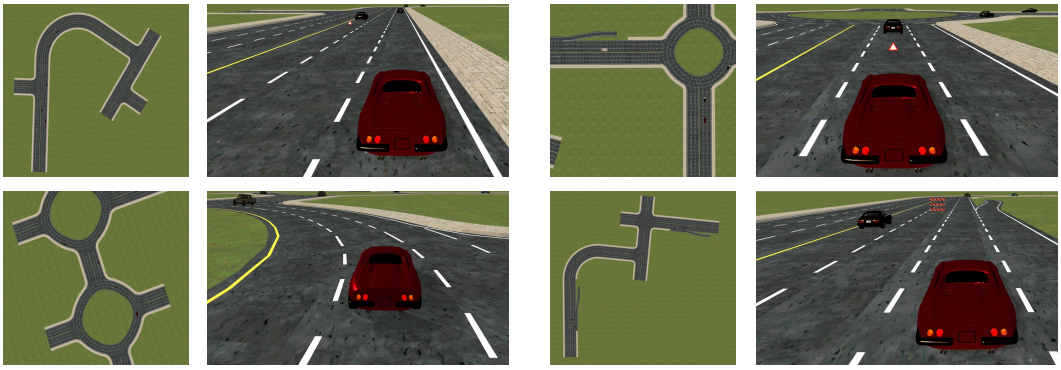

Figure 8: Four examples of the traffic scenes in MetaDrive.

C.2 IMPLEMENTATION DETAILS

The MetaDrive simulator (Li et al., 2022) is released under Apache License 2.0. The experiments are conducted using the RLLib (Liang et al., 2018) and the open-source code of EGPO (Peng et al., 2021). All algorithms are trained with 200,000 timesteps and 4 different random seeds. The training time of a single trial is around 6 hours with two CPUs and one GeForce RTX 2080Ti GPU. With regard to the Importance Advising algorithm, we follow the warmup procedure in TS2C and obtain the teacher Q-value function. On a certain state, we sample N actions from the teacher's policy distribution and compute their Q-values. The intervention happens if the range, i.e., the maximum value minus the minimum value, surpass a certain threshold ε. The choice of hyperparameters N and ε are shown in the next subsection.

C.3 HYPER-PARAMETERS

The hyper-parameters used in the experiments are shown in the following tables. In the TS2C algorithm, larger values of the intervention threshold ε_1 and ε_2 will lead to a more strict intervention criterion and steps with teacher control will be fewer. In order to control the policy distribution discrepancy, we choose ε_1 and ε_2 to ensure the average intervention rate to be less than 5%. Nevertheless, different ε_1 in the intervention function has little influence on the algorithm performance, as shown in Fig. 6 of our paper. The coefficient for intervention minimization λ is simply set to 1. If used in other environments, it may need some adjustments to fit the reward scale. The coefficient for maximum entropy learning α is updated during training as in the SAC algorithm. The number of warmup timesteps is empirically chosen so that the expert value function can be properly trained. Other parameters follow the setting in EGPO (Peng et al., 2021). The hyper-parameters of other algorithms follow their original setting.

Table 1: TS2C (Ours)

Hyper-parameter	Value
Discount Factor γ	0.99
τ for target network update	0.005
Learning Rate	0.0001
Environmental horizon T	2000
Warmup Timesteps W	50000
# of Ensembled Value-Functions N	10
Variance of Gaussian Noise C	0.5
Intervention Minimization Ratio λ	1
Value-based Intervention Threshold ε_1	1.2
Value-based Intervention Threshold ε_2	2.5
Activation Function	Relu
Hidden Layer Sizes	[256, 256]

Table 2: EGPO (Peng et al., 2021)

Hyper-parameter	Value
Discount Factor γ	0.99
τ for target network update	0.005
Learning Rate	0.0001
Environmental horizon T	2000
Steps before Learning start	10000
Intervention Occurrence Limit C	20
Number of Online Evaluation Episode	5
K_p	5
K_i	0.01
K_d	0.1
CQL Loss Temperature β	3.0
Activation Function	Relu
Hidden Layer Sizes	[256, 256]

Table 3: Importance Advising (Torrey & Taylor, 2013)

Hyper-parameter	Value
Discount Factor γ	0.99
τ for target network update	0.005
Learning Rate	0.0001
Environmental horizon T	2000
Warmup Timesteps W	50000
# of Actions Sampled N	10
Variance of Gaussian Noise C	0.5
Range-based Intervention Threshold ε	2.8
Activation Function	Relu
Hidden Layer Sizes	[256, 256]

Table 4: SAC (Haarnoja et al., 2018)

Hyper-parameter	Value
Discount Factor γ	0.99
τ for Target Network Update	0.005
Learning Rate	0.0001
Environmental Horizon T	2000
Steps before Learning starts	10000
Activation Function	Relu
Hidden Layer Sizes	[256, 256]

D Additional Experiment Results

We include additional experiment results here.

D.1 Additional Results on MetaDrive

In Fig. 9, we show the results of TS2C trained with various levels of teachers compared with baseline algorithms without shared control. The value-based intervention proposed by TS2C can utilize all these teacher policies, leading to safer and more efficient training compared to traditional RL algorithms.

Fig. 10 shows the results with different levels of teacher policy. We show the training cost, training reward and test success rate of TS2C compared with baseline methods with shared control respectively.

Figure 9: Comparison of training cost and test reward between our method TS2C and other algorithms without shared control.

Figure 10: Comparison of training reward and test success rate between our method TS2C and other algorithms with shared control.

D.2 Additional Results on MuJoCo

We conduct an additional experiment in the MuJoCo simulator. The original code is adapted to fit the Hopper environment. We train the PPO algorithm and use its checkpoint with average return 1531 as the teacher policy. When training the TS2C algorithm in the Hopper environment, we follow the warmup-training procedure and keep the hyperparameters unchanged. The comparative results are shown in Fig. 11.

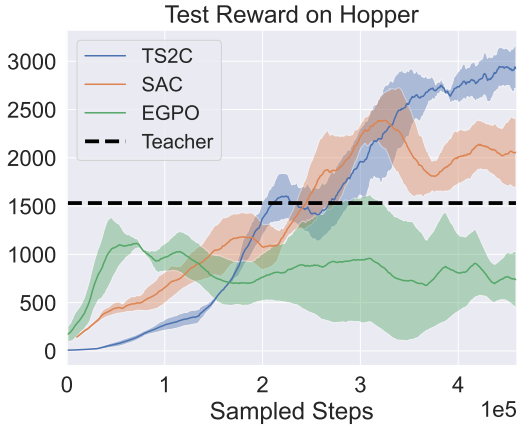

Figure 11: Comparison of test reward between the TS2C, SAC and the EGPO algorithm. Our TS2C algorithm achieves highest performance when it converges.

As shown in the results, the TS2C algorithm can still achieve best performance in the Hopper environment among baseline algorithms. The SAC algorithm has an unstable performance curve and inferior performance overall. The EGPO algorithm shows a constrained performance and can not outperform the imperfect teacher.