Feasibility Consistent Representation Learning for Safe Reinforcement Learning

Zhepeng Cen [1] Yihang Yao [1] Zuxin Liu [1] Ding Zhao [1]

Abstract

In the field of safe reinforcement learning (RL), finding a balance between satisfying safety constraints and optimizing reward performance presents a significant challenge. A key obstacle in this endeavor is the estimation of safety constraints, which is typically more difficult than estimating a reward metric due to the sparse nature of the constraint signals. To address this issue, we introduce a novel framework named Feasibility Consistent Safe Reinforcement Learning (FCSRL). This framework combines representation learning with feasibility-oriented objectives to identify and extract safety-related information from the raw state for safe RL. Leveraging self-supervised learning techniques and a more learnable safety metric, our approach enhances the policy learning and constraint estimation. Empirical evaluations across a range of vector-state and image-based tasks demonstrate that our method is capable of learning a better safety-aware embedding and achieving superior performance than previous representation learning baselines. The project website is available at https://sites.google.com/view/FCSRL.

1. Introduction

Reinforcement Learning (RL) has achieved remarkable success in various domains, leveraging its capability to learn optimal policies by interacting with the environment. This success has spanned from mastering complex games (Mnih et al., 2015; Silver et al., 2016; Hessel et al., 2018) to enabling autonomous systems (Kiran et al., 2021; Li et al., 2022; Ding et al., 2023). However, as RL applications venture into more critical areas such as healthcare, finance, and self-driving vehicles, ensuring safety alongside task performance becomes equally or even more imperative (Garcıa & Fernández, 2015). Safe RL aims to learn a constraint satisfaction policy either by interacting with the environment or from static offline datasets to reduce the risk of policy deployment in safety-critical scenarios (Brunke et al., 2021).

While many strategies have been explored for safe RL problem, from model-based approaches that predict and mitigate potential risks (Kaiser et al., 2019; As et al., 2022), to constrained optimization-based methods (Zhang et al., 2020; Chow et al., 2018; Yang et al., 2020) that ensure policy updates within the feasible set, there still remain several significant hurdles. Among them, one main challenge is cost estimation (Achiam et al., 2017; Tessler et al., 2018), which stems from two main sources: (1) the complexity of the raw state often complicates the prediction of single-step costs; and (2) the sparsity of cost signal intensifies the non-smoothness of ground-truth value function and causes larger noise in value estimation, which is similar to the cases in sparse-reward and goal-conditioned RL (Andrychowicz et al., 2017; Riedmiller et al., 2018; Hare, 2019; Liu et al., 2022a; Eysenbach et al., 2022; Cen et al., 2024). The former influences the evaluation of immediate state safety while the second impacts the estimation of long-term safety. These issues cause a significant over-estimation or under-estimation of agent on costs. Therefore, the safe RL agent struggles to balance the objectives of reward maximization and constraint satisfaction, thus obtaining a suboptimal policy performance.

We propose Feasibility Consistent Safe Reinforcement Learning (FCSRL) to employ representation learning (Bengio et al., 2013) to tackle above challenge. Inspired by the enhancement of dynamics-based representation learning on RL (Yang & Nachum, 2021; Fujimoto et al., 2023), we also leverage transition dynamics of environment to learn the underlying structure of the transition dynamics by applying self-supervised learning (Chen & He, 2021; He et al., 2020) loss on adjacent states in the sample trajectory. Additionally, due to the sparsity of cost, the representation loss based on traditional metrics (e.g., value function) does not boost the safety-aware embedding learning, although they exhibit notable improvement in standard RL setting (Ye et al., 2021; Hessel et al., 2021; Farquhar et al., 2021). To address this, we introduce a novel learning objective, feasibility score, with smoother nature compared to other cost metrics and

[1] Carnegie Mellon University. Correspondence to: Zhepeng Cen <zcen@andrew.cmu.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

adopt it as an auxiliary task in representation learning to refine the safety context features, providing a more accurate constraint estimation and a better trade-off between reward maximization and safety constraint satisfaction for policy learning.

The contributions of this paper can be summarized as:

- We demonstrate the smoothness of the adopted feasibility metric with both theoretical and empirical analysis.
- We apply feasibility metric in representation learning to tackle the cost estimation challenge and propose a representation learning framework, which is compatible with most existing model-free safe RL algorithms.
- The extensive experiments on vector-state and image-based tasks demonstrate that the proposed method learns a better safety-aware embedding and exhibits remarkable and consistent advantages over previous representation learning methods, especially in the case of more stringent constraint.

2. Related Work

Safe reinforcement learning. Safe RL aims to learn an optimal policy by maximizing the reward performance while satisfying the safety constraint (Garcıa & Fernández, 2015; Achiam et al., 2017; Wachi & Sui, 2020; Gu et al., 2022; Xu et al., 2022; Liu et al., 2022c) and another line of research is on safe exploration to improve the safety during training (Sui et al., 2015; Dalal et al., 2018; Wachi et al., 2018; Sootla et al., 2022; Wang et al., 2023). One common solution to the constrained optimization is primal-dual framework (Ray et al., 2019; Ding et al., 2020) and solve an unconstrained optimization with a Lagrangian multiplier (Chow et al., 2018). Gradient-based update methods (Tessler et al., 2018; Zhang et al., 2020) tunes the Lagrangian multiplier to maximize reward while satisfying constraint. Furthermore, Stooke et al. (2020) propose PID-based Lagrangian update to reduce the instability of cost; Liu et al. (2022b) and Huang et al. (2022) apply variational inference to solve optimal multiplier directly, which exhibits better stability and performance during training (Yao et al., 2023). Recent works incorporate model-based RL to improve data efficiency and final performance (Berkenkamp et al., 2017; As et al., 2022; Huang et al., 2023), but typically they require a much larger model to parameterize the environment dynamics.

Representation learning in RL. In context of RL, representation learning typically refers to learning an abstraction or latent embedding or extracting a features of state or action space (Lesort et al., 2018; Abel et al., 2018; Lesort et al., 2018), which is also applied in offline learning setting (Yang & Nachum, 2021; Lin et al., 2024). When state is high-dimensional (e.g., image-based input), it also involves compression from large raw state to a smaller latent vector (Finn et al., 2016; Liu et al., 2021). The advance in self-supervised learning (Chen et al., 2020; He et al., 2020; Grill et al., 2020; Chen & He, 2021) also inspires the representation learning to employ self-supervised loss on augmented states sourced from the same state (Laskin et al., 2020; Kostrikov et al., 2020) or two temporally adjacent states to capture the embedding from dynamics structure (Schwarzer et al., 2020; Yang & Nachum, 2021; Fujimoto et al., 2023). The idea of latent state representations via dynamics model is also closely related to the model-based RL, which also adopt latent embedding learning to improve the learning of world model (Gelada et al., 2019; Hafner et al., 2019; 2020; Kaiser et al., 2019; Schrittwieser et al., 2020; Ye et al., 2021). The model-based RL requires a more expressive parameterization of dynamics model to provide an accurate imagination for planning. In addition to learning on dynamics or world model, another supervision signal for representation learning is value consistency (Oh et al., 2017; Farahmand et al., 2017). Grimm et al. (2020) propose the value equivalence principle to enforce the value function prediction and Bellman backup by latent representation to align with the real environment model, which can improve the representation learning and value learning in recent works (Schrittwieser et al., 2020; Hessel et al., 2021; Farquhar et al., 2021; Yue et al., 2023).

3. Preliminaries

Safe reinforcement learning. Safe RL can be formulated in the framwork of Constrained Markov Decision Process (CMDP), which is defined by the tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, c, \gamma, \mu_0 \rangle$ (Altman, 1999), where $\mathcal{S}$ represents the state space, $\mathcal{A}$ is the action space, $\mathcal{T}: \mathcal{S}\times\mathcal{A}\times\mathcal{S} \to [0,1]$ is the transition function, $\mu_0 : \mathcal{S} \to [0,1]$ is the initial state distribution, γ is the discount factor, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $c : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the cost to characterize the constraint. Typically, the cost signal serves as an indicator of the safety of the current state, which is much sparser than reward. Therefore, in this paper, we focus on the case of binary cost, i.e., $c(s,a) = 1$ indicates the state-action pair (s,a) is unsafe and $c(s,a) = 0$ means it is safe. The objective of safe RL is to find the optimal policy within the constraints

$$\max_{\pi} J_r(\pi), \quad s.t. \quad J_c(\pi) \le \epsilon \tag{1}$$

where $J_{\boldsymbol{f}}(\pi) = \mathbb{E}_{\rho\sim\pi, s_0\sim\mu_0}[\sum_{t=0}^{\infty} \gamma^t \boldsymbol{f}(s,a)], \boldsymbol{f} \in \{r, c\}$ is the reward or cost return of policy π; ρ denotes the sampled trajectory; ϵ is the pre-defined constraint threshold. RL algorithms commonly adopt V value $V_{\boldsymbol{f}}^{\pi}(s) = \mathbb{E}_{\pi}[\sum_{t=0}^{\infty} \gamma^t \boldsymbol{f}(s,a) | s_0 = s]$ and Q value function $Q_{\boldsymbol{f}}^{\pi}(s,a) = \mathbb{E}_{\pi}[\sum_{t=0}^{\infty} \gamma^t \boldsymbol{f}(s,a) | s_0 = s, a_0 = a]$ during training.

To solve the constrained optimization problem, a common practice for safe RL (Tessler et al., 2018; Ray et al., 2019; Zhang et al., 2020) is to transform it to an unconstrained one by introducing a Lagrangian multiplier λ:

$$\min_{\lambda \geq 0} \max_{\pi} J_r(\pi) - \lambda(J_c(\pi) - \epsilon), \tag{2}$$

which can be solved in a general primal-dual framework.

Representation learning for safe RL. In this paper, the main objective of representation learning is to learn a mapping from raw state s to a latent embedding z to facilitate RL part. Particularly, we aim to extract safety-related features in state space and distinctly manifest it within the embedding space. By utilizing this representation as input, we can learn a better policy with both task utility efficiency and constraint satisfaction performances.

4. Method

In this section, we present **feasibility consistent** representation learning. Fig. 1 show the overview of representation learning pipeline, which includes following two main components.

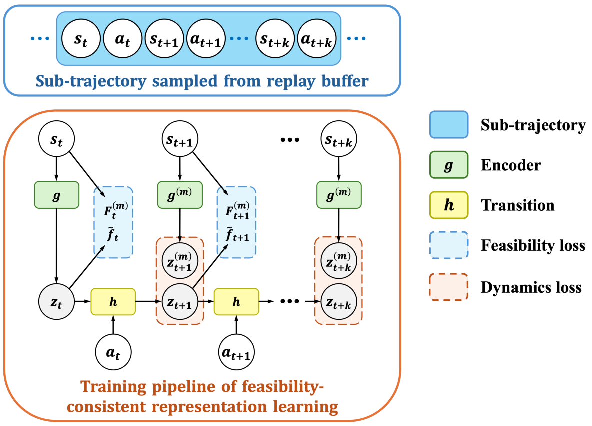


Figure 1: The pipeline of feasibility consistent representation learning. There are two main parts in learning objective: (1) the dynamics loss is between the predicted representation z_t and target $z_t^{(m)}$, and (2) the feasibility consistency loss is between $\tilde{f}_t$ predicted from representation z and target feasibility $F^{(m)}$ estimated by Bellman bootstrap.

4.1. Transition Dynamics Consistency

The representation learning achieves great success in non-sequential tasks and improves the sample efficiency of vision-input RL by mapping a high-dimensional observation to a low-dimensional embedding (Laskin et al., 2020). However, such mapping is not explicitly related to transition dynamics of the environment, which may be less helpful for decision-making. Therefore, based on the intuition that *a good state embedding in RL should be predictive of the future* (Schwarzer et al., 2020; Fujimoto et al., 2023), we adopt dynamics consistency loss in representation learning.

Given a transition pair $\tau = (s_t, a_t, s_{t+1})$ from replay buffer $\mathcal{D}$, we use an encoder g to encode the state s_t to embedding $z_t = g(s_t)$ and a transition model h to output $z_{t+1} = h(z_t, a_t)$ as the prediction of the next-state embedding z_{t+1}. To avoid potential monotonic increasing or representation collapse (Grill et al., 2020), we adopt a non-trainable target encoder (or momentum encoder) $g^{(m)}$ to compute the target next-state embedding $z_{t+1}^{(m)}$, which is updated by exponential moving average (EMA) of encoder g (He et al., 2020). The dynamics consistency loss of the transition is defined as

$$\ell^{\text{dyn}}(\tau) := D^{\text{dyn}}(z_{t+1}, z_{t+1}^{(m)}) \tag{3}$$

where $D^{\text{dyn}}(\cdot,\cdot)$ is a similarity function between two inputs. Similar to prior work (Schwarzer et al., 2020; Ye et al., 2021), we adopt SimSiam-style loss (Chen & He, 2021). See Appendix B.1 for more details.

Note that the dynamics consistency loss can be extended to longer transition sequence sampled from replay buffer: Given $\tau = (s_t, a_t, s_{t+1}, \ldots, s_{t+K})$, we can iteratively get the prediction $z_{t+k+1} = h(z_{t+k}, a_{t+k})$ as shown in fig. 1. The final dynamics loss is $\ell^{\text{dyn}}(\tau) = \sum_{k=1}^{K} D^{\text{dyn}}(z_{t+k}, z_{t+k}^{(m)})$.

Although our method takes the consistency on transition dynamics into consideration, the main target is to capture the state structure relation in a latent embedding space instead of predicting the dynamics precisely as model-based RL (Hafner et al., 2019; 2020), which requires a much larger number of parameters especially for tasks with high-dimensional state space and complex dynamics.

4.2. Feasibility Consistency

In safe RL, the agent performance is not only related to the dynamics estimation and reward critic learning, but also heavily depends on the constraint estimation. Both overestimation or underestimation of constraint violation can reduce the final reward: the formal makes the policy overly conservative while the latter can leads to higher Lagrangian multiplier λ (i.e., a larger cost penalty coefficient) during learning. Therefore, to improve the safety awareness of the representation in safe RL, we propose to add a feasibility consistency loss in learning objective.

We first define the feasibility score as maximum discounted cost:

$$F^{\pi}(s,a) := \mathbb{E}_{\rho\sim\pi}\left[\max_t \gamma^t c(s_t, a_t) | s_0 = s, a_0 = a\right]. \tag{4}$$

The formulation of feasibility score is closely related to Hamilton-Jacobi (HJ) reachability in safe control theory and state-wise safe RL (Bansal et al., 2017; Fisac et al., 2019; Yu et al., 2022). The HJ-reachability-based methods use a much more informative safe signal to solve a safe RL problem with stricter constraint. Specifically, the reachability is computed based on a dense state constraint function (e.g., the distance to hazard) and indicates the level of state-wise safety. However, such dense cost may not be accessible in practical application. On the contrary, our method utilizes feasibility as an representation learning supervision instead of enforcing state-wise constraint satisfaction, which boosts the performance in general sparse-cost setting.

The feasibility score satisfies the Bellman equation $F^\pi = \mathcal{P}_F^\pi F^\pi$, where the corresponding Bellman operator is

$$\begin{aligned}\mathcal{P}_F^\pi F^\pi(s,a) &:= (1-\gamma)c(s,a)\\ &+ \gamma \max\left\{c(s,a), \mathbb{E}_{s'\sim\mathcal{T}(\cdot|s,a),a'\sim\pi(\cdot|s')}F^\pi(s',a')\right\}.\end{aligned} \tag{5}$$

Different from previous (Fisac et al., 2019), we adopt expectation over next action $\mathbb{E}_{a'}$ instead of maximization in Bellman operator, which is also commonly used in off-policy algorithms (Lillicrap et al., 2015; Fujimoto et al., 2018).

The following proposition further shows that the feasibility score is also a safety indicator of future trajectory, which thus can be incorporated into safety-aware representation learning.

Proposition 4.1. *If the cost function c is binary and the discount factor $\gamma \to 1$, then $(1 - F^\pi(s,a))$ is equal to the probability of every following state-action is safe, i.e.,*

$$1 - F^\pi(s,a) = \Pr\left(\bigcap_{(s_t,a_t)\sim\rho}\{c(s_t,a_t)=0\}\right), \tag{6}$$

where ρ is the trajectory starting with (s,a) sampled by policy π.

The proof is in Appendix A.1. Proposition 4.1 shows that the defined feasibility score is a indicator the level of safety.

Therefore, we adopt feasibility score as a supervision signal to extract safety-related information. Specifically, given a sampled sub-trajectory $\tau = (s_t, a_t, r_t, c_t, s_{t+1}, \dots)$, we apply a feasibility prediction head $\tilde{f} : \mathcal{Z} \to \mathbb{R}$ upon the learned embedding and minimize the loss between the predicted score and bootstrap estimate:

$$\ell^{\text{fea}}(\tau) = D^{\text{fea}}\left(\tilde{f}(z_t), F^{(m)}(s_t)\right), \tag{7}$$

where $z_t = g(s_t)$, $F^{(m)}(s_t) = \max\{c_t, \gamma\tilde{f}(z_{t+1}^{(m)})\}$ [1] denotes the bootstrap estimate of feasibility score for state s_t, and $D^{\text{fea}}(\cdot,\cdot)$ denotes the distance between two feasibility scores.

Similarly, for longer sequence, we can further extend the loss as: $\ell^{\text{fea}}(\tau) = \sum_{k=0}^{K-1} D^{\text{fea}}(\tilde{f}(z_{t+k}), F^{(m)}(s_{t+k}))$.

Since we predict the feasibility by a representation z of state s without the action, the true feasibility $F^\pi(s,\cdot)$ is actually a distribution when inputting different actions. When such distribution is widespread, it may cause large estimation error if we use a single expected value to represent. Therefore, we leverage the idea of distributional RL (Bellemare et al., 2017) and apply discrete regression (Schrittwieser et al., 2020; Hafner et al., 2020; Schwarzer et al., 2020) to learn the prediction head. Specifically, we discretize the output space into several buckets and compute the target discrete distribution $d_{F^{(m)}}(s_t)$ by projecting the target feasibility value $F^{(m)}(s_t)$ into buckets. Along with the predicted discrete distribution $d_{\tilde{f}}(z_t)$ by feasibility head $\tilde{f}$, the optimization loss is defined as the KL divergence:

$$D^{\text{fea}}\left(\tilde{f}(z_t), F^{(m)}(s_t)\right) = D_{\text{KL}}\left(d_{F^{(m)}}(s_t) \| d_{\tilde{f}}(z_t)\right). \tag{8}$$

We also adopt the discrete regression in baselines (e.g., to predict cost value V_c or one-step cost) for fair comparison.

4.3. Summary of the Proposed Method

Let θ denote the parameters of $\{g, h, \tilde{f}\}$, the final objective for representation learning is

$$\mathcal{L}_\theta = \mathbb{E}_{\tau\sim\mathcal{D}}\left[\mathcal{L}_\theta^{\text{dyn}}(\tau) + \lambda^{\text{fea}}\mathcal{L}_\theta^{\text{fea}}(\tau)\right]. \tag{9}$$

Since the update of state representation may cause large noise in input space when optimizing RL objective (e.g., both actor and critic loss), we take the target representation $z^{(m)}$ generated by target encoder as their inputs when training the policy and value function of the agent.

By incorporating our representation learning method into an on-policy or off-policy model-free safe RL algorithm, we obtain **F**easibility **C**onsisent **S**afe **RL** (FCSRL). The main procedure of FCSRL is summarized in Algorithm 1.

When using an on-policy safe RL algoithm (e.g., PPO-Lagrangian) as the base RL method, we replace the bootstrapped value and feasibility score by Monte Carlo estimation from sampled on-policy trajectories: $V_c^{(m)}(s_i) = \sum_{t=i}^{T}\gamma^t c_t$, $F^{(m)}(s_i) = \max_{i\le t\le T}\gamma^t c_t$, which aligns the objective of representation learning with the critic learning in on-policy RL.

4.4. Comparison with Value Consistency

In previous work, another objective for state representation learning is value consistency (Schrittwieser et al., 2020; Ye et al., 2021; Hessel et al., 2021; Farquhar et al., 2021),

[1] By definition, the target feasibility score is $\max\{c_t, (1-\gamma)c_t + \gamma\tilde{f}(z_{t+1}^{(m)})\}$, but we can ignore $(1-\gamma)c_t$ when cost function is binary.

Algorithm 1 Feasibility Consistent Safe RL

```
Initialize policy $\pi$, value functions, Lagrangian multiplier;
Initialize encoder and target encoder $g, g^{(m)}$, transition model $h$ and feasibility head $\tilde{f}$, which are parameterized by $\theta_g, \theta_{gm}, \theta_h, \theta_f$ respectively; $\theta_{gm} \leftarrow \theta_g$.
Initialize replay buffer $\mathcal{D}$.
while Training do
    Collect experience $\{(s, a, r, c, s')\}$ by policy $\pi$ and add it to replay $\mathcal{D}$;
    Sample a minibatch of length-$K$ sub-trajectories $B$ from $\mathcal{D}$;
    ▷ state representation learning
    Compute the $\mathcal{L}_\theta$ by Eq.(9) by sub-trajectories $B$.
    Update the $g, h, \tilde{f}$ by loss $\mathcal{L}_\theta$;
    Update $\theta_{gm}$ by EMA of $\theta_g$;
    Compute the target representation $z^{(m)} = g^{(m)}(s)$;
    ▷ safe reinforcement learning
    Update value functions with $z^{(m)}$ as input by $B$;
    Update policy with $z^{(m)}$ as input by $B$;
    if Lagrangian flag then
        Update Lagrangian multiplier.
    end if
end while
```

which enforces the learned state representation to be able to predict the corresponding value function and has proven to improve the reward performances in standard RL setting. However, in safe RL problems, the consistency objective on cost value does not exhibit similar advantages due to sparsity of the cost signals, which results in large in-continuity and roughness (non-smoothness) of the value estimation, and thus increasing the difficulty of prediction.

On the contrary, within our method, the proposed feasibility function is always *smoother* than the original cost value function, which is introduced in the following proposition.

Definition 4.2 (Temporal smoothness)**.** Given a trajectory $\rho = \{s_0, a_1, s_1, \ldots, s_T\}$, with a slight abuse of notation, we denote an arbitrary state-action function $f(s_t, a_t)$ based on ρ as $f(t)$. The temporal smoothness of f along the trajectory ρ is defined as:

$$L(f, \rho) := \mathbb{E}_\rho |f(t) - f(t+1)|, \tag{10}$$

where the expectation is with respect to t along the given trajectory ρ. The smoothness L characterizes the rate of change of the function along the specified trajectory.

Proposition 4.3. *The single trajectory estimation of feasibility score is temporally smoother than cost value function for any trajectory, i.e.,*

$$L(\hat{F}, \rho) \leq L(\hat{V}_c, \rho), \quad \forall \rho, \tag{11}$$

where given $(s_t, a_t) \sim \rho$, the functions $\hat{F}$ and $\hat{V}_c$ is defined as:

$$\begin{cases} \hat{F}(s_t, a_t) = \max_{i \geq t} \gamma^{t-i} c(s_i, a_i) \\ \hat{V}_c(s_t, a_t) = \sum_{i \geq t} \gamma^{t-i} c(s_i, a_i) \end{cases}. \tag{12}$$

The proof is available in Appendix A.2.

Furthermore, to empirically compare the cost value and feasibility score, we visualize the target estimations of them in the same region of PointGoal2 task, which are computed by Bellman bootstrap. The figure 2 shows that the feasibility score is much less noised than the value function and matches better with unsafe region. Therefore, feasibility score is a better safety-awareness signal for representation learning in sparse cost setting.

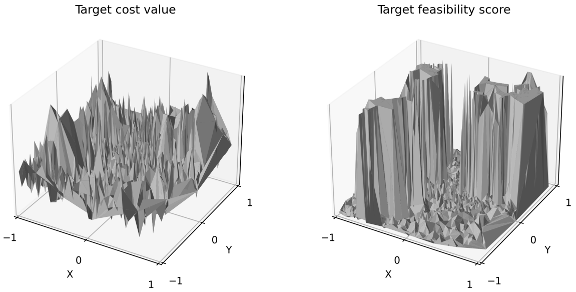


Figure 2: The landscape of target cost value $V_c^{(m)}(s)$ and target feasibility score $F^{(m)}(s)$ obtained by bootstrap estimation in PointGoal2 task. The X-Y axis means the *coordinate of agent* when its state is s. The values of z-axis have been rescaled. See Appendix B.2 for more details.

5. Experiment

In the experiment part, we empirically test the performances of proposed representation method. Particularly, we focus on two main questions: (1) Does FCSRL exhibit a consistent advantage over baselines across a wide range of environments, both in vector-state and image-based settings? (2) Is FCSRL capable of learning a safety-aware state representation to boost the performance of safe RL algorithm?

5.1. Tasks

To answer the above questions, we use 6 vector-state and 3 image-based continuous robotic control tasks as our testbeds adopted from safety-gymnasium (Ji et al., 2023), a widely used evaluation benchmark by previous work for safe RL (Liu et al., 2023).

5.2. Results on Vector-state Tasks

We first compare our method with previous representation learning baselines in low-dimensional state-input tasks.

5.2.1. Baselines

For ease of reading, we omit the expectation over transition or sub-trajectory in the learning objective when introducing the baselines. We adopt following baselines:

Raw state input: Without encoding state into embedding, both policy and value function take the raw state as input.

Forward raw model predicts the next state s_{t+1} based on the embedding z_t and a_t. Specifically, we train an additional state prediction model $f : \mathcal{Z} \times \mathcal{A} \to \Delta(\mathcal{S})$ which outputs a distribution over next state. The learning objective is $\min -\log P(s_{t+1}|f(z_t, a_t))$.

Forward latent model predicts the next state z_{t+1} based on the embedding z_t and a_t. The predictor $f : \mathcal{Z} \times \mathcal{A} \to \Delta(\mathcal{Z})$ outputs the distribution over next state representation. The learning objective is $\min -\log P(z_{t+1}|f(z_t, a_t))$, which can be generalized to longer sequence as $\min -\sum_{k=0}^{K-1} \log P(z_{t+k+1}|f(\hat{z}_{t+k}, a_{t+k}))$, where $\hat{z}_{t+k}$ is the mean output of the last prediction.

Inverse model predicts the action a_t based on the embeddings z_t, z_{t+1}. The predictor $f : \mathcal{Z} \times \mathcal{Z} \to \Delta(\mathcal{A})$ outputs the distribution over action and the learning objective is $\min -\log P(a_t|f(z_t, a_{t+1}))$.

Temporal contrastive learning (TCL) applies a contrastive loss between z_t and z_{t+1}. We use $\exp(z_1^T W z_2)$ as similarity function, where W is a trainable square matrix, and use InfoNCE loss (Oord et al., 2018) as learning objective:

$$\min -f(z_t^T W z_{t+1}) + \log \sum_{\tilde{z} \in Z} \exp(z_t^T W \tilde{z}) \tag{13}$$

where $Z = \{z_{t+1}^{(1)}, z_{t+1}^{(2)}, \cdots, z_{t+1}^{(N)}\}$ is a batch of embeddings randomly sampled from replay buffer. It can also be generalized to long sequence with multiple weight matrices $\{W_1, \dots, W_k\}$. We also use target encoder $g^{(m)}$ (non-trainable and updated by EMA of online encoder g) to obtain the representation of next state z_{t+1}, which empirically improves the reward performance in prior work (Yang & Nachum, 2021).

SALE (Fujimoto et al., 2023) learns an encoder g and transition model h. The learning objective is to minimize the MSE between z_{t+1} and $\hat{z}_{t+1} = h(z_t, a_t)$. SALE also rescales the embedding z by average L1 normalization to prevent the representation collapse. Meanwhile, SALE finds that inputting both state and its representation into policy and value function in RL empirically improves the final performance in vector state tasks.

Value consistent model (VC): The main idea of value consistent model is to enforce the learned embedding to predict cost value function with a prediction head $\tilde{v} : \mathcal{Z} \to \mathbb{R}$, i.e., $\tilde{v}(z_t) = V_c(s_t), z = g(s_t)$. Meanwhile, we observe that the value equivalence (Grimm et al., 2020) may deteriorate the final performance, which learns the critic in an end-to-end manner by 1) directly feeding the z_t to RL critic, and 2) jointly using critic loss and other representation loss to train encoder g. This phenomenon has also been observed by previous work (Farquhar et al., 2021). Therefore, we still learn the representation with an additional value prediction head. The only difference from FCSRL is that the VC models predicts the cost value and we keep all other settings the same, e.g., using target encoder and discrete regression loss.

For fair comparison, we adopt the same network architecture of neural network. Specifically, we parameterize the encoder g, transition model h, and all prediction heads as two-layer MLPs. Following SALE, FCSRL and VC also input the concatenation of state and representation to the RL policy and value function.

5.2.2. Evaluation Results

To compare the performances of our method with above representation learning baselines, we test them with two different base safe RL algorithms: an on-policy method `PPO-Lag`, and an off-policy one `TD3-Lag`, which augment PPO (Schulman et al., 2017) and TD3 (Fujimoto et al., 2018) with Lagrangian method (Ray et al., 2019). We also use PID Lagrangian update (Stooke et al., 2020) to improve the stability of cost performance. We adopt cost limit $\epsilon = 10$ for `TD3-Lag` experiments and $\epsilon = 25$ for `PPO-Lag` because almost all representation learning methods with `PPO-Lag` fail to get a positive reward while satisfying constraint when ϵ is small. We set prediction length $K = 4$ for forward latent, TCL, SALE, VC and FSCRL. We train every method for 2M environment steps. The training curve is attached in Appendix B.6.

In fig. 3, we report the normalized rewards and costs of baselines and our method. The normalized reward is computed by NR $= (R - R_l)/(R_h - R_l)$, where R is the unnormalized reward return and R_l, R_h denote the reward performances of *random policy* and *unconstrained PPO policy* respectively on the given task. The normalized cost is computed by NC $= C/\epsilon$, where C, ϵ are the unnormalized cost return and cost threshold.

As illustrated in fig. 3, the cost performances of most representation learning methods converge to the preset cost limit, which validates that the Lagrangian method is able to balance the reward and cost performances and tunes the value of Lagrangian multiplier λ according to the discrepancy between the cost performance and the constraint threshold.

Overall, the methods based on `TD3-Lag` outperform those based on `PPO-Lag` but the raw-state-input `TD3-Lag` fails to meet the constraint criteria. FCSRL exceeds all baselines on most tasks with either `PPO-Lag` or `TD3-Lag`. In Goal1 tasks, the advantage margin of FCSRL over SALE

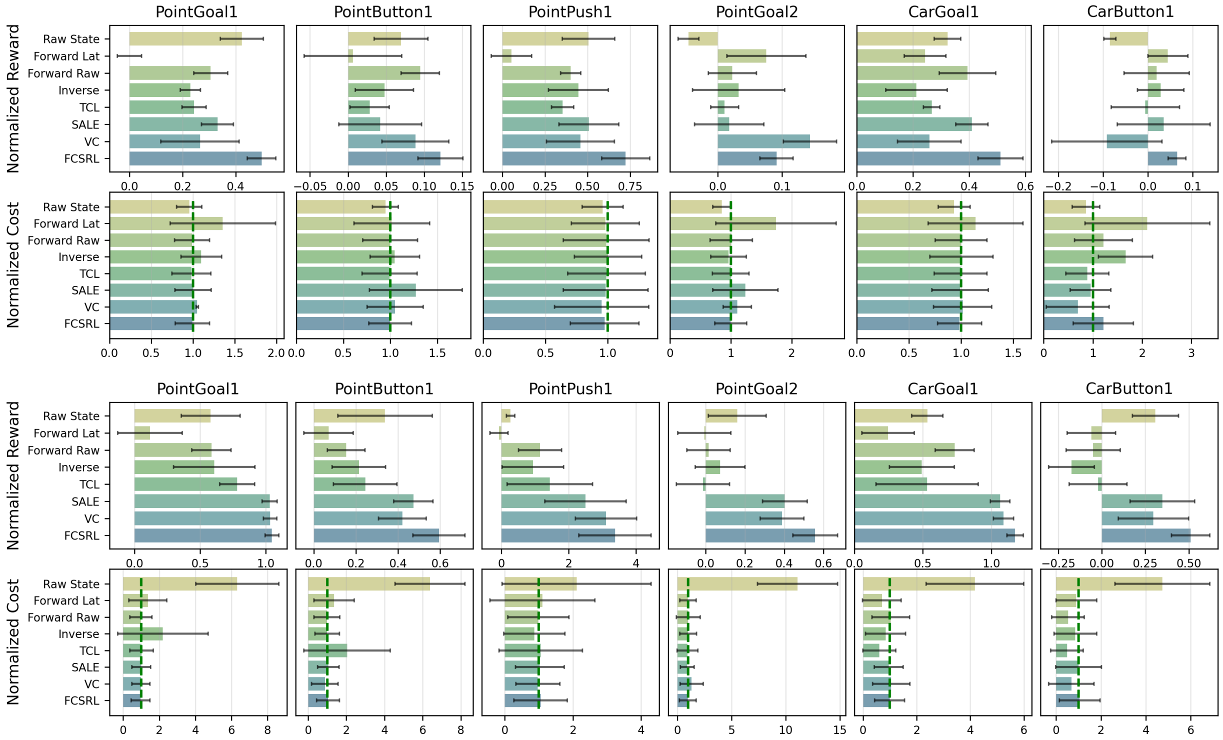


Figure 3: The converged performances of different representation learning methods based on PPO-Lagrangian (**top**) and TD3-Lagrangian (**bottom**). The error bar indicates the standard deviation of 5 seeds. The *green dash line* in normalized cost plots indicates the constraint threshold.

and VC model seems to be relatively small. This may be attributed to their close performances to the theoretical maximum reward and one evidence is that their rewards exceed unconstrained PPO (with normalize reward = 1). Additionally, FCSRL achieves superior performance on Push1 task than unconstrained PPO, which validates the advantages of feasibility consistency in representation learning.

Besides, the SALE, VC, and FCSRL have relatively higher performances than the remaining baseline, suggesting the advantages in feeding both state and representation for low-dimensional state tasks. It may stem from reward-related information loss (e.g., the distance of agent to ”Goal” is closely related to the reward) in representation learning, although the learned embedding captures the safety-related information in state space. We also provide a comparison in Sec.5.5. Meanwhile, it is noteworthy that the forward raw method (which predicts raw state) outperforms forward latent (which only predicts latent embedding) in most tasks, which further highlights the necessity of retaining the information in raw state.

Compared to VC model, the reward margin of FCSRL over value consistent model is larger on tasks where constraint is harder to satisfy (this means the reward is much smaller than the unconstrained tasks, e.g., PointButton1, PointGoal2 and CarButton1), which shows the advantages of using feasibility score to supervise the embedding learning. Furthermore, We provide a detailed comparison of learned embedding z by two methods in Appendix B.3.

5.3. Results on Image-based Tasks

We further consider high-dimensional image-based tasks, where the image-input RL usually encounters the issue of data inefficiency and it requires an encoder to extract a low-dimensional representation from high-dimensional raw observation. Therefore, we compare our method with other vision-based representation learning methods to verify the effectiveness of feasibility consistency.

We adopt following image-based representation learning baseline: 1) **CURL** (Laskin et al., 2020) extracts the embedding by applying contrastive learning on two augmentation given the same original image; 2) **SPR** (Schwarzer et al., 2020) learns the representations by applying contrastive learning on adjacent states in the same trajectory; and 3) **Value consistent model**. We directly input the raw image to encoder without augmentation for SPR, VC and FCSRL. To exclude the influence of network structure, we keep it the same for fair comparison. We use `TD3-Lag` as the base safe RL algorithm, where the policy and value function take the low-dimensional embedding z as input. We report the

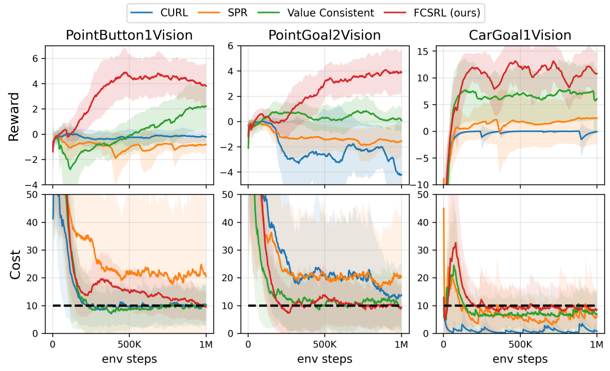


Figure 4: Training curve of image-based tasks. The black dash line is the cost limit. The shadow region is the standard deviation of 5 seeds.

training curve in fig. 4.

The results clearly demonstrate that our method significantly surpasses the baseline models. Specifically, CURL fails to achieve high rewards and exhibits excessive conservatism in the CarGoal1 task, with both reward and cost metrics nearing zero. This limitation arises because CURL only learns an embedding for a single state, focusing solely on vision representation and neglecting the temporal features crucial for sequential decision-making. SPR, on the other hand, does not meet the constraints in the PointButton1 and PointGoal2 tasks. This leads to an increase in the Lagrangian multiplier, consequently diminishing the reward performance during training. In contrast, both the value-consistent model and FCSRL excel in terms of reward and cost, underscoring the benefits of steering representation learning with cost-related signals. Furthermore, the performance of FCSRL indicates that representations with feasibility consistency are more effectively learned, exhibiting a better capability to distill a low-dimensional embedding from high-dimensional image states.

5.4. Performances with Different Cost Limits

In this section, we aim to study how the safety-awareness of learned representation affects the safe RL performance. Based on the intuition that safe RL policy performance increasingly depends on safety-aware representations as cost constraints become more stringent, we evaluate the performance discrepancies among different representation learning methods under cost limit $\epsilon \in \{5, 10, 20, 30, 40\}$.

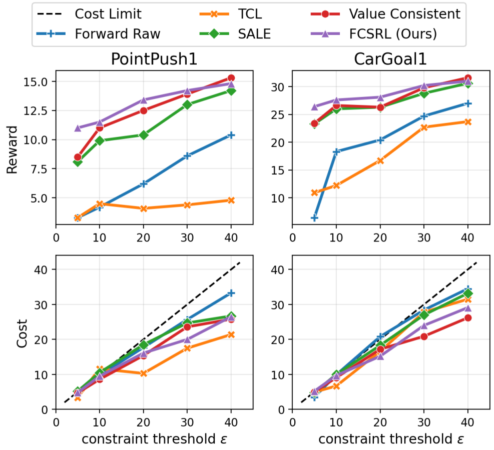


Figure 5: Comparison of reward and cost performances with different constraint thresholds.

We test the performances of forward raw, TCL, SALE, VC, and FCSRL based on `TD3-Lag` on PointPush1 and CarGoal1 tasks. The converged reward and cost are plotted in fig. 5. Overall, most representation methods have a monotonic reward increasing when we set larger cost limit. When cost limit ≥ 20, the final cost performances do not exactly converge to the given threshold. This is because the corresponding unconstrained policy (i.e., reward maximization without considering cost) is already constraint satisfactory.

The FCSRL outperforms the other baselines, which suggests a better safety-related feature extraction by our method. Regarding constraint strength, FCSRL achieves very similar performance to SALE and VC models when cost > 30; however, the performance advantage of FCSRL grows, displaying the boost on safe RL from the safety-aware representation. This not only empirically validates the aforementioned intuition but also highlights the effectiveness of our method under stricter safety requirements.

5.5. Ablation Studies

The effectiveness of each components. We test it by remove each component in representation learning. We test each variant based on `TD3-Lag` on PointGoal2 and PointButton1 tasks and set cost limit as 10. The results are reported in table 1.

Table 1: Ablation study on each component in FCSRL.

	PointGoal2		PointButton1	
	reward	cost	reward	Cost
full FCSRL	12.7±2.4	10.0±7.8	10.7±2.3	9.8±6.0
w.o. dynamics loss	10.3±2.9	9.4±8.5	10.5±2.6	9.6±5.5
w.o. feasibility loss	7.1±2.7	9.8±9.4	9.0±2.9	10.2±5.8
only input z	5.9±3.3	9.7±0.8	7.3±4.3	9.6±6.3

We observe that feasibility loss plays an importance role in representation learning of FCSRL. The performance of FCSRL has a large drop when only inputting z but it is less affected by removing dynamics loss. This may because the FCSRL heavily relies on raw state capture to capture the features of dynamics in vector-state tasks if we input both state and its representation. Meanwhile, the raw state may contain additional information for reward-maximization optimization in RL, e.g., the features related to the reward but neglected by representation learning, which has also been observed in standard RL setting (Fujimoto et al., 2023).

Prediction length. Previous works observe that increasing prediction length facilitates the representation learning (Schwarzer et al., 2020; Yang & Nachum, 2021). Therefore, we further present a comparison with different prediction lengths on vector-state PointPush1 task. We still use `TD3-Lag` as base RL algorithm and set cost limit as 10.

Table 2: The performances with different prediction lengths.

Prediction length K		2	4	6	8
Forward latent	reward	-1.1±1.3	-0.6±0.9	-0.4±0.8	-0.5±1.0
	cost	8.9±11.5	10.1±13.5	8.8±10.0	10.1±16.7
TCL	reward	2.2±3.6	4.9±1.6	4.7±2.1	5.1±0.6
	cost	9.3±9.2	8.4±8.0	10.2±15.8	9.5±17.3
SALE	reward	7.7±4.4	8.6±2.4	8.6±3.0	9.1±2.6
	cost	9.4±7.4	10.5±7.1	10.5±7.1	10.5±7.3
FCSRL	reward	**9.7±2.5**	**11.6±4.3**	**12.4±3.4**	**12.6±3.8**
	cost	**10.7±8.9**	**9.6±6.8**	**9.6±8.0**	**9.3±7.3**

The results in table 2 show that most representation learning benefits as the prediction length extends, with our approach reliably surpassing the baseline methods across various lengths. However, we notice that the reward increase becomes less pronounced when $K \geq 4$. This trend suggests that the benefits of predicting longer sequences diminish for the actor and critic components of reinforcement learning.

6. Conclusion

In this paper, we present a novel framework for safe RL that leverages feasibility consistent representation learning to improve safety and efficiency. Through extensive experiments, we demonstrate that our approach outperforms existing methods by effectively balancing task performance and safety constraints. The capability of our model to extract safety-related features from complex environments and its application across various tasks underscore its potential to promote existing safe RL methods. One future direction of our work is to employ feasibility score as auxiliary signal for policy learning to achieve state-wise safety in safe RL.

Acknowledgements

The research is partly supported by the National Science Foundation under grants CNS-2047454.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Abel, D., Arumugam, D., Lehnert, L., and Littman, M. State abstractions for lifelong reinforcement learning. In *International Conference on Machine Learning*, pp. 10–19. PMLR, 2018.

Achiam, J., Held, D., Tamar, A., and Abbeel, P. Constrained policy optimization. In *International conference on machine learning*, pp. 22–31. PMLR, 2017.

Altman, E. *Constrained Markov decision processes: stochastic modeling*. Routledge, 1999.

Andrychowicz, M., Wolski, F., Ray, A., Schneider, J., Fong, R., Welinder, P., McGrew, B., Tobin, J., Pieter Abbeel, O., and Zaremba, W. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

As, Y., Usmanova, I., Curi, S., and Krause, A. Constrained policy optimization via bayesian world models. *arXiv preprint arXiv:2201.09802*, 2022.

Bansal, S., Chen, M., Herbert, S., and Tomlin, C. J. Hamilton-jacobi reachability: A brief overview and recent advances. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 2242–2253. IEEE, 2017.

Bellemare, M. G., Dabney, W., and Munos, R. A distributional perspective on reinforcement learning. In *International conference on machine learning*, pp. 449–458. PMLR, 2017.

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8): 1798–1828, 2013.

Berkenkamp, F., Turchetta, M., Schoellig, A., and Krause, A. Safe model-based reinforcement learning with stability guarantees. *Advances in neural information processing systems*, 30, 2017.

Brunke, L., Greeff, M., Hall, A. W., Yuan, Z., Zhou, S., Panerati, J., and Schoellig, A. P. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5, 2021.

Cen, Z., Liu, Z., Wang, Z., Yao, Y., Lam, H., and Zhao, D. Learning from sparse offline datasets via conservative density estimation. *arXiv preprint arXiv:2401.08819*, 2024.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Chen, X. and He, K. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15750–15758, 2021.

Chow, Y., Ghavamzadeh, M., Janson, L., and Pavone, M. Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, 18 (167):1–51, 2018.

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

Ding, D., Zhang, K., Basar, T., and Jovanovic, M. Natural policy gradient primal-dual method for constrained markov decision processes. *Advances in Neural Information Processing Systems*, 33:8378–8390, 2020.

Ding, W., Xu, C., Arief, M., Lin, H., Li, B., and Zhao, D. A survey on safety-critical driving scenario generation—a methodological perspective. *IEEE Transactions on Intelligent Transportation Systems*, 2023.

Eysenbach, B., Zhang, T., Levine, S., and Salakhutdinov, R. R. Contrastive learning as goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 35:35603–35620, 2022.

Farahmand, A.-m., Barreto, A., and Nikovski, D. Value-aware loss function for model-based reinforcement learning. In *Artificial Intelligence and Statistics*, pp. 1486–1494. PMLR, 2017.

Farquhar, G., Baumli, K., Marinho, Z., Filos, A., Hessel, M., van Hasselt, H. P., and Silver, D. Self-consistent models and values. *Advances in Neural Information Processing Systems*, 34:1111–1125, 2021.

Finn, C., Tan, X. Y., Duan, Y., Darrell, T., Levine, S., and Abbeel, P. Deep spatial autoencoders for visuomotor learning. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 512–519. IEEE, 2016.

Fisac, J. F., Lugovoy, N. F., Rubies-Royo, V., Ghosh, S., and Tomlin, C. J. Bridging hamilton-jacobi safety analysis and reinforcement learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8550–8556. IEEE, 2019.

Fujimoto, S., Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pp. 1587–1596. PMLR, 2018.

Fujimoto, S., Chang, W.-D., Smith, E. J., Gu, S. S., Precup, D., and Meger, D. For sale: State-action representation learning for deep reinforcement learning. *arXiv preprint arXiv:2306.02451*, 2023.

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

Gelada, C., Kumar, S., Buckman, J., Nachum, O., and Bellemare, M. G. Deepmdp: Learning continuous latent space models for representation learning. In *International Conference on Machine Learning*, pp. 2170–2179. PMLR, 2019.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

Grimm, C., Barreto, A., Singh, S., and Silver, D. The value equivalence principle for model-based reinforcement learning. *Advances in Neural Information Processing Systems*, 33:5541–5552, 2020.

Gu, S., Yang, L., Du, Y., Chen, G., Walter, F., Wang, J., Yang, Y., and Knoll, A. A review of safe reinforcement learning: Methods, theory and applications. *arXiv preprint arXiv:2205.10330*, 2022.

Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., and Davidson, J. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019.

Hafner, D., Lillicrap, T., Norouzi, M., and Ba, J. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

Hare, J. Dealing with sparse rewards in reinforcement learning. *arXiv preprint arXiv:1910.09281*, 2019.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., Horgan, D., Piot, B., Azar, M., and Silver, D. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Hessel, M., Danihelka, I., Viola, F., Guez, A., Schmitt, S., Sifre, L., Weber, T., Silver, D., and Van Hasselt, H. Muesli: Combining improvements in policy optimization. In *International conference on machine learning*, pp. 4214–4226. PMLR, 2021.

Huang, S., Abdolmaleki, A., Vezzani, G., Brakel, P., Mankowitz, D. J., Neunert, M., Bohez, S., Tassa, Y., Heess, N., Riedmiller, M., et al. A constrained multi-objective reinforcement learning framework. In *Conference on Robot Learning*, pp. 883–893. PMLR, 2022.

Huang, W., Ji, J., Zhang, B., Xia, C., and Yang, Y. Safe dreamerv3: Safe reinforcement learning with world models. *arXiv preprint arXiv:2307.07176*, 2023.

Ji, J., Zhang, B., Zhou, J., Pan, X., Huang, W., Sun, R., Geng, Y., Zhong, Y., Dai, J., and Yang, Y. Safety-gymnasium: A unified safe reinforcement learning benchmark. *arXiv preprint arXiv:2310.12567*, 2023.

Kaiser, L., Babaeizadeh, M., Milos, P., Osinski, B., Campbell, R. H., Czechowski, K., Erhan, D., Finn, C., Kozakowski, P., Levine, S., et al. Model-based reinforcement learning for atari. *arXiv preprint arXiv:1903.00374*, 2019.

Kiran, B. R., Sobh, I., Talpaert, V., Mannion, P., Al Sallab, A. A., Yogamani, S., and Pérez, P. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23 (6):4909–4926, 2021.

Kostrikov, I., Yarats, D., and Fergus, R. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020.

Laskin, M., Srinivas, A., and Abbeel, P. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, pp. 5639–5650. PMLR, 2020.

Lesort, T., Díaz-Rodríguez, N., Goudou, J.-F., and Filliat, D. State representation learning for control: An overview. *Neural Networks*, 108:379–392, 2018.

Li, Q., Peng, Z., Feng, L., Zhang, Q., Xue, Z., and Zhou, B. Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning. *IEEE transactions on pattern analysis and machine intelligence*, 2022.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Lin, H., Ding, W., Liu, Z., Niu, Y., Zhu, J., Niu, Y., and Zhao, D. Safety-aware causal representation for trustworthy offline reinforcement learning in autonomous driving. *IEEE Robotics and Automation Letters*, 2024.

Liu, G., Zhang, C., Zhao, L., Qin, T., Zhu, J., Li, J., Yu, N., and Liu, T.-Y. Return-based contrastive representation learning for reinforcement learning. *arXiv preprint arXiv:2102.10960*, 2021.

Liu, M., Zhu, M., and Zhang, W. Goal-conditioned reinforcement learning: Problems and solutions. *arXiv preprint arXiv:2201.08299*, 2022a.

Liu, Z., Cen, Z., Isenbaev, V., Liu, W., Wu, S., Li, B., and Zhao, D. Constrained variational policy optimization for safe reinforcement learning. In *International Conference on Machine Learning*, pp. 13644–13668. PMLR, 2022b.

Liu, Z., Guo, Z., Cen, Z., Zhang, H., Tan, J., Li, B., and Zhao, D. On the robustness of safe reinforcement learning under observational perturbations. *arXiv preprint arXiv:2205.14691*, 2022c.

Liu, Z., Guo, Z., Lin, H., Yao, Y., Zhu, J., Cen, Z., Hu, H., Yu, W., Zhang, T., Tan, J., et al. Datasets and benchmarks for offline safe reinforcement learning. *arXiv preprint arXiv:2306.09303*, 2023.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533, 2015.

Oh, J., Singh, S., and Lee, H. Value prediction network. *Advances in neural information processing systems*, 30, 2017.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ray, A., Achiam, J., and Amodei, D. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 7, 2019.

Riedmiller, M., Hafner, R., Lampe, T., Neunert, M., Degrave, J., Wiele, T., Mnih, V., Heess, N., and Springenberg, J. T. Learning by playing solving sparse reward tasks from scratch. In *International conference on machine learning*, pp. 4344–4353. PMLR, 2018.

Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839): 604–609, 2020.

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

Schwarzer, M., Anand, A., Goel, R., Hjelm, R. D., Courville, A., and Bachman, P. Data-efficient reinforcement learning with self-predictive representations. *arXiv preprint arXiv:2007.05929*, 2020.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Sootla, A., Cowen-Rivers, A. I., Jafferjee, T., Wang, Z., Mguni, D. H., Wang, J., and Ammar, H. Sauté rl: Almost surely safe reinforcement learning using state augmentation. In *International Conference on Machine Learning*, pp. 20423–20443. PMLR, 2022.

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

Sui, Y., Gotovos, A., Burdick, J., and Krause, A. Safe exploration for optimization with gaussian processes. In *International conference on machine learning*, pp. 997–1005. PMLR, 2015.

Tessler, C., Mankowitz, D. J., and Mannor, S. Reward constrained policy optimization. *arXiv preprint arXiv:1805.11074*, 2018.

Wachi, A. and Sui, Y. Safe reinforcement learning in constrained markov decision processes. In *International Conference on Machine Learning*, pp. 9797–9806. PMLR, 2020.

Wachi, A., Sui, Y., Yue, Y., and Ono, M. Safe exploration and optimization of constrained mdps using gaussian processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Wang, Y., Zhan, S. S., Jiao, R., Wang, Z., Jin, W., Yang, Z., Wang, Z., Huang, C., and Zhu, Q. Enforcing hard constraints with soft barriers: Safe reinforcement learning in unknown stochastic environments. In *International Conference on Machine Learning*, pp. 36593–36604. PMLR, 2023.

Xu, M., Liu, Z., Huang, P., Ding, W., Cen, Z., Li, B., and Zhao, D. Trustworthy reinforcement learning against intrinsic vulnerabilities: Robustness, safety, and generalizability. *arXiv preprint arXiv:2209.08025*, 2022.

Yang, M. and Nachum, O. Representation matters: Offline pretraining for sequential decision making. In *International Conference on Machine Learning*, pp. 11784–11794. PMLR, 2021.

Yang, T.-Y., Rosca, J., Narasimhan, K., and Ramadge, P. J. Projection-based constrained policy optimization. *arXiv preprint arXiv:2010.03152*, 2020.

Yao, Y., Liu, Z., Cen, Z., Zhu, J., Yu, W., Zhang, T., and Zhao, D. Constraint-conditioned policy optimization for versatile safe reinforcement learning. *arXiv preprint arXiv:2310.03718*, 2023.

Ye, W., Liu, S., Kurutach, T., Abbeel, P., and Gao, Y. Mastering atari games with limited data. *Advances in Neural Information Processing Systems*, 34:25476–25488, 2021.

Yu, D., Ma, H., Li, S., and Chen, J. Reachability constrained reinforcement learning. In *International Conference on Machine Learning*, pp. 25636–25655. PMLR, 2022.

Yue, Y., Kang, B., Xu, Z., Huang, G., and Yan, S. Value-consistent representation learning for data-efficient reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11069–11077, 2023.

Zhang, Y., Vuong, Q., and Ross, K. First order constrained optimization in policy space. *Advances in Neural Information Processing Systems*, 33:15338–15349, 2020.

A. Proof for Theoretical Analysis

A.1. Proof of Proposition 4.1

Proof. Given a trajectory $\rho = \{s_1, a_1, ..., s_T, a_T\}$ with length T, by definition, we have

$$\max_t \left(c\left(s_t, a_t\right)\right) = 1 - \mathbf{1}\left(\bigcap_{t=1}^{T} c\left(s_t, a_t\right) > 0\right), \tag{14}$$

which means along the trajectory ρ, the maximal value of $c\left(s_t, a_t\right)$ is the complementary of the event that at least one of the state-action pair $\left(s_t, a_t\right)$ violates the safety constraint. Then add expectations on both sides of (14), we can get:

$$\begin{aligned}\mathbb{E}_{\rho\sim\pi} \max_t \left(c\left(s_t, a_t\right)\right) =& 1 - \mathbb{E}_{\rho\sim\pi}\mathbf{1}\left(\bigcap_{t=1}^{T} c\left(s_t, a_t\right) > 0\right) \\ =& 1 - \Pr\left(\bigcap_{(s_t,a_t)\sim\rho} \{c(s_t, a_t) = 0\}\right)\end{aligned} \tag{15}$$

Since with definition (4), the left side is the feasibility function. Then we can conclude that:

$$1 - F^{\pi}(s, a) = \Pr\left(\bigcap_{(s_t,a_t)\sim\rho} \{c(s_t, a_t) = 0\}\right) \tag{16}$$

□

A.2. Proof of Proposition 4.3

Proof. Given trajectory ρ with length T, we first define functions $\hat{F}(i)$ and $\hat{V}(i)$:

$$\hat{F}(i) := \max_{t\geq i} \gamma^{t-i} c(s_t, a_t), \quad \hat{V}(i) := \sum_{t=i}^{T} \gamma^{t-i} c(s_t, a_t), \tag{17}$$

where i means time step. Then we can first observe that:

$$\hat{F}(i) \leq \hat{V}(i), \quad \forall\, i; \tag{18}$$

Then we come to the temporal smoothness part. The smoothness is discussed with the following conditions:

1. If $c(i) = 0,\ \forall i \leq T$: Then $\sum \hat{F}(i) = \sum \hat{V}_c(i) = 0$. In this case, $L(\hat{F}, \rho) = L(\hat{V}_c, \rho) = 0$.

2. If exist one unique i_0 such that $c(i_0) = 0$, then for the time step $i > i_0$, $\sum_{i>i_0} \hat{F}(i) = \sum_{i>i_0} \hat{V}_c(i) = 0$ based on the analysis above. For the time step $i \leq i_0$:

$$\hat{F}(i) = 1, \quad \hat{V}_c(i) = \gamma^{i_0 - i}, \tag{19}$$

Then we can get the smoothness of both functions:

$$L(\hat{F}, \rho) = \mathbb{E}_{\rho}|\hat{F}(t) - \hat{F}(t+1)| = \frac{1}{T}, \tag{20}$$

$$L(\hat{V}_c, \rho) = \mathbb{E}_{\rho}|\hat{V}_c(t) - \hat{V}_c(t+1)| = \frac{1}{T}(1 + \sum_{t=1}^{i_0-1} \gamma^{t-1}(1-\gamma)) = \frac{2-\gamma^{i_0-1}}{T}, \tag{21}$$

Then we can get that $L(\hat{V}_c, \rho) \geq L(\hat{F}, \rho)$.

3. If exist two time steps $i_0 < i_1$ such that $c(i_0) = c(i_1) = 0$.

$$\sum_{t=i_0}^{i_1-1} L(\hat{F}) = \hat{F}(i_0) - \gamma^{i_1-1}\hat{F}(i_1) + \sum_{t=i_0+1}^{i_1-1} \gamma^{t-1}(1-\gamma), \tag{22}$$

   Since $\hat{F}(i_0) = \hat{F}(i_1) = 1$, we can get:

$$\sum_{t=i_0}^{i_1-1} L(\hat{F}) = 2(1-\gamma^{i_1-i_0-1}). \tag{23}$$

   Then we turn to the smoothness score sum from time step 0 to t_0 for the cost value function $\hat{V}_c$:

$$\begin{aligned}\sum_{t=i_0}^{i_1-1} L(\hat{V}_c) &= 1 + \gamma^{i_1-i_0}\hat{V}_c(i_1) - \gamma^{i_1-i_0-1}\hat{V}_c(i_1) + \sum_{t=i_0+1}^{i_1-1}(1-\gamma)\hat{V}_c(i_1) \\ &= \hat{V}_c(i_1)(1-\gamma^{t_0-1}) + 1 + \hat{V}_c(i_1)(\gamma^{i_1-i_0} - \gamma^{i_1-i_0-1}) \\ &= 1 + \hat{V}_c(i_1)(1 - 2\gamma^{i_1-i_0-1} + \gamma^{i_1-i_0})\end{aligned} \tag{24}$$

   Then since $\hat{V}_c(t_0) \geq \hat{F}(t_0) = 0$, we can derive that:

$$\begin{aligned}\sum_{t=i_0}^{i_1-1} L(\hat{V}_c) &= 1 + \hat{V}_c(i_1)(1 - 2\gamma^{i_1-i_0-1} + \gamma^{i_1-i_0}) \\ &\geq 1 + (1 - 2\gamma^{i_1-i_0-1} + \gamma^{i_1-i_0}) \\ &= 2(1-\gamma^{i_1-i_0-1}) + \gamma^{i_1-i_0}) \\ &\geq 2(1-\gamma^{i_1-i_0-1}) = \sum_{t=i_0}^{i_1-1} L(\hat{F}),\end{aligned} \tag{25}$$

   which means for the trajectory between i_0 and i_1, we have the averaged smoothness score order: $\mathbb{E}_{\rho[i_0:i_1]}\hat{F} \leq \mathbb{E}_{\rho[i_0:i_1]}\hat{V}_c$.

4. For other conditions, the trajectories can be separated into sub-trajectories with conditions 1, 2, 3, then the smoothness order is $L(\hat{V}_c, \rho) \geq L(\hat{F}, \rho)$.

□

B. Supplementary Materials for Experiments

B.1. Dynamics loss

For dynamics loss function $D^{\text{dyn}}(z_1, z_2)$ in eq.(3), we use SimSiam (Chen & He, 2021) as self-supervision loss function. Specifically, we employ two trainable projection functions p_1, p_2 (modeled as 2-layer MLPs), the loss function is

$$\ell^{\text{dyn}}(z_1, z_2) = -\text{cosine}\langle p_2(p_1(z_1)), \text{sg}(p_1(z_2))\rangle, \tag{26}$$

where cosine$\langle \cdot, \cdot \rangle$ is the consine similarity and sg means stopping gradient. During training, we will update p_1, p_2 along with the parameters of encoder, transition model, and prediction head θ.

B.2. Landscape

The landscape shown in fig. 6 corresponds to the red frame in the bird-view of the task. From the figure, we can observe that the landscape of value function is less smoother than feasibility score. Meanwhile, the peaks in landscape of feasibility score correspond to the four blue square obstacles, which shows a better match than value function. We also conduct a comparison of the embeddings learned by value consistent model and feasibility consistent model in Appendix B.3.

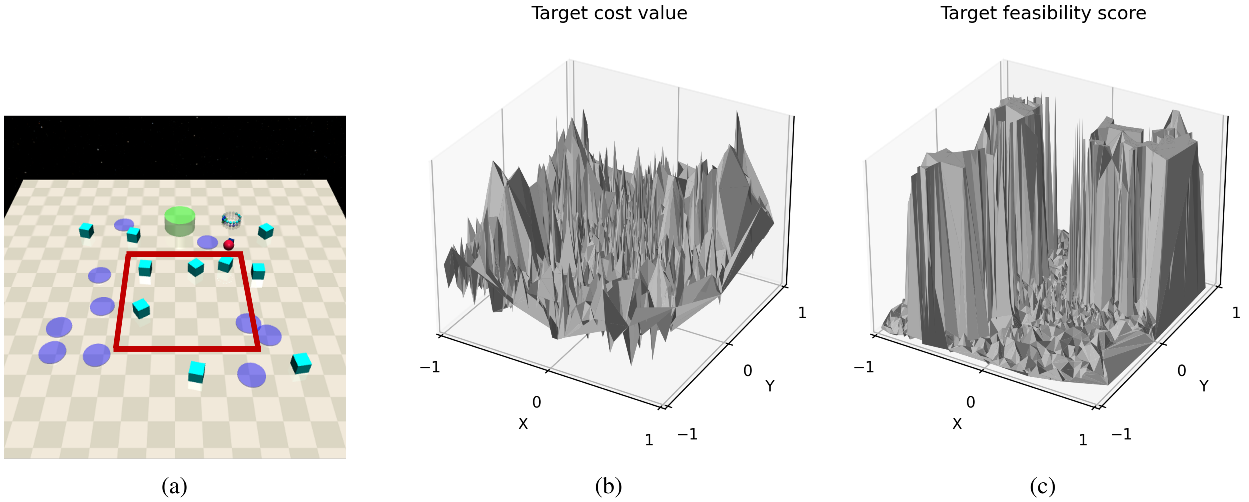


Figure 6: (a): The overview of the tested PointGoal2 task. Red point is the agent, blue squares are the obstacles, blue circles are hazards, and green cylinder is the goal, the agent should reach the goal with colliding with obstacles or stepping into hazards. We test the value and feasibility score of agent in the same positions in the *red bounding box*. (b)(c): The landscape of target cost value and feasibility score obtained by bootstrapping in PointGoal2 task. The values of them have been rescaled.

B.3. Comparison between the embedding learned by Value Consistent model and FCSRL

To further compare the learned embeddings by VC model and FCSRL, we test their quality by measuring the capability of predicting cost value $V_c(s)$ and feasibility score $F_c(s)$. Specifically, we

1. Train a SALE policy on PointGoal2 task (vector state) and sample 50 trajectories by SALE policy, store them to buffer B.

2. Train a VC policy and FCSRL policy seperately, denote their encoder as g_v and g_f; obtain corresponding embeddings of states in buffer B as $\{z_v\}, \{z_f\}$; obtain target cost value $v_c^{(m)}$ and target feasibility $f_c^{(m)}$ by bootstrap estimate, respectively by VC and FCSRL.

3. Use linear regression models to train 4 models (input: z_v or z_f; output v_c or f_c); record the final MSE.

4. Repeat above for several seeds.

The final results are reported in table 3. We can find that FCSRL actually achieves similar MSE to VC model on cost value prediction although VC model has explicitly regressed the embedding to V_c during training. Meanwhile, FCSRL has smaller MSE on feasibility score prediction. We also plot the prediction results on V_c in fig. 7. The comparison shows that the value function information is harder to extract via representation learning and the effectiveness of value consistent model is not very remarkable. In contrast, the feasibility score is easier to learned.

Table 3: The prediction MSE of embeddings from FCSRL and VC model.

	MSE of cost value	MSE of feasibility
FCSRL	17.6±5.2	0.139±0.047
VC model	17.3±4.6	0.217±0.096

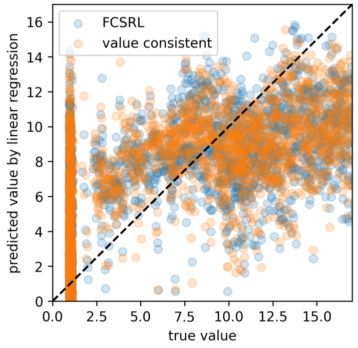


Figure 7: The comparison of VC model and FCSRL on prediction of V_c.

B.4. Comparison with other safe RL baselines

We add a comparison of FCSRL and other safe RL baselines on Safety-Gymnasium tasks. We adopt FSRL[2] as the implementations of baselines to test their performances. Our method FCSRL adopts TD3-Lag as base algorithm. The cost threshold ϵ is 10 for all tasks.

Table 4: The value is averaged over 5 seeds. **Bold** value indicates the algorithm can roughly satisfy the constraint (cost return ≤ 11). Blue value indicates the highest rewards ($\geq 0.95*$ highest reward) when satisfying the constraint.

Method	PointGoal1		PointButton1		PointPush1		PointGoal2		CarGoal1		CarButton1	
	reward	cost	reward	cost	reward	cost	reward	cost	reward	cost	reward	Cost
CPO [(Achiam et al., 2017)]	**4.2**±1.3	**10.8**±9.2	-0.3±0.6	12.4±7.7	0.4±0.6	15.9±10.0	-0.2±0.5	23.8±16.0	7.5±3.3	14.8±6.4	0.1±0.4	**9.4**±3.5
PPO-Lag [(Ray et al., 2019)]	**13.4**±1.6	**10.2**±4.5	**1.6**±1.4	**9.5**±3.2	**2.0**±0.9	**9.6**±7.5	3.0±1.7	22.6±8.5	14.9±4.8	26.1±7.5	-0.6±1.1	19.4±17.2
FOCOPS [(Zhang et al., 2020)]	10.8±4.0	12.7±9.6	6.6±3.4	42.6±18.0	**0.6**±0.3	**10.2**±7.6	6.7±3.7	73.6±39.8	15.0±2.2	36.2±6.9	0.6±1.7	17.2±19.6
CVPO [(Liu et al., 2022b)]	**5.5**±3.5	**9.9**±7.7	**0.7**±0.5	**10.8**±7.6	**3.3**±1.5	**11.0**±5.6	0.2±0.6	35.6±33.2	**5.8**±2.0	**10.2**±6.6	**-0.8**±0.7	**7.3**±3.2
FCSRL (Ours)	**24.4**±1.4	**9.3**±5.8	**10.5**±2.3	**10.1**±6.3	**11.8**±4.5	**9.7**±8.1	**13.5**±2.6	**9.7**±7.8	**27.6**±2.0	**9.9**±6.4	**3.6**±1.3	**10.4**±7.3

B.5. More details of experiment settings

B.5.1. The performances of random policy and unconstrained PPO policy

We report the reward performance of random policy R_l and unconstrained PPO policy R_h (which is used to normalize reward in fig. 3) in table 5.

B.5.2. Network structure

We adopt 2-layer MLPs for all transition models, prediction heads, projection networks in SimSiam loss, including both our method and baselines, both vector-state and image-based tasks. Besides, for encoder, we still adopt a 2-layer MLP for state-vector tasks while using a 4-layer CNN for image-based tasks. For fair comparison, we adopt the same neural network architecture for all baselines and our method.

B.5.3. Hyperparameters

We adopt the same hyperparameters for all tasks in the same domain. For the vector-input and image-based tasks, we also keep most hyperparameters the same. The hyperparameters are summaries in table 6. More details can be found in codes

[2] https://github.com/liuzuxin/FSRL/tree/main

Table 5: The performances of random and unconstrained PPO policies.

	random policy R_l	unconstrained PPO R_h
PointGoal1	-0.35	23.3
PointButton1	-0.01	17.6
PointPush1	-0.41	3.1
PointGoal2	-0.20	22.2
CarGoal1	-2.08	24.8
CarButton1	-1.38	9.0

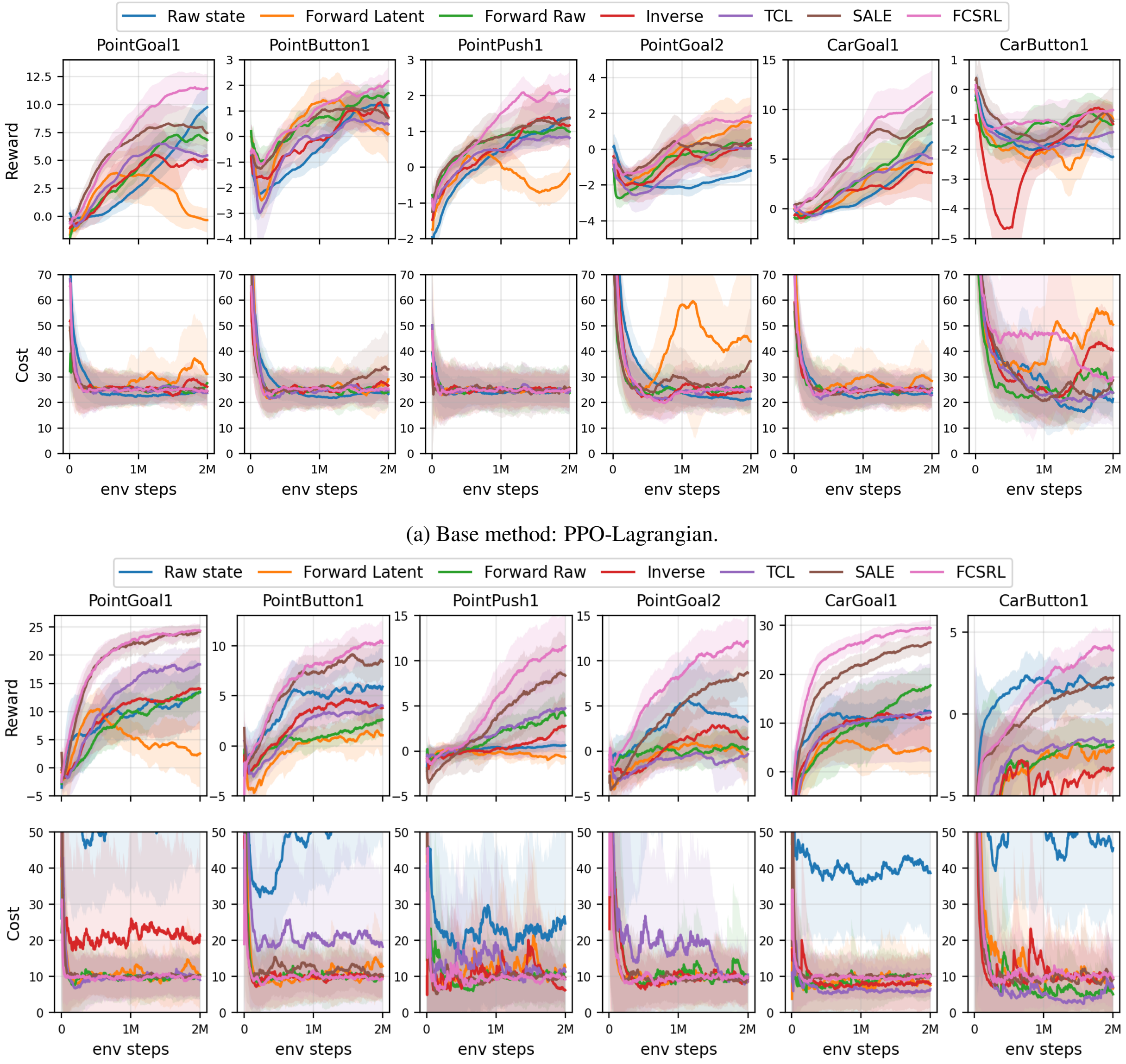
provided in `https://github.com/czp16/FCSRL`.

Table 6: The hyperparameters adopted in experiments.

Hyperparameter	Value
hidden layers of actor	[256, 256]
hidden layers of critic	[256, 256]
hidden layers of transition h	[256, 256]
hidden layers of encoder g for vector tasks	[256, 256]
NN optimizer	Adam
NN learning rate	3e-4
Number of bins in discrete regression	63
discount factor γ	0.99
prediction length K	4
PID cofficient for Lagrangian $[K_p, K_i, K_d]$	[0.02, 0.005, 0.01]
dimension of embedding z	64/128 for vector/image
soft update coefficient for target network	0.05
coefficient of feasibility loss λ^{fea}	2.0

B.6. Training curves

We attach the training curves below.



(a) Base method: PPO-Lagrangian.

(b) Base method: TD3-Lagrangian.

Figure 8: The training curves of different representation learning methods based on PPO-Lagrangian and TD3-Lagrangian.