OASIS: Conditional Distribution Shaping for Offline Safe Reinforcement Learning

Yihang Yao[*1], **Zhepeng Cen**[*1], **Wenhao Ding**[1], **Haohong Lin**[1], **Shiqi Liu**[1],
Tingnan Zhang[2], **Wenhao Yu**[2], **Ding Zhao**[1]
[1] Carnegie Mellon University, [2] Google DeepMind
[*] Equal contribution, {yihangya, zcen}@andrew.cmu.edu

Abstract

Offline safe reinforcement learning (RL) aims to train a policy that satisfies con-
straints using a pre-collected dataset. Most current methods struggle with the
mismatch between imperfect demonstrations and the desired safe and rewarding
performance. In this paper, we mitigate this issue from a *data-centric* perspective
and introduce OASIS (c**O**ndition**A**l di**S**tribut**I**on **S**haping), a new paradigm in
offline safe RL designed to overcome these critical limitations. OASIS utilizes a
conditional diffusion model to synthesize offline datasets, thus shaping the data dis-
tribution toward a beneficial target domain. Our approach makes compliance with
safety constraints through effective data utilization and regularization techniques to
benefit offline safe RL training. Comprehensive evaluations on public benchmarks
and varying datasets showcase OASIS's superiority in benefiting offline safe RL
agents to achieve high-reward behavior while satisfying the safety constraints, out-
performing established baselines. Furthermore, OASIS exhibits high data efficiency
and robustness, making it suitable for real-world applications, particularly in tasks
where safety is imperative and high-quality demonstrations are scarce. More details
are available at the website https://sites.google.com/view/saferl-oasis/home.

1 Introduction

Offline Reinforcement Learning (RL), which aims to learn high-reward behaviors from a pre-collected
dataset [1, 2], has emerged as a powerful paradigm for handling sequential decision-making tasks
such as autonomous driving [3, 4, 5, 6], and robotics [7, 8, 9, 10, 11]. Although standard offline RL
has achieved remarkable success in some environments, many real-world tasks cannot be adequately
addressed by simply maximizing a scalar reward function due to the existence of various safety
constraints that limit feasible solutions. The requirement for *safety*, or constraint satisfaction, is
particularly crucial in RL algorithms when deployed in real-world tasks [12, 13, 14, 15, 16, 17, 18, 19].

To develop an optimal policy within a constrained manifold [20, 21], *offline safe RL* has been actively
studied in recent years, offering novel ways to integrate safety requirements into offline RL [22].
Existing research in this area incorporates techniques from both offline RL and safe RL, including the
use of stationary distribution correction [23, 24], regularization [25, 26], and constrained optimization
formulations [27]. Researchers have also proposed the use of sequential modeling methods, such as
the decision transformer [28, 29] and the decision diffuser [30, 31] to achieve advantageous policies
and meet safety requirements.

Although these methods show promise, it is difficult to handle state-action pairs that are absent from
the dataset, which is known notably as out-of-distribution (OOD) extrapolation issues [31, 32, 33,
34, 35, 36, 37]. To solve this, many works utilize regularization methods to push the policy toward
behavior policy to achieve pessimism [35, 36]. However, this approach worsens the situation when
the dataset is imbalanced and biased: regularization by imperfect demonstrations such as datasets

composed predominantly of low-reward data or containing few safe trajectories collected using unsafe behavior policies. This regularization also leads to another challenge: striking the optimal balance between learning objectives such as task utility efficiency and safety requirements, leading to reward degradation or aggressive behavior [28, 38, 39].

To address these challenges, we introduce a *data-centric* learning paradigm in offline safe RL, OASIS (c**O**ndition**A**l di**S**tribut**I**on **S**haping), which focuses on improving the training dataset quality by steering the offline data distribution to a beneficial target domain as shown in Fig. 1. OASIS distills knowledge from the imperfect dataset, and generates rewarding and safe data using a conditional diffusion model according to the safety preference to benefit offline safe RL training. This *data-centric* approach is parallel to and compatible with general *model-centric* offline safe RL
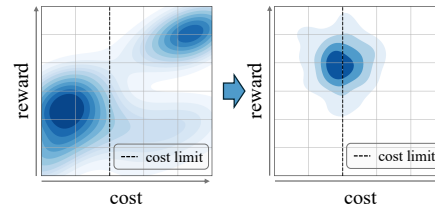

Figure 1: An example of distribution shaping in offline safe RL. We generate a low-cost and high-reward dataset from the original dataset for subsequent RL training.

algorithms which emphasize improvements to the learning algorithm and model architecture. The key contributions are summarized as follows.

1. Identification of the safe dataset mismatch (SDM) problem in offline safe RL. We identify the mismatch between the behavior policy and the target policy and investigate the underlying reasons for performance degradation with this condition.

2. Introduction of the OASIS method to address the SDM problem. To the best of our knowledge, this is the first successful application of a distribution shaping paradigm within offline safe RL. Our theoretical analysis further provides insights on performance improvement and safety guarantees.

3. A comprehensive evaluation of our method across various offline safe RL tasks. The experiment results demonstrate that OASIS outperforms baseline methods in terms of both safety and task efficiency for varying tasks and datasets.

2 Related Work

Offline RL. Offline RL addresses the limitations of traditional RL, which requires interaction with the environment. The key literature on offline RL includes BCQ [35], which mitigates the extrapolation error using a generative model, and CQL [40], which penalizes the overestimation of Q-values for unseen actions. BEAR [41] further addresses the extrapolation error by constraining the learned policy to stay close to the behavior policy. OptiDICE [24] directly estimates the stationary distribution corrections of the optimal policy, and COptiDICE [23] extends the method to the constrained RL setting. Recent advances have increasingly focused on the use of data generation to improve policy learning. S4RL [42] shows that surprisingly simple augmentations can dramatically improve policy performance. [43] explores leveraging unlabeled data to improve policy robustness, while [44] proposes survival instincts to enhance agent performance in challenging environments. Counterfactual data augmentation is another promising direction in offline RL [45, 46, 47, 48], highlighting the potential of data generation to significantly improve efficiency and effectiveness.

Safe RL. Safe RL is formulated as a constrained optimization to maximize reward performance while satisfying the pre-defined safety constraints [20, 49, 50, 51, 52, 53, 54, 55]. Primal-dual framework is one common approach to solve safe RL problem [56, 57, 58, 59, 60, 61, 62, 63]. Another line of work for safe RL is to extend to offline settings, which learn from a fixed dataset to achieve both high reward and constraint satisfaction [64, 29]. Among them, [26, 65, 66, 67] tailor online prime-dual-style algorithms to reduce the out-of-distribution issue in the offline setting. [28, 68] use decision transformer [69] to avoid value estimation and exhibit consistent performances across various tasks. In addition to these *Model-centric* approaches, *Data-centric* approaches, which emphasize improving or optimizing the quality of the dataset used for model training [70, 71, 72, 73], have gained more attention in recent studies. While some previous work proposed methods for learning from safe demonstration [28, 74] or relabeling data to achieve conservativeness [28], how to systematically curate datasets for offline safe learning remains a largely unexplored area.

Diffusion Models for RL. Diffusion models have recently gained attention in RL for their capabilities in planning and data generation [75, 76, 77, 78]. Specifically, Diffuser [79] uses a diffusion process to plan the entire trajectory in complex environments. [80] extends this to the Decision Diffuser,

which conditions the diffusion process on specific goals and rewards to improve decision-making. SafeDiffuser [81] and FISOR [31] enhance safety by ensuring the planned trajectories satisfying constraints. Combined with the data augmentation capability of diffusion models, AdaptDiffuser [82] achieves state-of-the-art results on offline RL benchmarks. [83] proposes Synthetic Experience Replay, leveraging diffusion models to create synthetic experiences for more efficient learning. [84] demonstrates that diffusion models are effective planners and data synthesizers for multi-task RL, showcasing their versatility and efficiency. In this work, we investigate the power of diffusion models for safe RL, where the balance between reward and cost presents further complexities.

3 Problem Formulation

3.1 Safe RL with Constrained Markov Decision Process

We formulate Safe RL problems under the Constrained Markov Decision Process (CMDP) framework [85]. A CMDP $\mathcal{M}$ is defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, c, \gamma, \mu_0)$, where $\mathcal{S} \in \mathbb{R}^m$ is the state space, $\mathcal{A} \in \mathbb{R}^n$ is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition function, $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the reward function, $c : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}_{\geq 0}$ is the cost function, γ is the discount factor, and $\mu_0 : \mathcal{S} \to [0, 1]$ is the initial state distribution. Note that this work can also be applied to multiple-constraint tasks, but we use a single-constraint setting for easy demonstration. A safe RL problem is specified by a CMDP and a constraint threshold $\kappa \in [0, +\infty)$. Denote $\pi \in \Pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ as the policy and $\tau = \{(s_1, a_1, r_1, c_1), (s_2, a_2, r_2, c_2), \dots\}$ as the trajectory. The stationary state-action distribution under the policy π is defined as $d^\pi(s, a) = (1 - \gamma) \sum_t \gamma^t \Pr(s_t = s, a_t = a)$. The reward and cost returns are defined as $R(\tau) = \sum_\tau r$, and $C(\tau) = \sum_\tau c$. The value function is $V_{\mathbf{f}}^\pi(\mu_0) = \mathbb{E}_{\tau \sim \pi, s_0 \sim \mu_0}[\sum_{t=0}^\infty \gamma^t \mathbf{f}_t], \mathbf{f} \in \{r, c\}$, which is the expectation of discounted return under the policy π and the initial state distribution μ_0. The goal of safe RL is to find the optimal policy π^* that maximizes the expectation of reward return while constraining the expectation of cost return to the threshold κ:

$$\pi^* = \arg \max_\pi \mathbb{E}_{\tau \sim \pi}\big[R(\tau)\big], \quad s.t. \quad \mathbb{E}_{\tau \sim \pi}\big[C(\tau)\big] \leq \kappa. \tag{1}$$

3.2 Regularized offline safe RL

For an offline safe RL problem, the agent can only access a pre-collected dataset $\mathcal{D} = \cup_i \mathcal{D}_i$, where $\mathcal{D}_i \sim \pi_i^B$ is collected by the behavior policy $\pi_i^B \in \Pi^B$. To solve the problem in Eq. (1), we convert the constraint optimization problem into an unconstrained form:

$$(\pi^*, \lambda^*) = \arg \max_\lambda \min_\pi \mathcal{J}(\pi, \lambda), \quad \mathcal{J}(\pi, \lambda) = -\mathbb{E}_{\tau \sim \pi} R(\tau) + \lambda(\mathbb{E}_{\tau \sim \pi} R(\tau) - \kappa). \tag{2}$$

The primal-dual-based algorithm solves the optimal policy π^* and the dual variable λ^* by updating (π, λ) iteratively [86, 23, 66]. In offline safe RL tasks, a regularization term is usually introduced to prevent the action OOD issue [87], that is, the objective is converted to:

$$(\pi^*, \lambda^*) = \arg \max_\lambda \min_\pi \mathcal{J}_{\text{off}}(\pi, \lambda), \quad \mathcal{J}_{\text{off}}(\pi, \lambda) = \mathcal{J}(\pi, \lambda) + wL(\pi, \pi^B), \tag{3}$$

where $w > 0$ is a constant weight, $L(\pi, \pi^B)$ is a regularization term and π^B is the empirical behavior policy and can be viewed as a mixture of $\{\pi_i^B\}$. Practically, regularization is formulated as the MSE regularization [88] or the evidence lower bound regularization [35, 41]. In offline safe RL, there are two main challenges: (1) **Distribution shift** [25]. The agent has poor generalizability when facing OOD state-action pairs during online evaluation; and (2) **Efficiency-safety performance balancing** [28]. The agent tends to be over-conservative or aggressive when overestimating or underestimating the safety requirements.

4 Method

In this section, we first identify the *safe dataset mismatch* (SDM) problem, which leads to performance degradation when solving the regularized offline safe RL objective in Eq. (3). Then we present the proposed OASIS (c**O**ndition**A**l di**S**tribut**I**on **S**haping) method to solve this problem. In contrast to the *model-centric* safe RL approaches that focus on optimizing policy update process and network architecture, OASIS is a *data-centric* learning method that aims to improve the quality of the dataset, thus benefiting offline safe RL training. The OASIS method utilizes the diffusion model to realize conditional distribution shaping, solving the challenges mentioned above, thus benefiting offline safe RL training. Following the proposed algorithm, we provide a theoretical guarantee of the safety performance of the policy learned in this paradigm.

4.1 Safe Dataset Mismatch Problem

The regularized offline safe RL objective in Eq. (3) pushes policy to behavior policy to prevent action OOD issues [36]. When given an imperfect dataset, the state-action distribution deviates from the optimal distribution, and the SDM problem arises: if the behavior policy is too conservative with low costs and low rewards, it leads to task efficiency degradation; if the behavior policy is too aggressive with high costs and high rewards, it leads to safety violations. To further investigate the SDM problem and the effect of dataset distribution on offline safe RL, we define the **tempting dataset** and **conservative dataset**, which are based on tempting and conservative policies:

Definition 1 (Tempting policy [51] and conservative policy). The tempting policy class is defined as the set of policies that have a higher reward return expectation than the optimal policy, and the conservative policy class is defined as the set of policies that have lower reward and cost return expectations than the optimal policy:
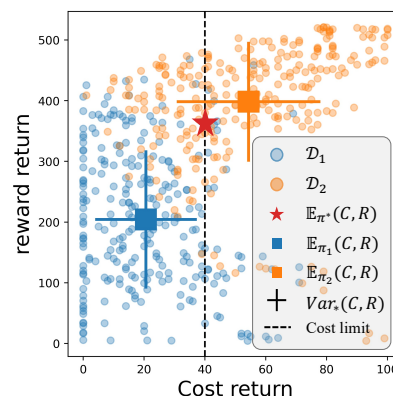

Figure 2: $\mathcal{D}_1$ is a conservative dataset, and $\mathcal{D}_2$ is a tempting dataset. Each point represents $(C(\tau), R(\tau))$ of a trajectory τ in the dataset.

$$\Pi^T := \{\pi : V_r^\pi(\mu_0) > V_r^{\pi^*}(\mu_0)\}, \quad \Pi^C := \{\pi : V_r^\pi(\mu_0) < V_r^{\pi^*}(\mu_0), V_c^\pi(\mu_0) < V_c^{\pi^*}(\mu_0)\}. \quad (4)$$

Intuitively, a tempting policy is a more rewarding but less safe policy than the optimal one, and a conservative policy is with lower cost but less rewarding. According to these policies, we define two types of datasets:

Definition 2 (Tempting and conservative dataset). For an offline dataset $D_i \sim \pi_i$, if $\pi_i \in \Pi^B \cap \Pi^T$, then the dataset is tempting; if $\pi_i \in \Pi^B \cap \Pi^C$, then the dataset is conservative.

Staying within the tempting dataset distribution results in tempting (unsafe) behavior, while staying within the conservative dataset distribution causes reward degradation. A theoretical analysis of performance degradation due to the SDM problem is presented in Sec. 4.4. Fig. 2 illustrates examples of both conservative and tempting datasets.

It is important to note that tempting and conservative datasets are prevalent in offline safe RL since optimal policies are rarely available for data collection. The SDM problem is a distinct feature of offline safe RL, indicating that training the policy on either tempting or conservative datasets will violate safety constraints or result in sub-optimality, both of which are undesirable. Therefore, addressing the SDM problem is essential for the development of regularized offline safe RL algorithms.

4.2 Mitigating the Safe Dataset Mismatch

Inspired by recent research works on *data-centric* learning [70], which emphasizes the quality of data used for training the agent, we focus on enhancing the data pipeline to ensure the agent learns effectively instead of modifying the RL algorithm or model architecture. We propose to use the distribution shaping of the dataset to mitigate the SDM problem, that is, generating a new dataset $\mathcal{D}_g$ by reshaping the original data distribution. As shown in Fig. 1, the key
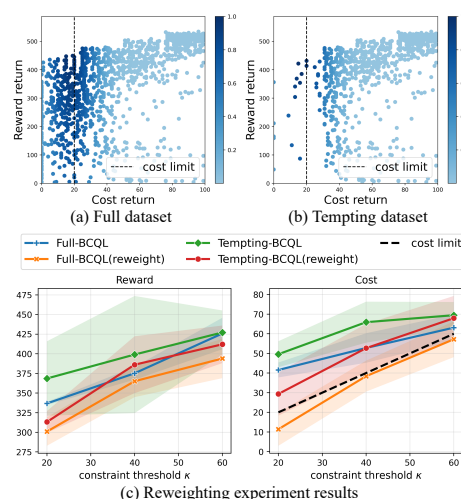

Figure 3: (a) Reweighting in the dataset with comprehensive coverage. (b) Reweighting in the tempting dataset. (c) Performance evaluation with different weights and datasets.

idea is to adjust the dataset distribution towards the optimal distribution under π^*, reducing the distribution discrepancy and mitigating the SDM problem, thus both mitigating the action OOD issue and balancing efficiency and safety in offline safe RL.
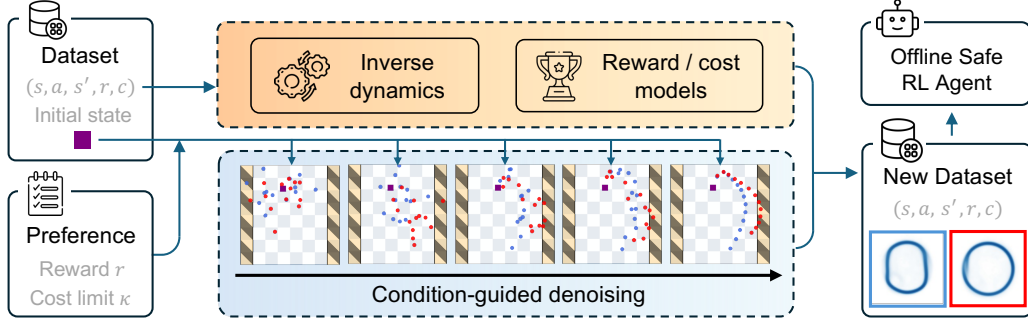
Figure 4: OASIS: a *data-centric* approach for offline safe RL. Conditioned on the human preference, OASIS first curates an offline dataset with a conditioned diffusion data generator and learned labeling models, then trains safe RL agents with this generated dataset.

Among the *data-centric* algorithms, Dataset reweighting (weighted sampling) which assigns different sampling weights to data points, is a straightforward way to shape the data distribution [39, 89]. In the offline RL domain, researchers proposed methods to assign high weights to data that achieve high rewards and superior performance in many RL tasks [39, 89].

To validate this idea, we deploy a Boltzmann energy function considering both the reward and the cost for the reweighing strategy to solve the problem (see Appendix B for details). The experimental results, shown in Fig. 3, validate the effectiveness of this distribution shaping method when the coverage of the dataset is complete.

However, for a more general case where we can only access the low-quality dataset (e.g., tempting datasets in Fig. 3), simply performing data reweighting does not work well due to the absence of necessary data. Thus, we propose to use a conditional generative model for more flexible distribution shaping, which generates new data by stitching sub-optimal trajectories for offline training.

4.3 Constraint-Conditioned Diffusion Model as Data Generator

To overcome the limitation of reweighing methods, we propose using diffusion models to generate the dataset that fits the target cost limit to achieve distribution shaping. In the following, we introduce the details of the generator training and dataset generation phases.

Training. In previous works [82, 79], the trajectory planning in offline RL can be viewed as the sequential data generation: $\boldsymbol{\tau} = [s_0, s_1, \ldots, s_{L-1}]$, where $\boldsymbol{\tau}$ is a subsequence of trajectory with length L. Denote $\boldsymbol{x}_k(\boldsymbol{\tau})$ and $\boldsymbol{y}(\boldsymbol{\tau})$ as the k-step denoising output of the diffusion model and the denoising conditions such as reward and cost returns, respectively. Then the forward diffusion process is to add noise to $\boldsymbol{x}_k(\boldsymbol{\tau})$ and gradually convert it into Gaussian noise:

$$q\left(\boldsymbol{x}_k(\tau) \mid \boldsymbol{x}_{k-1}(\tau)\right) := \mathcal{N}\left(\boldsymbol{x}_k(\tau); \sqrt{1 - \beta_k}\boldsymbol{x}_{k-1}(\tau), \beta_k \boldsymbol{I}\right), \; k = 1, ..., K \tag{5}$$

where β_k is a pre-defined beta schedule, K is the total denoising timestep. Then the trainable denoising step aims at gradually converting the Gaussian noise back to a valid trajectory:

$$p_\theta\left(\boldsymbol{x}_{k-1}(\tau) \mid \boldsymbol{x}_k(\tau), \boldsymbol{y}(\boldsymbol{\tau})\right) := \mathcal{N}\left(\boldsymbol{x}_{k-1}(\tau) \mid \mu_\theta\left(\boldsymbol{x}_k(\tau), \boldsymbol{y}(\tau), k\right), \Sigma_k\right), \tag{6}$$

where θ is the trainable parameter. We use a simplified surrogate loss [90] for optimization:

$$\mathcal{L}_{\text{denoise}} := \mathbb{E}_{\boldsymbol{x}_0(\tau) \sim q, \epsilon \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})}\left[\left\|\epsilon - \epsilon_\theta\left(\boldsymbol{x}_k(\tau), \boldsymbol{y}(\tau), k\right)\right\|^2\right]. \tag{7}$$

In this work, we use the classifier-free guidance [91] for conditional data generation. The condition $\boldsymbol{y}(\tau)$ in Eq. (6) and Eq. (7) is set to $\boldsymbol{y}(\tau) = [C(\tau), R(\tau)]$. Thus, the denoising process depends on the target reward and cost returns of the planned subtrajectory. During training time, the diffusion model learns both an unconditional denoising core $\epsilon_\theta\left(\boldsymbol{x}_k(\tau), \varnothing, k\right)$ and a conditional denoising core $\epsilon_\theta\left(\boldsymbol{x}_k(\tau), \boldsymbol{y}(\tau), k\right)$. We adopt masking [80] for the training to zero out the condition of one training trajectory and categorize it as the $\varnothing$ class with probability $0 < p < 1$. Within the given raw dataset, we also train an inverse dynamics model $\hat{f} : \mathcal{S} \times \mathcal{S} \to \mathcal{A}$, and reward and cost models $\hat{r}(s, a, s'), \hat{c}(s, a, s') : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ for labeling.

Generation. After obtaining a trained model, the next step is to generate a new dataset following the conditions. For diffusion model inference, the denoising core $\epsilon_\theta(\boldsymbol{x}_k(\tau), \boldsymbol{y}, k)$ is calculated by:

$$\epsilon_\theta(\boldsymbol{x}_k(\tau), \boldsymbol{y}(\tau), k) = \epsilon_\theta(\boldsymbol{x}_k(\tau), \varnothing, k) + w_\alpha(\epsilon_\theta(\boldsymbol{x}_k(\tau), \boldsymbol{y}_c(\tau), k) - \epsilon_\theta(\boldsymbol{x}_k(\tau), \varnothing, k)), \quad (8)$$

where $w_\alpha > 0$ is a constant guidance scale and $\boldsymbol{y}_c := [\hat{C}, \hat{R}]$ is the generation condition, we set $\hat{C} \le \kappa$ to align safety preference. For guided generation, we fix the initial state, which means that we replace the initial state of each k-step noised trajectory as $\boldsymbol{x}_k[0] = \boldsymbol{x}_0[0]$[1].

After generating one subtrajectory $\tau_g = \boldsymbol{x}_0$, we can get the state and action sequence $s_g = \tau_g[: -1], s'_g = \tau_g[1 :], a_g = \hat{f}(s_g, s'_g)$, then label the data $r_g = \hat{r}(s_g, a_g, s'_g), c_g = \hat{c}(s_g, a_g, s'_g)$. Finally, we get a generated dataset $\mathcal{D}_g = \{s_g, a_g, s'_g, r_g, c_g\}$ with $|\tau_g| - 1$ transition pairs. With this new dataset, we can further train offline safe RL agents.

In this work, we consider `BCQ-Lag` [35, 86] as the base offline safe RL algorithm. The process of generating one subtrajectory τ_g is summarized in Algorithm 1. More details of the implementation are available in Appendix C.

Algorithm 1: OASIS (dataset generation)

Input: Raw dataset $\mathcal{D}$, safety threshold κ
Output: Generated sub-trajectory τ_g
1: Sample a initial state: $s \sim \mathcal{D}$
2: Get initial noisy sub-trajectory:
   $\quad \boldsymbol{x}_k = [s, s_1, ..., s_{L-1}], s_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
3: Determine the condition $\boldsymbol{y}_c$;
4: **for** $k = K, ..., 1$ **do**
5: $\quad$ Calculate $\epsilon(\boldsymbol{x}_k, \boldsymbol{y}_c, k)$ (8);
6: $\quad$ Inverse sampling sequence $\boldsymbol{x}_{k-1}$ (6)
7: **end for**
8: Get actions, rewards, and costs from $\boldsymbol{x}_0$;
9: **Return:** trajectory τ_g

4.4 Theoretical analysis

We first investigate how the distribution mismatch degrades the policy performance on constraint satisfaction. Suppose that the maximum one-step cost is $C_{\max} = \max_{s,a} c(s, a)$. Based on Lemma 6 in [92] and Lemma 2 in [93], the performance gap between the policy π learned with the dataset $\mathcal{D}$ and the optimal policy is bounded by

$$|V_c^\pi(\mu_0) - V_c^*(\mu_0)| \le \frac{2C_{\max}}{1-\gamma} D_{\text{TV}}(d^{\mathcal{D}}(s) \| d^*(s)) + \frac{2C_{\max}}{1-\gamma} \mathbb{E}_{d^*(s)}[D_{\text{TV}}(\pi(a|s) \| \pi^*(a|s))], \quad (9)$$

where $d^{\mathcal{D}}(s), d^*(s)$ denote the stationary state distribution of the dataset and optimal policy. The proof is given in Appendix A.1. Therefore, a significant mismatch between the dataset and the optimal policy results in both a substantial state distribution TV distance and a large policy shift from the optimal one, which can cause notable performance degradation, especially when the offline RL algorithm enforces the learned policy to closely resemble the behavior policy of the offline data.

Then we provide a theoretical analysis of how our method mitigates this mismatch issue by shrinking the distribution gap, which provides a guarantee of the safety performance of the regularized offline safe RL policy. Let $d_g(s|\boldsymbol{y})$ denote the state marginal of the generated data with condition $\boldsymbol{y}$. We first make the following assumptions.

Assumption 1 (Score estimation error of the state marginal). There exists a condition $\boldsymbol{y}^*$ such that the score function error of the state marginal is bounded by

$$\mathbb{E}_{d^*(s)} \|\nabla_s \log d_g(s|\boldsymbol{y}^*) - \nabla_s \log d^*(s)\| \le \varepsilon_{\text{score}}^2, \quad (10)$$

where $d^*(s)$ is the stationary state distribution induced by the optimal policy π^*.

This assumption is also adopted in previous work [94, 95]. For simplicity, we omit the condition $\boldsymbol{y}^*$ in the following analysis and use $d_g(s), d_g(s, a)$ to denote the generated state or the state-action distribution with condition $\boldsymbol{y}^*$. As we use inverse dynamics $f(a|s, s')$ to calculate actions based on the generated state sequence, the quality of the dataset is also determined by the inverse dynamics. Therefore, we further make the following assumption.

Assumption 2 (Error of inverse policy). The error of action distribution generated by the inverse dynamics is bounded by

$$\mathbb{E}_{d^*(s)}[D_{\text{KL}}(\hat{\pi}_{\text{inv}}(\cdot|s) \| \pi^*(\cdot|s))] \le \varepsilon_{\text{inv}}, \quad (11)$$

---

[1]This training/generation formula fixing initial state in the sequence is adopted from Decision Diffuser [80]. However, we also found that fixing the initial state is unnecessary for our dataset curation and conditional distribution shaping task. Keeping this condition or not does not significantly impact the generation quality.

where $\hat{\pi}_{\text{inv}}(a|s) = \mathbb{E}_{s'}[\hat{f}(a|s,s')]$ denotes the empirical inverse policy, which is a marginal of inverse dynamics over s'.

Then the distance of generated data distribution to the optimal one is bounded as:

Theorem 1 (Distribution shaping error bound). *Suppose that the optimal stationary state distribution satisfies that 1) its score function $\nabla_s \log d^*(s)$ is L-Lipschitz and 2) its second momentum is bounded. Under Assumption 1 and 2, the gap of generated state-action distribution to the optimal stationary state-action distribution is bounded by*

$$D_{\text{TV}}\left(d_g(s,a)\|d^*(s,a)\right) \leq \tilde{\mathcal{O}}\left(\varepsilon_{score}\sqrt{K}\right) + \sqrt{\varepsilon_{inv}/2} + C(d^*(s), L, K), \tag{12}$$

where $C(d^*(s), L, K)$ represents a constant determined by $d^*(s)$, L and K.

The proof is given in Appendix A.2. Theorem 1 indicates that using the proposed OASIS method, we can shape the dataset distribution towards a bounded neighborhood of the optimal distribution.

Given the generated data, we will then train a regularized offline safe RL policy by Eq. (3). Notice that the regularization term in the objective function in Eq. (3) is equivalent to an explicit policy constraint, and the coefficient w is the corresponding dual variable. Therefore, we make the following assumption on the distance between the learned policy π_ϕ and the behavior policy.

Assumption 3. Denote the generated dataset as $\mathcal{D}_g$ and the corresponding behavior policy as π_g, given a fixed coefficient w, for the policy π_ϕ optimized by Eq. (3), there exists a ε_{reg} such that

$$\mathbb{E}_{d_g(s)}\left[D_{\text{KL}}\left(\pi_\phi(\cdot|s)\|\pi_g(\cdot|s)\right)\right] \leq \varepsilon_{\text{reg}}. \tag{13}$$

Based on the above assumptions, we can derive the bound of constraint violation of the policy learned on the offline data generated by OASIS. The proof is given in Appendix A.3.

Theorem 2 (Constraint violation bound). *For policy π_ϕ optimized by regularized-based offline safe RL on generated dataset $\mathcal{D}_g$, under Assumption 1, 2 and 3, the constraint violation of the trained policy is bounded as:*

$$V_c^{\pi_\phi}(\mu_0) - \kappa \leq \frac{2C_{max}}{1-\gamma}\left(\tilde{\mathcal{O}}\left(\varepsilon_{score}\sqrt{K}\right) + C(d^*(s), L, K) + \sqrt{\varepsilon_{inv}/2} + \sqrt{\varepsilon_{reg}/2}\right), \tag{14}$$

where $C(d^*(s), L, K)$ represents a constant determined by $d^*(s)$, L and K.

The theoretical analysis sheds insights by answering two questions: (1) Why do we use the diffusion model for conditional distribution shaping, and (2) How does conditional distribution shaping benefit offline safe RL training? Theorem 1 shows that by using the conditional diffusion model as a data generator, the TV distance between optimal and generated state-action distribution is bounded; Theorem 2 shows that the safety performance of agent trained on the generated dataset is guaranteed with OASIS.

5 Experiment

In the experiments, we answer these questions: (1) How does the distribution of the dataset influence the performance of regularized offline safe RL? (2) How does our proposed distribution shaping method perform in offline safe RL tasks? (3) How well does the conditional data generator shape the dataset distribution? To address these questions, we set up the following experiment tasks.

Environments. We adopt the continuous robot locomotion control tasks in the public benchmark `Bullet-Safety-Gym` [96] for evaluation, which is commonly used in previous works [68, 28, 60]. We consider two tasks, `Run` and `Circle`, and three types of robots, `Ball`, `Car`, and `Drone`. We name the environment as `Agent-Task`. A detailed description is available in the Appendix B.

Datasets. Our experiment tasks are mainly built upon the offline safe RL dataset `OSRL` [74]. To better evaluate the tested algorithms with the challenging SDM problem, we create four different training dataset types, `full`, `tempting`, `conservative`, and `hybrid`. The `tempting` dataset contains sparse safe demonstrations, the `conservative` dataset lacks rewarding data points, and the `hybrid` dataset has scarcity in the medium-reward, medium-cost trajectories. We set different cost thresholds for different datasets. A detailed description and visualization of the datasets are available in Appendix B. For the main experiments presented in Table 1, we train on `tempting` dataset, with threshold $\kappa = 20$.
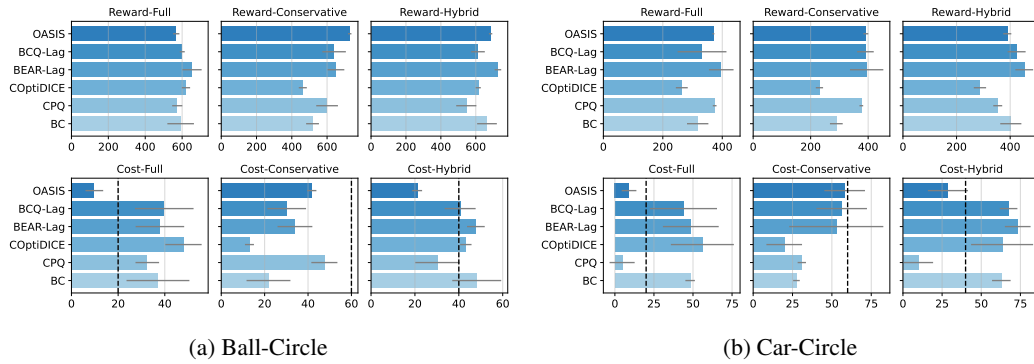
Figure 5: Performance with different datasets and varying constraint thresholds. The visualization of these datasets is available in Appendix B.

Baselines and OASIS. For baselines, we compared our method with both *model-centric* and *data-centric* baseline methods. *Model-cenrtic*: (1) Q-learning-based algorithms: `BCQ-Lag` [35, 86], `BEAR-Lag` [41, 86], and `CPQ` [26]; (2) Imitation learning: Behavior Cloning (`BC`) [26]; (3) Distribution correction estimation: `COptiDICE` [23], and (4) Sequential modeling algorithms: `CDT` [28] and `FISOR` [31]; *Data-cenrtic*: (5) Data augmentation: `CVAE-BCQL`: we train BCQ-Lag agents on the datasets generated by Conditional Variational Autoencoder (CVAE) [97]. The `CVAE` training set and the dataset generation conditions are set as the same with our OASIS. For `OASIS` implementation, we set three cost conditions $\hat{C} \leq \kappa$ to align safety preference and guarantee data coverage. Code is available on our Github repository, checkpoints and curated datasets are available on our HuggingFace repository.

Metrics. We use the normalized cost return and the normalized reward return as the evaluation metric for comparison in Table 1 and 2. The normalized cost is defined as $C_{\text{normalized}} = C_\pi/\kappa$, where C_π is the cost return and κ is the cost threshold. The agent is safe if $C_{\text{normalized}} \leq 1$. The normalized reward is computed by $R_{\text{normalized}} = R_\pi/r_{\max}(\mathcal{M})$, where $r_{\max}(\mathcal{M})$ is the maximum empirical reward return for task $\mathcal{M}$ within the given dataset. We report the averaged results and standard deviations over 3 seeds for all the quantity evaluations.

Table 1: Evaluation results of the normalized reward and cost. The cost threshold is 1. ↑: the higher the reward, the better. ↓: the lower the cost (up to threshold 1), the better. **Bold**: Safe agents whose normalized cost is smaller than 1. Gray: Unsafe agents. **Blue**: Safe agent with the highest reward.

Algorithm	Stats	Tasks					
		BallRun	CarRun	DroneRun	BallCircle	CarCircle	DroneCircle
BC	reward ↑	0.55 ± 0.23	0.94 ± 0.02	0.62 ± 0.11	0.73 ± 0.05	0.59 ± 0.11	0.82 ± 0.01
	cost ↓	2.04 ± 1.32	1.50 ± 1.11	3.48 ± 0.68	2.53 ± 0.15	3.39 ± 0.85	3.29 ± 0.18
CPQ	reward ↑	0.25 ± 0.11	0.63 ± 0.51	0.13 ± 0.30	**0.39 ± 0.34**	**0.64 ± 0.02**	0.01 ± 0.02
	cost ↓	1.34 ± 1.32	1.43 ± 1.82	2.29 ± 1.98	**0.73 ± 0.66**	**0.12 ± 0.19**	3.16 ± 3.85
COptiDICE	reward ↑	0.63 ± 0.04	**0.90 ± 0.03**	0.71 ± 0.01	0.73 ± 0.02	0.52 ± 0.01	**0.35 ± 0.02**
	cost ↓	3.13 ± 0.17	**0.28 ± 0.24**	3.87 ± 0.08	2.83 ± 0.23	3.56 ± 0.16	**0.12 ± 0.10**
BEAR-Lag	reward ↑	0.65± 0.08	0.55 ± 0.62	0.10 ± 0.33	0.89 ± 0.02	0.80 ± 0.08	0.89 ± 0.04
	cost ↓	4.38 ± 0.28	8.44 ± 0.62	3.72 ± 3.22	2.84 ± 0.28	2.89 ± 0.84	4.03 ± 0.51
BCQ-Lag	reward ↑	0.51 ± 0.19	0.96 ± 0.06	0.76 ± 0.07	0.76 ± 0.04	0.79 ± 0.02	0.88 ± 0.04
	cost ↓	1.96 ± 0.88	2.31 ± 3.22	5.19 ± 1.08	2.62 ± 0.29	3.25 ± 0.28	3.90 ± 0.55
CDT	reward ↑	0.35 ± 0.01	**0.96 ± 0.01**	0.84 ± 0.12	0.73 ± 0.01	0.71 ± 0.01	0.17 ± 0.08
	cost ↓	1.56 ± 1.10	**0.67 ± 0.03**	7.56 ± 0.33	1.36 ± 0.03	2.39 ± 0.15	1.08 ± 0.62
FISOR	reward ↑	**0.17 ± 0.03**	**0.85 ± 0.02**	0.44 ± 0.14	**0.28 ± 0.03**	**0.24 ± 0.05**	**0.49 ± 0.05**
	cost ↓	**0.04 ± 0.06**	**0.15 ± 0.20**	2.52 ± 0.61	**0.00 ± 0.00**	**0.15 ± 0.27**	**0.02 ± 0.03**
CVAE-BCQL	reward ↑	0.25 ± 0.02	**0.88 ± 0.00**	0.21 ± 52.07	0.49 ± 0.03	0.60 ± 0.05	0.01 ± 0.02
	cost ↓	1.40 ± 0.35	**0.00 ± 0.00**	2.80 ± 0.63	1.39 ± 0.27	1.77 ± 0.47	3.31 ± 1.66
OASIS (ours)	reward ↑	**0.28 ± 0.01**	0.85 ± 0.04	**0.13 ± 0.08**	**0.70 ± 0.01**	**0.76 ± 0.03**	**0.60 ± 0.01**
	cost ↓	**0.79 ± 0.37**	0.02 ± 0.03	**0.79 ± 0.54**	**0.45 ± 0.14**	**0.89 ± 0.59**	**0.25 ± 0.10**

5.1 How can conditional data generation benefit offline safe RL?

Performance degradation with SDM problems. The comparison results on the `tempting` dataset are presented in Table 1 with the cost threshold $\kappa = 20$ before normalization. Results of `BC` show the mismatch between the behavior policy and the safe policy, as the cost returns significantly violate the safety constraints. The results of `BCQ-Lag` and `BEAR-Lag` show this mismatch further influences the regularized-based algorithms, leading to constraint violations. This is because the regularization term pushes the policy towards the unsafe behavior policy. The conservative Q function estimation method `CPQ`, exhibits a significant reward degradation in all tasks, which arises from the drawback of the pessimistic estimation methods that learn over-conservative behaviors. `COptiDICE` also fails to learn safe and rewarding policies, showing that even using distribution correction estimation is not enough to solve the SDM problem. For sequential modeling algorithms, `CDT` shows poor safety performance and `FISOR` tends to be over-conservative with poor reward performance. This is because both methods require a large amount of high-quality data while the trajectories with low cost and high reward are sparse in this task. The unsatisfactory performance of these *model-centric* algorithms further motivates the effective *data-centric* learning algorithm for offline safe RL.

Performance improvement using OASIS. From Table 1, we find that only our method OASIS can learn safe and rewarding policies by mitigating the SDM problem. In addition to the results on the `tempting` dataset, we also provide evaluation results within different types of datasets and constraint thresholds in Fig. 5a and Fig. 5b. We can observe that most baselines still fail to learn a safe policy within different task conditions due to the SDM issue. In contrast, our proposed OASIS method achieves the highest reward among all safe agents, which shows strength in more general cases.

Figure 6: Data efficiency on the `Ball-Circle` task with a `tempting` dataset.

High data efficiency of OASIS. In this experiment, we vary the amount of data for offline RL agent training to evaluate data efficiency. The evaluation results are shown in Fig. 6. The x-axis α represents the size of the RL agent training dataset. For OASIS, it denotes the size of the generated data. A subsequent BCQ-Lag agent is trained on this generated dataset to obtain the safe RL policy. For the baseline methods, we randomly sample trajectories from the original dataset to construct the training dataset.

The comparison results indicate that we can still learn a good policy using a small amount of high-quality data ($\alpha < 2\%$) generated by OASIS. In contrast, baseline methods show significant performance degradation when the data are sparse as the noisy data is of low quality.

This observation demonstrates that the agent can learn a good policy with high data efficiency given high-quality data with minimal safe dataset mismatch (SDM) issues. As OASIS offers a solution to shape the dataset distribution, it also reduces the required training dataset size while maintaining good performance, further

Table 2: Ablation study on denoising step K

K		10	20	40
Ball-Circle	reward	0.71 ± 0.02	0.70 ± 0.01	0.71± 0.01
	cost	0.72 ± 0.10	0.45 ± 0.14	0.99 ± 0.13
Ball-Run	reward	0.29 ± 0.04	0.28 ± 0.01	0.29 ± 0.01
	cost	0.16 ± 0.14	0.79 ± 0.37	0.00 ± 0.00

reaffirming the effectiveness of *data-centric* approaches in offline safe learning and highlighting that prioritizing quality is essential [70].

5.2 How can OASIS shape the dataset distribution?

Successful distribution shaping. To show the distribution shaping capability of the proposed OASIS and baseline `CVAE`, we generate the dataset under different conditions and visualize them in Fig. 7. When using different conditions, the expectations of reward and cost of the generated dataset change accordingly. This shows the strong capability of our method in distribution shaping. We also visualize the density of the generated data. In the `Car-circle` task, the robot receives high rewards when moving along the circle boundary and receives costs when it exceeds the boundaries on both sides, as shown in Fig. 7(c). The original dataset contains trajectories with various safety performances.
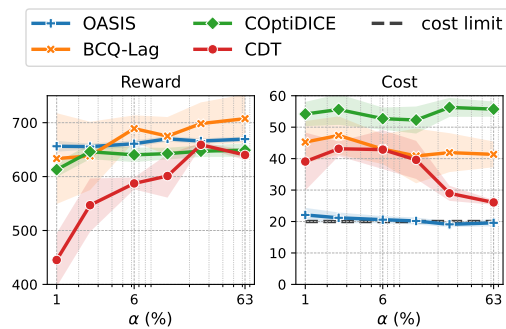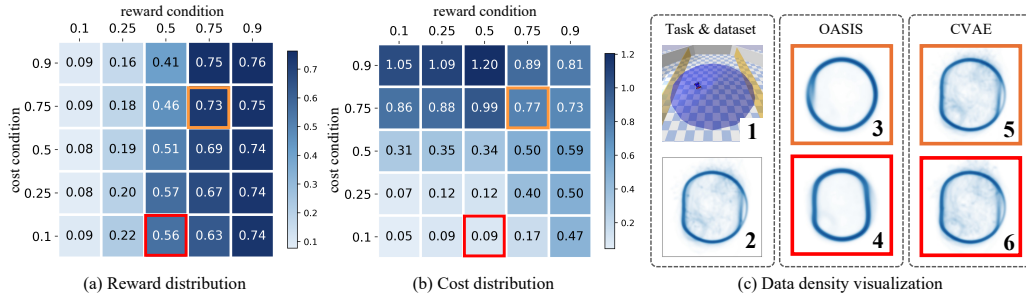
Figure 7: (a)(b) Reward and cost performance of the generated data: $\mathbb{E}\left[r(s, a)\right], \mathbb{E}\left[c(s, a)\right], (s, a) \sim d_g$. The x-axis and y-axis mean the reward and cost conditions and the values of both conditions and expectations are normalized with the same scale. (c) Visualization of the data density. 1: `Car-circle` task; 2: The density of the (x, y) position of the raw dataset; 3, 4: The density of the position (x, y) of the data generated by OASIS under conditions $[0.1, 0.5]$ and $[0.75, 0.75]$; 5, 6: the density of the position (x, y) of the data generated by `CVAE` under conditions $[0.1, 0.5]$ and $[0.75, 0.75]$

When using a low-cost condition, the generated data are clustered within the safety boundary to satisfy the constraints. When using a high-reward condition, the generated data points are closer to the circle boundary and receive higher rewards. In contrast, the baseline `CVAE` cannot successfully incorporate the conditions in data generation, resulting in almost similar datasets with different conditions as shown in Fig. 7(c). More experiment results including the visualization of generated dataset comparison and corresponding analysis are available in Appendix B.2.

5.3 Robust performance against denoising steps

We conduct an ablation study on the key hyperparameter of the proposed OASIS method. The experiment related to the denoising steps K is presented in Table 2. Performance does not change much with different values, which shows the robustness of the proposed OASIS method.

6 Conclusion

In this paper, we study the challenging problem in offline safe RL: the safe data mismatch between the imperfect demonstration and the target performance requirements. To address this issue, we proposed the OASIS method to employ a conditional diffusion model to shape the dataset distribution and benefit offline safe RL training. In addition to the theoretical guarantee of performance improvement, we also conduct extensive experiments to show the superior performance of OASIS in learning a safe and rewarding policy on many challenging offline safe RL tasks. More importantly, our method shows good data efficiency and robustness to hyperparameters, which makes it preferable for applications in many real-world tasks.

There are two limitations of OASIS: (1) Offline training takes longer: our method involves preprocessing the offline dataset to enhance quality, which requires more time and computing resources; (2) Achieving zero-constraint violations remains challenging with imperfect demonstrations. One potential negative social impact is that misuse of our method may cause harmful consequences and safety issues. Nevertheless, we believe that our proposed method can inspire further *data-centric* research in the safe learning community and help to adapt offline RL algorithms to real-world tasks with safety requirements.

Acknowledgment

References

[1] Rafael Figueiredo Prudencio, Marcos ROA Maximo, and Esther Luna Colombini. A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[2] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

[3] Kyle Stachowicz and Sergey Levine. Racer: Epistemic risk-sensitive rl enables fast driving with fewer crashes. *arXiv preprint arXiv:2405.04714*, 2024.

[4] Haohong Lin, Wenhao Ding, Zuxin Liu, Yaru Niu, Jiacheng Zhu, Yuming Niu, and Ding Zhao. Safety-aware causal representation for trustworthy offline reinforcement learning in autonomous driving. *IEEE Robotics and Automation Letters*, 2024.

[5] Zhili Zhang, Songyang Han, Jiangwei Wang, and Fei Miao. Spatial-temporal-aware safe multi-agent reinforcement learning of connected autonomous vehicles in challenging scenarios. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5574–5580. IEEE, 2023.

[6] Xing Fang, Qichao Zhang, Yinfeng Gao, and Dongbin Zhao. Offline reinforcement learning for autonomous driving with real world driving data. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 3417–3422. IEEE, 2022.

[7] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.

[8] Wenhao Ding, Laixi Shi, Yuejie Chi, and Ding Zhao. Seeing is not believing: Robust reinforcement learning against spurious correlation. *Advances in Neural Information Processing Systems*, 36, 2024.

[9] Weiye Zhao, Rui Chen, Yifan Sun, Ruixuan Liu, Tianhao Wei, and Changliu Liu. Guard: A safe reinforcement learning benchmark. *arXiv preprint arXiv:2305.13681*, 2023.

[10] Jinning Li, Xinyi Liu, Banghua Zhu, Jiantao Jiao, Masayoshi Tomizuka, Chen Tang, and Wei Zhan. Guided online distillation: Promoting safe reinforcement learning by offline demonstration. *arXiv preprint arXiv:2309.09408*, 2023.

[11] Wenhao Ding, Tong Che, Ding Zhao, and Marco Pavone. Bayesian reparameterization of reward-conditioned reinforcement learning with energy-based models. In *International Conference on Machine Learning*, pages 8053–8066. PMLR, 2023.

[12] Dohyeong Kim, Yunho Kim, Kyungjae Lee, and Songhwai Oh. Safety guided policy optimization. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2462–2467. IEEE, 2022.

[13] Tairan He, Chong Zhang, Wenli Xiao, Guanqi He, Changliu Liu, and Guanya Shi. Agile but safe: Learning collision-free high-speed legged locomotion. *arXiv preprint arXiv:2401.17583*, 2024.

[14] Wenli Xiao, Tairan He, John Dolan, and Guanya Shi. Safe deep policy adaptation. *arXiv preprint arXiv:2310.08602*, 2023.

[15] Weidong Huang, Jiaming Ji, Borong Zhang, Chunhe Xia, and Yaodong Yang. Safedreamer: Safe reinforcement learning with world models. In *The Twelfth International Conference on Learning Representations*, 2023.

[16] Zhaorun Chen, Binhao Chen, Tairan He, Liang Gong, and Chengliang Liu. Progressive adaptive chance-constrained safeguards for reinforcement learning. *arXiv preprint arXiv:2310.03379*, 2023.

[17] Wenjun Zou, Yao Lv, Jie Li, Yujie Yang, Shengbo Eben Li, Jingliang Duan, Xianyuan Zhan, Jingjing Liu, Yaqin Zhang, and Keqiang Li. Policy bifurcation in safe reinforcement learning. *arXiv preprint arXiv:2403.12847*, 2024.

[18] Yujie Yang, Yuxuan Jiang, Yichen Liu, Jianyu Chen, and Shengbo Eben Li. Model-free safe reinforcement learning through neural barrier certificate. *IEEE Robotics and Automation Letters*, 8(3):1295–1302, 2023.

[19] Qian Lin, Zongkai Liu, Danying Mo, and Chao Yu. An offline adaptation framework for constrained multi-objective reinforcement learning. *arXiv preprint arXiv:2409.09958*, 2024.

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

[21] Lukas Brunke, Melissa Greeff, Adam W Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5, 2021.

[22] Akifumi Wachi, Xun Shen, and Yanan Sui. A survey of constraint formulations in safe reinforcement learning. *arXiv preprint arXiv:2402.02025*, 2024.

[23] Jongmin Lee, Cosmin Paduraru, Daniel J Mankowitz, Nicolas Heess, Doina Precup, Kee-Eung Kim, and Arthur Guez. Coptidice: Offline constrained reinforcement learning via stationary distribution correction estimation. *arXiv preprint arXiv:2204.08957*, 2022.

[24] Jongmin Lee, Wonseok Jeon, Byungjun Lee, Joelle Pineau, and Kee-Eung Kim. Optidice: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning*, pages 6120–6130. PMLR, 2021.

[25] Ilya Kostrikov, Rob Fergus, Jonathan Tompson, and Ofir Nachum. Offline reinforcement learning with fisher divergence critic regularization. In *International Conference on Machine Learning*, pages 5774–5783. PMLR, 2021.

[26] Haoran Xu, Xianyuan Zhan, and Xiangyu Zhu. Constraints penalized q-learning for safe offline reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8753–8760, 2022.

[27] Hoang Le, Cameron Voloshin, and Yisong Yue. Batch policy learning under constraints. In *International Conference on Machine Learning*, pages 3703–3712. PMLR, 2019.

[28] Zuxin Liu, Zijian Guo, Yihang Yao, Zhepeng Cen, Wenhao Yu, Tingnan Zhang, and Ding Zhao. Constrained decision transformer for offline safe reinforcement learning. *arXiv preprint arXiv:2302.07351*, 2023.

[29] Qin Zhang, Linrui Zhang, Haoran Xu, Li Shen, Bowen Wang, Yongzhe Chang, Xueqian Wang, Bo Yuan, and Dacheng Tao. Saformer: A conditional sequence modeling approach to offline safe reinforcement learning. *arXiv preprint arXiv:2301.12203*, 2023.

[30] Qian Lin, Bo Tang, Zifan Wu, Chao Yu, Shangqin Mao, Qianlong Xie, Xingxing Wang, and Dong Wang. Safe offline reinforcement learning with real-time budget constraints. *arXiv preprint arXiv:2306.00603*, 2023.

[31] Yinan Zheng, Jianxiong Li, Dongjie Yu, Yujie Yang, Shengbo Eben Li, Xianyuan Zhan, and Jingjing Liu. Safe offline reinforcement learning with feasibility-guided diffusion model. *arXiv preprint arXiv:2401.10700*, 2024.

[32] Rasool Fakoor, Jonas W Mueller, Kavosh Asadi, Pratik Chaudhari, and Alexander J Smola. Continuous doubly constrained batch reinforcement learning. *Advances in Neural Information Processing Systems*, 34:11260–11273, 2021.

[33] Xianyuan Zhan, Haoran Xu, Yue Zhang, Xiangyu Zhu, Honglei Yin, and Yu Zheng. Deepthermal: Combustion optimization for thermal power generating units using offline reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4680–4688, 2022.

[34] Yao Liu, Pratik Chaudhari, and Rasool Fakoor. Budgeting counterfactual for offline rl. *Advances in Neural Information Processing Systems*, 36, 2024.

[35] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*, pages 2052–2062. PMLR, 2019.

[36] Laixi Shi, Gen Li, Yuting Wei, Yuxin Chen, and Yuejie Chi. Pessimistic q-learning for offline reinforcement learning: Towards optimal sample complexity. In *International conference on machine learning*, pages 19967–20025. PMLR, 2022.

[37] Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. Provably good batch off-policy reinforcement learning without great exploration. *Advances in neural information processing systems*, 33:1264–1274, 2020.

[38] Yihang Yao, Zuxin Liu, Zhepeng Cen, Peide Huang, Tingnan Zhang, Wenhao Yu, and Ding Zhao. Gradient shaping for multi-constraint safe reinforcement learning. *arXiv preprint arXiv:2312.15127*, 2023.

[39] Zhang-Wei Hong, Aviral Kumar, Sathwik Karnik, Abhishek Bhandwaldar, Akash Srivastava, Joni Pajarinen, Romain Laroche, Abhishek Gupta, and Pulkit Agrawal. Beyond uniform sampling: Offline reinforcement learning with imbalanced datasets. *Advances in Neural Information Processing Systems*, 36:4985–5009, 2023.

[40] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33:1179–1191, 2020.

[41] Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in neural information processing systems*, 32, 2019.

[42] Samarth Sinha, Ajay Mandlekar, and Animesh Garg. S4rl: Surprisingly simple self-supervision for offline reinforcement learning in robotics. In *Conference on Robot Learning*, pages 907–917. PMLR, 2022.

[43] Tianhe Yu, Aviral Kumar, Yevgen Chebotar, Karol Hausman, Chelsea Finn, and Sergey Levine. How to leverage unlabeled data in offline reinforcement learning. In *International Conference on Machine Learning*, pages 25611–25635. PMLR, 2022.

[44] Anqi Li, Dipendra Misra, Andrey Kolobov, and Ching-An Cheng. Survival instinct in offline reinforcement learning. *Advances in neural information processing systems*, 36, 2024.

[45] Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. *Advances in Neural Information Processing Systems*, 33:3976–3990, 2020.

[46] Ahmed Aloui, Juncheng Dong, Cat P Le, and Vahid Tarokh. Counterfactual data augmentation with contrastive learning. *arXiv preprint arXiv:2311.03630*, 2023.

[47] Chaochao Lu, Biwei Huang, Ke Wang, José Miguel Hernández-Lobato, Kun Zhang, and Bernhard Schölkopf. Sample-efficient reinforcement learning via counterfactual-based data augmentation. *arXiv preprint arXiv:2012.09092*, 2020.

[48] Silviu Pitis, Elliot Creager, Ajay Mandlekar, and Animesh Garg. Mocoda: Model-based counterfactual data augmentation. *Advances in Neural Information Processing Systems*, 35:18143–18156, 2022.

[49] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International Conference on Machine Learning*, pages 22–31. PMLR, 2017.

[50] Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, Yaodong Yang, and Alois Knoll. A review of safe reinforcement learning: Methods, theory and applications. *arXiv preprint arXiv:2205.10330*, 2022.

[51] Zuxin Liu, Zijian Guo, Zhepeng Cen, Huan Zhang, Jie Tan, Bo Li, and Ding Zhao. On the robustness of safe reinforcement learning under observational perturbations. *arXiv preprint arXiv:2205.14691*, 2022.

[52] Dohyeong Kim, Kyungjae Lee, and Songhwai Oh. Trust region-based safe distributional reinforcement learning for multiple constraints. *Advances in neural information processing systems*, 36, 2024.

[53] Sheng Xu and Guiliang Liu. Uncertainty-aware constraint inference in inverse constrained reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2023.

[54] Haonan Yu, Wei Xu, and Haichao Zhang. Towards safe reinforcement learning with a safety editor policy. *arXiv preprint arXiv:2201.12427*, 2022.

[55] Dohyeong Kim, Mineui Hong, Jeongho Park, and Songhwai Oh. Scale-invariant gradient aggregation for constrained multi-objective reinforcement learning. *arXiv preprint arXiv:2403.00282*, 2024.

[56] Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson, and Marco Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, 18(167):1–51, 2018.

[57] Chen Tessler, Daniel J Mankowitz, and Shie Mannor. Reward constrained policy optimization. *arXiv preprint arXiv:1805.11074*, 2018.

[58] Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 7, 2019.

[59] Dongsheng Ding, Kaiqing Zhang, Tamer Basar, and Mihailo Jovanovic. Natural policy gradient primal-dual method for constrained markov decision processes. *Advances in Neural Information Processing Systems*, 33:8378–8390, 2020.

[60] Yiming Zhang, Quan Vuong, and Keith Ross. First order constrained optimization in policy space. *Advances in Neural Information Processing Systems*, 2020.

[61] Zhepeng Cen, Yihang Yao, Zuxin Liu, and Ding Zhao. Feasibility consistent representation learning for safe reinforcement learning. *arXiv preprint arXiv:2405.11718*, 2024.

[62] Zifan Wu, Bo Tang, Qian Lin, Chao Yu, Shangqin Mao, Qianlong Xie, Xingxing Wang, and Dong Wang. Off-policy primal-dual safe reinforcement learning. *arXiv preprint arXiv:2401.14758*, 2024.

[63] Dongsheng Ding, Zhengyan Huan, and Alejandro Ribeiro. Resilient constrained reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 3412–3420. PMLR, 2024.

[64] Jiayi Guan, Li Shen, Ao Zhou, Lusong Li, Han Hu, Xiaodong He, Guang Chen, and Changjun Jiang. Poce: Primal policy optimization with conservative estimation for multi-constraint offline reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26243–26253, 2024.

[65] Nicholas Polosky, Bruno C Da Silva, Madalina Fiterau, and Jithin Jagannath. Constrained offline policy optimization. In *International Conference on Machine Learning*, pages 17801–17810. PMLR, 2022.

[66] Kihyuk Hong, Yuhang Li, and Ambuj Tewari. A primal-dual-critic algorithm for offline constrained reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 280–288. PMLR, 2024.

[67] Jiayi Guan, Guang Chen, Jiaming Ji, Long Yang, Zhijun Li, et al. Voce: Variational optimization with conservative estimation for offline safe reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[68] Zijian Guo, Weichao Zhou, and Wenchao Li. Temporal logic specification-conditioned decision transformer for offline safe reinforcement learning. *arXiv preprint arXiv:2402.17217*, 2024.

[69] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *arXiv preprint arXiv:2106.01345*, 2021.

[70] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

[71] Zuxin Liu, Thai Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, Yihao Feng, et al. Apigen: Automated pipeline for generating verifiable and diverse function-calling datasets. *arXiv preprint arXiv:2406.18518*, 2024.

[72] Suneel Belkhale, Yuchen Cui, and Dorsa Sadigh. Data quality in imitation learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[73] Yiming Huang, Xiao Liu, Yeyun Gong, Zhibin Gou, Yelong Shen, Nan Duan, and Weizhu Chen. Key-point-driven data synthesis with its enhancement on mathematical reasoning. *arXiv preprint arXiv:2403.02333*, 2024.

[74] Zuxin Liu, Zijian Guo, Haohong Lin, Yihang Yao, Jiacheng Zhu, Zhepeng Cen, Hanjiang Hu, Wenhao Yu, Tingnan Zhang, Jie Tan, et al. Datasets and benchmarks for offline safe reinforcement learning. *arXiv preprint arXiv:2306.09303*, 2023.

[75] Zhengbang Zhu, Hanye Zhao, Haoran He, Yichao Zhong, Shenyu Zhang, Yong Yu, and Weinan Zhang. Diffusion models for reinforcement learning: A survey. *arXiv preprint arXiv:2311.01223*, 2023.

[76] Jaewoo Lee, Sujin Yun, Taeyoung Yun, and Jinkyoo Park. Gta: Generative trajectory augmentation with guidance for offline reinforcement learning. *arXiv preprint arXiv:2405.16907*, 2024.

[77] Siyuan Zhou, Yilun Du, Shun Zhang, Mengdi Xu, Yikang Shen, Wei Xiao, Dit-Yan Yeung, and Chuang Gan. Adaptive online replanning with diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

[78] Hui Yuan, Kaixuan Huang, Chengzhuo Ni, Minshuo Chen, and Mengdi Wang. Reward-directed conditional diffusion: Provable distribution estimation and reward improvement. *Advances in Neural Information Processing Systems*, 36, 2024.

[79] Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

[80] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua Tenenbaum, Tommi Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision-making? *arXiv preprint arXiv:2211.15657*, 2022.

[81] Wei Xiao, Tsun-Hsuan Wang, Chuang Gan, and Daniela Rus. Safediffuser: Safe planning with diffusion probabilistic models. *arXiv preprint arXiv:2306.00148*, 2023.

[82] Zhixuan Liang, Yao Mu, Mingyu Ding, Fei Ni, Masayoshi Tomizuka, and Ping Luo. Adaptdiffuser: Diffusion models as adaptive self-evolving planners. *arXiv preprint arXiv:2302.01877*, 2023.

[83] Cong Lu, Philip Ball, Yee Whye Teh, and Jack Parker-Holder. Synthetic experience replay. *Advances in Neural Information Processing Systems*, 36, 2024.

[84] Haoran He, Chenjia Bai, Kang Xu, Zhuoran Yang, Weinan Zhang, Dong Wang, Bin Zhao, and Xuelong Li. Diffusion model is an effective planner and data synthesizer for multi-task reinforcement learning. *Advances in neural information processing systems*, 36, 2024.

[85] Eitan Altman. Constrained markov decision processes with total cost criteria: Lagrangian approach and dual linear program. *Mathematical methods of operations research*, 48(3):387–417, 1998.

[86] Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, pages 9133–9143. PMLR, 2020.

[87] Qian Lin, Chao Yu, Zongkai Liu, and Zifan Wu. Policy-regularized offline multi-objective reinforcement learning. *arXiv preprint arXiv:2401.02244*, 2024.

[88] Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. *Advances in neural information processing systems*, 34:20132–20145, 2021.

[89] Zhang-Wei Hong, Pulkit Agrawal, Rémi Tachet des Combes, and Romain Laroche. Harnessing mixed offline reinforcement learning datasets via trajectory weighting. *arXiv preprint arXiv:2306.13085*, 2023.

[90] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[91] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[92] Tian Xu, Ziniu Li, and Yang Yu. Error bounds of imitating policies and environments. *Advances in Neural Information Processing Systems*, 33:15737–15749, 2020.

[93] Zhepeng Cen, Zuxin Liu, Zitong Wang, Yihang Yao, Henry Lam, and Ding Zhao. Learning from sparse offline datasets via conservative density estimation. *arXiv preprint arXiv:2401.08819*, 2024.

[94] Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence for score-based generative modeling with polynomial complexity. *Advances in Neural Information Processing Systems*, 35:22870–22882, 2022.

[95] Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru R Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. *arXiv preprint arXiv:2209.11215*, 2022.

[96] Sven Gronauer. Bullet-safety-gym: Aframework for constrained reinforcement learning. 2022.

[97] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27, 2014.

[98] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[99] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[100] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

Appendix

Table of Contents

A Proofs

A.1 Proof of Eq.(9)

By definition of the stationary state-action distribution,

$$|V_c^\pi(\mu_0) - V_c^*(\mu_0)| \tag{15}$$

$$= \frac{1}{1-\gamma} \left| \mathbb{E}_{(s,a)\sim d^\pi}[c(s,a)] - \mathbb{E}_{(s,a)\sim d^*}[c(s,a)] \right| \tag{16}$$

$$\leq \frac{C_{\max}}{1-\gamma} \sum_{s,a} |d^\pi(s,a) - d^*(s,a)| \tag{17}$$

$$= \frac{2C_{\max}}{1-\gamma} D_{\mathrm{TV}}(d^\pi(s,a)\|d^*(s,a)) \tag{18}$$

$$\leq \frac{2C_{\max}}{1-\gamma} \left(D_{\mathrm{TV}}(d^\pi(s,a)\|d^*(s)\pi(a|s)) + D_{\mathrm{TV}}(d^*(s)\pi(a|s)\|d^*(s,a)) \right) \tag{19}$$

$$= \frac{2C_{\max}}{1-\gamma} D_{\mathrm{TV}}(d^\pi(s)\|d^*(s)) + \frac{2C_{\max}}{1-\gamma} \mathbb{E}_{d^*(s)}[D_{\mathrm{TV}}(\pi(a|s)\|\pi^*(a|s))] \tag{20}$$

The second inequality holds by triangle inequality for total variation distance.

In general, the stationary distribution of learned policy is in between of the empirical distribution of offline data $d^{\mathcal{D}}$ and optimal d^*. Therefore, we can obtain

$$|V_c^\pi(\mu_0) - V_c^*(\mu_0)| \leq \frac{2C_{\max}}{1-\gamma} D_{\mathrm{TV}}(d^*(s)\|d^{\mathcal{D}}(s)) + \frac{2C_{\max}}{1-\gamma} \mathbb{E}_{d^*(s)}[D_{\mathrm{TV}}(\pi(a|s)\|\pi^*(a|s))]. \tag{21}$$

A.2 Proof of Theorem 1

Proof. By triangle inequality, we first decompose the TV distance between state-action distributions into a state distribution distance and a policy distance,

$$D_{\mathrm{TV}}\left(d_g(s,a)\|d^*(s,a)\right) \tag{22}$$

$$= D_{\mathrm{TV}}\left(d_g(s)\pi_g(a|s)\|d^*(s)\pi^*(a|s)\right) \tag{23}$$

$$\leq D_{\mathrm{TV}}\left(d_g(s)\pi_g(a|s)\|d^*(s)\pi_g(a|s)\right) + D_{\mathrm{TV}}\left(d^*(s)\pi_g(a|s)\|d^*(s)\pi^*(a|s)\right) \tag{24}$$

$$= D_{\mathrm{TV}}\left(d_g(s)\|d^*(s)\right) + \mathbb{E}_{d^*(s)}[D_{\mathrm{TV}}(\pi_g(a|s)\|\pi^*(a|s))] \tag{25}$$

We then consider two parts separately.

For the stationary state distribution distance, we suppose that the optimal distribution $d^*(s)$ has a L-Lipschitz smooth score function and a bounded second momentum. Meanwhile, note that the score function in Assumption 1 is closely related to the denoising model ϵ_θ [98, 99]:

$$\nabla_s \log d_g(s|\boldsymbol{y}) = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}}\epsilon_\theta(s|\boldsymbol{y}), \tag{26}$$

where $\epsilon_\theta(s|\boldsymbol{y})$ is the state marginal of the practical denoising model in Eq. (8). Therefore, by theorem 2 in [95], under Assumption 1, we have

$$D_{\mathrm{TV}}(d_g(s)\|d^*(s)) \lesssim \sqrt{D_{\mathrm{KL}}(d^*(s)\|\mathcal{N}(\mathbf{0}, \mathbf{I}^{|\mathcal{S}|}))}\exp(-K)+L(\sqrt{|\mathcal{S}|}+\mathbf{m}_2)\sqrt{K}+\varepsilon_{\mathrm{score}}\sqrt{K} \tag{27}$$

where K is the number of denoising timestep, $|\mathcal{S}|$ is the dimension of the state space, and $\mathbf{m}_2$ is the second momentum of $d^*(s)$. Therefore, aggregating the first two terms in RHS, we have

$$D_{\mathrm{TV}}(d_g(s)\|d^*(s)) \leq \tilde{\mathcal{O}}\left(\varepsilon_{\mathrm{score}}\sqrt{K}\right) + C(d^*(s), L, K), \tag{28}$$

where $C(\dots)$ is a constant w.r.t d^*, L, K.

Regarding the policy distance. By Pinsker's inequality,

$$D_{\mathrm{TV}}(\pi_g(a|s)\|\pi^*(a|s)) \leq \sqrt{\frac{1}{2}D_{\mathrm{KL}}(\pi_g(a|s)\|\pi^*(a|s))} \tag{29}$$

Meanwhile, since the action is generated by the inverse policy, i.e., $\pi_g = \hat{\pi}_{\mathrm{inv}}$, by Assumption 2, we have

$$\mathbb{E}_{d^*(s)}\left[D_{\mathrm{TV}}(\pi_g(a|s)\|\pi^*(a|s))\right] \leq \mathbb{E}_{d^*(s)}\left[\sqrt{\frac{1}{2}D_{\mathrm{KL}}(\pi_g(a|s)\|\pi^*(a|s))}\right] \tag{30}$$

$$\leq \sqrt{\frac{1}{2}\mathbb{E}_{d^*(s)}\left[D_{\mathrm{KL}}(\pi_g(a|s)\|\pi^*(a|s))\right]} \tag{31}$$

$$= \sqrt{\varepsilon_{\mathrm{inv}}/2} \tag{32}$$

where the second inequality holds by Jensen's inequality.

Combining the Eq. (28)) and (32), we finish the proof of Theorem 1. $\qquad\square$

A.3 Proof of Theorem 2

We start from the Eq. (9). The policy distance can be further decomposed into

$$D_{\mathrm{TV}}(\pi(a|s)\|\pi^*(a|s)) \leq D_{\mathrm{TV}}(\pi(a|s)\|\pi^{\mathcal{D}}(a|s)) + D_{\mathrm{TV}}(\pi^{\mathcal{D}}(a|s)\|\pi^*(a|s)). \tag{33}$$

By Assumption 2 and 3 and Jensen's inequality, we have

$$\mathbb{E}_{d^*(s)}[D_{\mathrm{TV}}(\pi(a|s)\|\pi^*(a|s))] \tag{34}$$

$$\leq \mathbb{E}_{d^*(s)}[D_{\mathrm{TV}}(\pi(a|s)\|\pi^{\mathcal{D}}(a|s))] + \mathbb{E}_{d^*(s)}[D_{\mathrm{TV}}(\pi^{\mathcal{D}}(a|s)\|\pi^*(a|s))] \tag{35}$$

$$\leq \mathbb{E}_{d^*(s)}\left[\sqrt{D_{\mathrm{KL}}(\pi(a|s)\|\pi^{\mathcal{D}}(a|s))/2}\right] + \mathbb{E}_{d^*(s)}\left[\sqrt{D_{\mathrm{KL}}(\pi^{\mathcal{D}}(a|s)\|\pi^*(a|s))/2}\right] \tag{36}$$

$$\leq \sqrt{\mathbb{E}_{d^*(s)}\left[D_{\mathrm{KL}}(\pi(a|s)\|\pi^{\mathcal{D}}(a|s))\right]/2} + \sqrt{\mathbb{E}_{d^*(s)}\left[D_{\mathrm{KL}}(\pi^{\mathcal{D}}(a|s)\|\pi^*(a|s))\right]/2} \tag{37}$$

$$\leq \sqrt{\varepsilon_{\mathrm{reg}}/2} + \sqrt{\varepsilon_{\mathrm{inv}}/2} \tag{38}$$

Plug-in Eq. (28) and (38) into Eq. (21), we have

$$|V_c^\pi(\mu_0) - V_c^*(\mu_0)| \leq \frac{2C_{\max}}{1 - \gamma}\left(\tilde{\mathcal{O}}\left(\varepsilon_{\mathrm{score}}\sqrt{K}\right) + C(d^*(s), L, K) + \sqrt{\varepsilon_{\mathrm{inv}}/2} + \sqrt{\varepsilon_{\mathrm{reg}}/2}\right). \tag{39}$$

Meanwhile, notice that the optimal policy is constraint satisfactory, i.e.,

$$V_c^*(\mu_0) \leq \kappa. \tag{40}$$

Therefore, we have

$$V_c^\pi(\mu_0) - \kappa \leq \frac{2C_{\max}}{1 - \gamma}\left(\tilde{\mathcal{O}}\left(\varepsilon_{\mathrm{score}}\sqrt{K}\right) + C(d^*(s), L, K) + \sqrt{\varepsilon_{\mathrm{inv}}/2} + \sqrt{\varepsilon_{\mathrm{reg}}/2}\right), \tag{41}$$

which finishes the proof of Theorem 2.

B   Supplementary experiments

B.1   Trajectory reweighting for distribution shaping

In this section, we provide details about the trajectory reweighting experiment presented in Sec. 4.1. Following previous work [39, 89] in offline RL, we adopted datapoint reweighting in policy optimization, which can be formulated via importance sampling as:

$$\mathcal{J}^w_{\text{off}}(\pi, \lambda) \approx \mathbb{E}_{(s,a)\sim\mathcal{D}_w}[\mathcal{J}_{\text{off}}(\pi, \lambda)] = \mathbb{E}_{(s,a)\sim\mathcal{D}}[w(s, a)\mathcal{J}_{\text{off}}(\pi, \lambda)], \tag{42}$$

where $\mathcal{J}^w_{\text{off}}(\pi, \lambda)$ is the objective function after reweighting. In this experiment, we utilize a Boltzmann energy function as adopted in [39] for offline RL tasks:

$$w(\tau) \propto \exp\left(\alpha_1 R(\tau)^2 + \alpha_2(C(\tau) - \kappa)^2\right) \tag{43}$$

Here $w(\tau)$ means that all the state-action pairs in one trajectory share the same weight, which is related to the cost and reward returns $C(\tau), R(\tau)$. We adopt $\alpha_1 = \alpha_2 = 1$ in the experiments shown in Fig. 3.

B.2   Supplementary data generation comparison results

Due to the page limit, we omit the visualization of reward and cost distribution using the CVAE method for data generation. Here we provide the results in Fig. 8. From the reward performance and the cost performance, we can observe that CVAE can hardly encode conditions into the data reconstruction, leading to similar results when setting different conditions. From the trajectory reconstruction results shown in Fig. 8(c), we can observe that the generated trajectories are almost the same as the original one. This feature is not desirable for our distribution shaping purpose. In contrast, our method OASIS can successfully shape the distribution as shown in Fig. 7, with the strong capability of the diffusion model in the condition-guided denoising process.
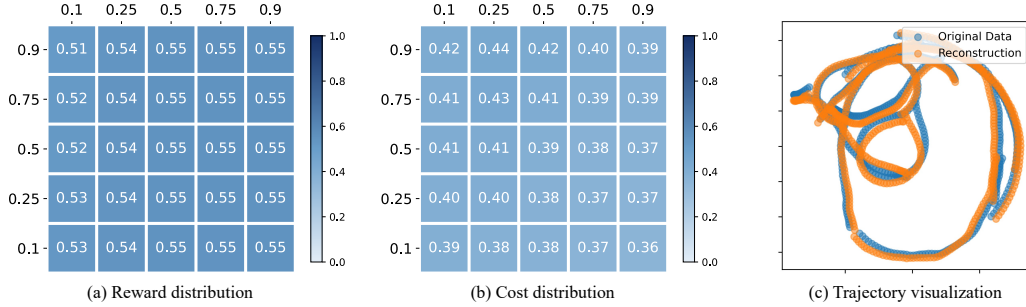

Figure 8: CVAE reconstruction. (a) Reward performance of the generated data: $\mathbb{E}\left[r(s,a)\right], (s,a) \sim d_g$, (b) Cost performance of the generated data: $\mathbb{E}\left[c(s,a)\right], (s,a) \sim d_g$. In (a) and (b), the x-axis and y-axis mean the reward and cost conditions, and the value of both conditions and expectations are normalized to the same scale: $[0, 1]$; (c) The data reconstruction results using the condition $[0.1, 0.5]$ of 10 sampled trajectories in the dataset.

The comparison results of generated trajectories using OASIS and `CVAE` in the `Drone-Circle` task are presented in Fig. 9. This figure illustrates the generation results of OASIS and CVAE under two conditions: low-cost-medium-reward, and medium-cost-high-reward. Although CVAE can reconstruct the trajectories, it fails to integrate conditions into the generation process. In contrast, OASIS successfully controls the generated trajectories, avoiding the restricted area when conditioned on low-cost-medium-reward.

C   Implementation details

C.1   Environment details

Due to the page limit, we omit some descriptions of experiments in the main context. Here we give more details about our experiment tasks. Both the Circle task and the Run task are from a publicly available benchmark [96].
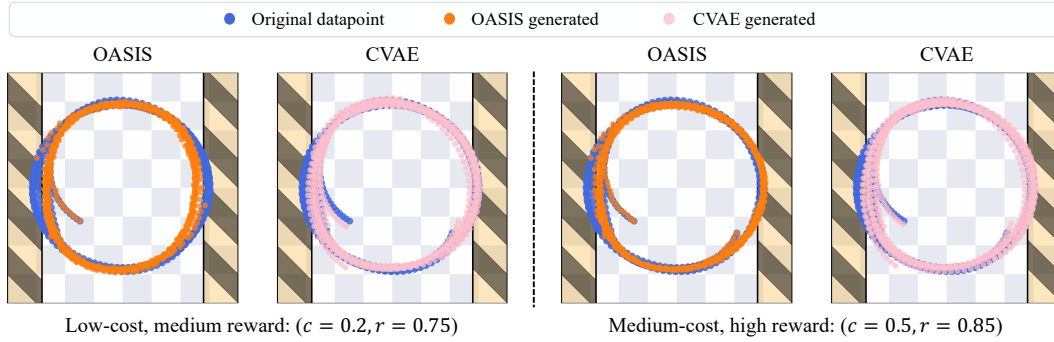
Figure 9: Visualization of generated trajectories in the `Drone-Circle` task.

Circle tasks. The agents are rewarded for running along a circle boundary. The reward function is defined as:

$$r(s, a, s') = \frac{-yv_x + xv_y}{1 + |\sqrt{x^2 + y^2} - radius|} + r_{robot}(s) \tag{44}$$

where x, y are the positions of the agent with state s', v_x, and v_y are velocities of the agent with state s'. $radius$ is the radius of the circle area, and $r_{robot}(s_t)$ is the specific reward for different robot.

The agent gets cost when exceeding the boundaries. The cost function is defined as:

$$\text{Boundary: } c(s_t) = \mathbf{1}(|x| > x_{\lim}) \tag{45}$$

where x_{lim} is the boundary position.

Run tasks. Agents are rewarded for running fast along one fixed direction and are given costs if they run across the boundaries or exceed a velocity limit. The reward function is defined as:

$$r(s, a, s') = ||\boldsymbol{x_{t-1}} - \boldsymbol{g}||_2 - ||\boldsymbol{x_t} - \boldsymbol{g}||_2 + r_{robot}(s_t) \tag{46}$$

The cost function is defined as:

$$c(s, a, s') = \max(1, \mathbf{1}(|y| > y_{lim}) + \mathbf{1}(||\boldsymbol{v_t}||_2 > v_{lim})) \tag{47}$$

where v_{lim} is the speed limit, and y_{lim} is the y position of the boundary, $\boldsymbol{v_t} = [v_x, v_y]$ is the velocity of the agent with state s', $\boldsymbol{g} = [g_x, g_y]$ is the position of a virtual target, $\boldsymbol{x_t} = [x_t, y_t]$ is the position of the agent at timestamp t, $\boldsymbol{x_{t-1}}$ is the Cartesian coordinates of the agent with state s, $\boldsymbol{x_t}$ is the Cartesian coordinates of the agent with state s', and $r_{robot}(s_t)$ is the specific reward for the robot.

Agents. We use three different robot agents in our experiments: `Ball`, `Car`, and `Drone`. The action space dimension, observation space dimension, and the timesteps for these six tasks are shown in Table 3.

Table 3: Environment description

	Max timestep	Action space dimension	Observation space dimension
BallRun	100	2	7
CarRun	200	2	7
DroneRun	200	4	17
BallCircle	200	2	8
CarCircle	300	2	8
DroneCircle	300	4	18

C.2 Dataset details

We provide details about the dataset types we presented in the experiment part. The `Full`, `Tempting`, `Conservative`, and `Hybrid` datasets for `Ball-Circle` and `Car-Circle` tasks are shown in Fig. 10, 11, respectively. All the `Tempting` datasets associated with results in Table 1 are shown in Fig. 12.
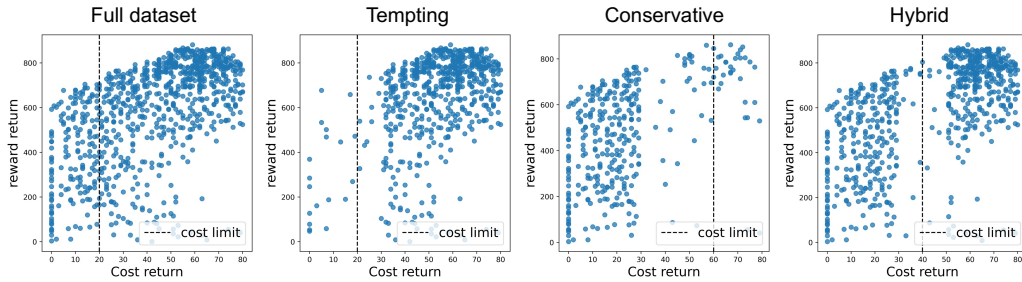

Figure 10: BallCircle Dataset types. Each point represents $(C(\tau), R(\tau))$ of a trajectory τ in the dataset.
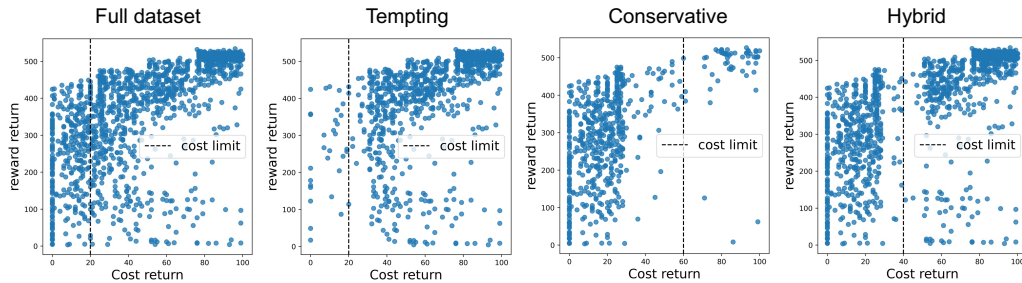

Figure 11: CarCircle Dataset types. Each point represents $(C(\tau), R(\tau))$ of a trajectory τ in the dataset.
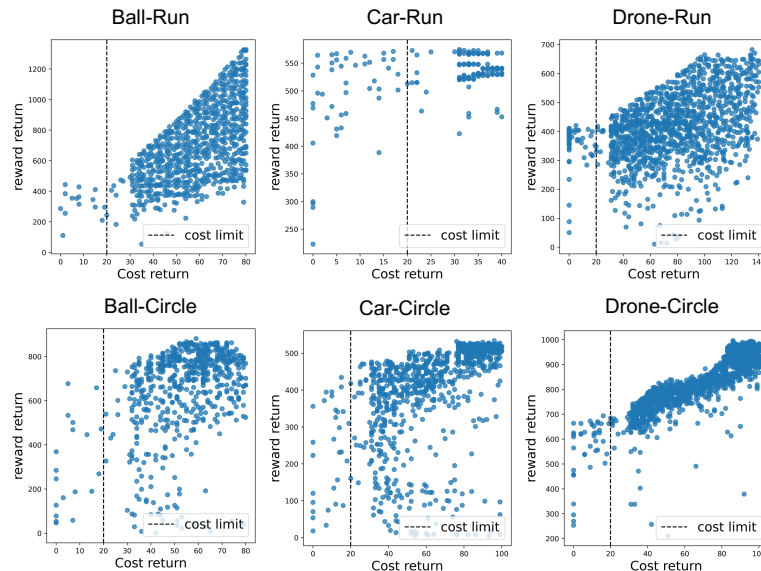

Figure 12: All tempting datasets. Each point represents $(C(\tau), R(\tau))$ of a trajectory τ in the dataset.

C.3 Algorithm details

OASIS algorithm training diagram In this work, we use the cosine β schedule [100] to calculate $\beta_t, t = 1, ..., K$. Then we let $\alpha_t = 1 - \beta_t$, and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$ and denote the state dimension as m. With these notations, we show the training process of the OASIS data generator for one epoch in Algorithm 2.

Algorithm 2: OASIS (training)

1: **Input:** Original Dataset $\mathcal{D}$, predefined β_t and $\bar{\alpha}_t$, diffusion core $\epsilon_\theta (\boldsymbol{x}_k, \boldsymbol{y}, k)$, learning rate lr, loss function $L(\cdot, \cdot)$.
2: **for** each sub-trajectory $\tau_i \in \mathcal{D}$ **do**
3:     Extract the states from τ_i: $\{(s_0, s_1, \ldots, s_T)\}$ # $[T, m]$;
4:     Get the return conditions $\boldsymbol{y} = [C, R]$ associated with these sub-trajectories;
5:     With probability $p = 0.25$ to mask the condition information as: $\boldsymbol{y} \leftarrow \varnothing$
6:     Get Gaussian Noise $noise = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ # $[T, m]$;
7:     Randomly sample time $t \in [0, ..., K - 1]$;
8:     Calculate the forward sampling state $\boldsymbol{x}_{noise} = \bar{\alpha}_t * \tau_i + (1 - \bar{\alpha}_t) * noise$;
9:     Apply initial state condition $\boldsymbol{x}_{noise}[0] \leftarrow s_0$;
10:    Reconstruct noisy sub trajectory $\boldsymbol{x}_{recon} = \epsilon_\theta (\boldsymbol{x}_{noise}, \boldsymbol{y}, k)$;
11:    Minimize the reconstruction loss $\theta \leftarrow \theta - lr * \nabla_\theta L(\boldsymbol{x}_{noise}, \boldsymbol{x}_{recon})$;
12: **end for**
13: **Output:** Updated diffusion core $\epsilon_\theta (\boldsymbol{x}_k, \boldsymbol{y}, k)$;

Network and hyperparameter details. For the dynamics model $\hat{p}$, we utilize a MLP. For the denoising core, we utilize a `U-net`, which has also been used in previous works [82, 80]. The `U-net` is visualized in Fig. 13. The hyperparameters for our method are summarized in Table 4. More details are available in the code.
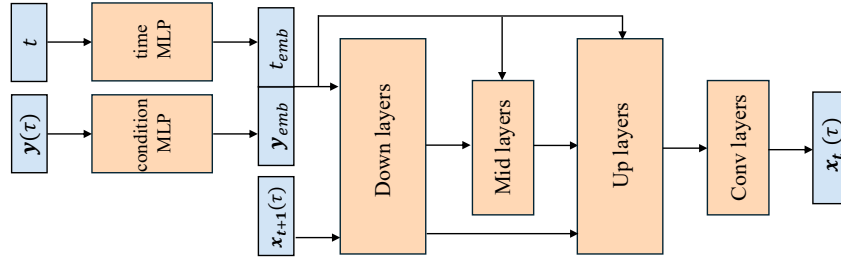

Figure 13: `U-Net` structure.

Baseline details. For the baseline methods `BC`, `BCQ-Lag`, `BEAR-Lag`, `COptiDICE`, `CPQ`, and `CDT`, we adopt the code base provided in the benchmark [74]. For the `FISOR` method, we use the code provided by the authors [31].

Computing resources. The experiments are run on a server with $2\times$AMD EPYC 7542 32-Core Processor CPU, $2\times$NVIDIA RTX A6000 graphics, and 252 GB memory. For one single experiment, OASIS takes about 4 hours with $200,000$ steps to train the data generator. It takes about 1.5 hours to train a `BCQ-Lag` agent on this generated dataset for $200,000$ steps.

Table 4: Hyperparameters

Hyperparameters	Value
L (length of subsequence)	32
K (denoising timestep)	20
Batch size	256
Learning rate	3.0e-5
w_α	2.0

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: They are provided in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: They are provided in the conclusion section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The assumptions are provided in Section 4.4, and the proof is provide in Section A.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: They are provided in Section 5 and Appendix C.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is provided in the anonymous link.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: They are provided in Section 5 and Appendix C.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: They are provided in Section 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The computer resources are provided in Appendix C. More details about experiment results such as reweighting for distribution shaping are also provided in Appendix B.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: Provided in the openreview form.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: They are provided in Section 6.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This paper does not contain pre-trained language models, image generators, or scraped datasets.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The codebase we utilize for baseline comparison is under Apache License 2.0. Related materials are cited in our code (See the README file in the code).

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

   Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

   Answer: [NA]

   Justification: The paper does not release new assets

   Guidelines:

   - The answer NA means that the paper does not release new assets.
   - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
   - The paper should discuss whether and how consent was obtained from people whose asset is used.
   - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

   Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

   Answer: [NA]

   Justification: The paper does not involve crowdsourcing nor research with human subjects.

   Guidelines:

   - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
   - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
   - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

   Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

   Answer: [NA]

   Justification: The paper does not involve crowdsourcing nor research with human subjects.

   Guidelines:

   - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
   - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
   - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
   - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.