Robust Optimization with Diffusion Models for Green Security

Lingkai Kong[1] **Haichuan Wang**[1] **Yuqi Pan**[1] **Cheol Woo Kim**[1] **Mingxiao Song**[1] **Alayna Nguyen**[1]
Tonghan Wang[1] **Haifeng Xu**[2] **Milind Tambe**[1]

[1]Harvard University
[2]University of Chicago

Abstract

In green security, defenders must forecast adversarial behavior—such as poaching, illegal logging, and illegal fishing—to plan effective patrols. These behaviors are often highly uncertain and complex. Prior work has leveraged game theory to design robust patrol strategies to handle uncertainty, but existing adversarial behavior models primarily rely on Gaussian processes or linear models, which lack the expressiveness needed to capture intricate behavioral patterns. To address this limitation, we propose a conditional diffusion model for adversary behavior modeling, leveraging its strong distribution-fitting capabilities. To the best of our knowledge, this is the first application of diffusion models in the green security domain. Integrating diffusion models into game-theoretic optimization, however, presents new challenges, including a constrained mixed strategy space and the need to sample from an unnormalized distribution to estimate utilities. To tackle these challenges, we introduce a mixed strategy of mixed strategies and employ a twisted Sequential Monte Carlo (SMC) sampler for accurate sampling. Theoretically, our algorithm is guaranteed to converge to an ϵ-equilibrium with high probability using a finite number of iterations and samples. Empirically, we evaluate our approach on both synthetic and real-world poaching datasets, demonstrating its effectiveness.

1 INTRODUCTION

In green security, mitigating threats such as illegal logging, illegal fishing, poaching, and environmental pollution requires defenders to anticipate and counteract adversarial behaviors [Fang et al., 2015]. For example, in wildlife conservation, rangers must predict poachers' movements and then strategically allocate patrols to protect endangered species. Over the years, numerous predictive models have been developed [Kar et al., 2017, Gurumurthy et al., 2018, Xu et al., 2020b], alongside robust patrol optimization methods that leverage game theory to enhance decision-making based on these predictions [Xu et al., 2021].

However, existing adversary predictive models [Kar et al., 2017, Gurumurthy et al., 2018] either lack uncertainty quantification [Kong et al., 2023b, Li et al., 2023] or provide only parameterized predictive distributions with limited expressiveness [Xu et al., 2020b]. In reality, adversarial behaviors—such as those of poachers—are high-dimensional and highly complex, driven by diverse motivations, constraints, and strategies. Capturing the full extent of uncertainty is particularly challenging in the strategic environments, where conventional models may struggle to account for the variability of real-world threats.

In this work, we propose using diffusion models to capture adversarial behavior in green security. Diffusion models are a powerful framework for modeling complex, high-dimensional, non-parametric distributions, and they have been successfully applied to image modeling [Ho et al., 2020, Rombach et al., 2021], video generation [Ho et al., 2022], and time-series forecasting [Yang et al., 2024]. By iteratively refining samples through a denoising process, diffusion models can generate diverse and plausible scenarios, offering a more comprehensive representation of potential attacker strategies. To the best of our knowledge, ours is the first attempt to apply diffusion models in green security.

To enhance the robustness of our approach against potential errors in the learned diffusion model (arising from noisy data, limited sample sizes, or imperfect network training), we assume the attacker's true mixed strategy lies within a KL-divergence ball centered around the learned model distribution. We then optimize for the worst-case expected utility within this constrained space. This formulation naturally gives rise to a two-player zero-sum game: while a defender aims to maximize the expected utility, a nature adversary selects the mixed strategy from the KL ball to

Proceedings of the 41st Conference on Uncertainty in Artificial Intelligence (UAI 2025), PMLR 244:2325–2344.

minimize it.

This game-theoretic formulation involving diffusion models introduces new technical challenges that have not been addressed in the literature. First, the KL-divergence constraint on the adversary's mixed strategy prevents the direct application of the standard double oracle method. To resolve this, we shift the constraint from the mixed strategy space to the pure strategy space, treating the original mixed strategy as a pure strategy and introducing a "mixed strategy over mixed strategies." This reformulation yields a more tractable optimization problem. Another challenge arises from the need to sample from a reweighted version of the diffusion model to estimate utilities. To address this, we employ twisted sequential Monte Carlo (SMC) sampling, ensuring asymptotic correctness when evaluating the relevant expected utilities.

Our contributions are as follows: (1) **Novel Adversary Modeling:** We are the first to leverage diffusion models for modeling adversarial behavior in green security domains. (2) **Robust Optimization with Diffusion Model Framework:** We propose DIFFORACLE to mitigate imperfections in learned adversary models by introducing a double oracle algorithm that efficiently computes robust mixed patrol strategies. (3) **Theoretical Guarantees:** We prove that our method converges to an ϵ-equilibrium with high probability under a finite number of iterations and a finite number of samples. (4) **Empirical Performance:** We empirically evaluate our method on both synthetic and real-world poaching data.

2 RELATED WORKS

Diffusion Models Diffusion models have achieved remarkable success across various generative modeling tasks, including image generation [Song et al., 2021, Ho et al., 2020], decision-making [Kong et al., 2024, 2025], and scientific discovery [Gruver et al., 2024, Watson et al., 2023, Kong et al., 2023a]. These models are particularly adept at capturing complex, high-dimensional distributions, making them a powerful tool for diverse applications. Conditional diffusion models extend this capability by integrating contextual information to guide the generative process. By conditioning on textual descriptions, semantic masks, or other relevant features, these models enable tasks such as text-to-image generation [Saharia et al., 2022], image-to-image translation [Saharia et al., 2021], and time series forecasting [Shen and Kwok, 2023].

Double Oracle for Robust Optimization Prior work has framed robust optimization as a two-player zero-sum game [Mastin et al., 2015, Gilbert and Spanjaard, 2017], where the optimizing player selects a potentially randomized feasible strategy, while an adversary chooses problem parameters to maximize regret. The double oracle (DO) algorithm is a standard method for computing equilibria in such games [McMahan et al., 2003, Adam et al., 2021] and has been applied to robust influence maximization in social networks [Wilder et al., 2017], robust patrol planning [Xu et al., 2021], robust submodular optimization [Wilder, 2018], and robust policy design for restless bandits [Killian et al., 2022]. However, these applications restrict the uncertainty set to a compact interval. In contrast, our problem involves a diffusion model that provides full distribution-level predictions, making the uncertainty set a space of distributions, which introduces new theoretical challenges in applying double oracle.

Distributionally Robust Optimization Our work is also closely related to Distributionally Robust Optimization (DRO) [Rahimian and Mehrotra, 2019], which seeks to find robust solutions by optimizing for the worst-case scenario over a set of plausible distributions, known as the ambiguity set. This framework is particularly effective for handling uncertainty and distributional shifts in optimization objectives or constraints. DRO has seen widespread application in areas such as supply chain management [Ash et al., 2022], finance [Kobayashi et al., 2023], and machine learning [Madry et al., 2018, Sagawa* et al., 2020], where resilience to data perturbations is critical. However, most existing DRO methods focus on identifying a single pure strategy, which is dangerous in the green security setting that adversaries can learn to anticipate and exploit. To address this, we propose a game-theoretic approach that derives a mixed strategy for the defender, leveraging randomness to enhance unpredictability and bolster robustness against adversarial exploitation.

Green Security Games Green Security Games (GSGs) use game-theoretic frameworks to safeguard valuable environmental resources from illegal activities such as poaching and illegal fishing [Fang et al., 2015, Hasan et al., 2022]. In these settings, a resource-limited defender protects expansive, spatially distributed areas against attackers with bounded rationality. Prior work focused on forecasting poaching behaviors [Gurumurthy et al., 2018, Moore et al., 2018], learning attacker behavior models from data [Nguyen et al., 2016, Gholami et al., 2018, Xu et al., 2020b], designing patrol strategies [Fang et al., 2015, Xu et al., 2017], and balancing data collection with poaching detection [Xu et al., 2020a].

Among existing studies, Xu et al. [2021] is most closely related to ours, as it also employs a double oracle method to design robust patrolling strategies. However, our approach differs in two key ways. First, we are the first to use diffusion models to predict poaching behavior, addressing the limited expressiveness of the linear approach in Xu et al. [2021]. Second, while Xu et al. [2021] focuses on minimax regret with interval-shaped uncertainty sets, our work adopts a distributionally robust optimization objective.

3 PRELIMINARIES ON DIFFUSION MODEL

A diffusion model [Sohl-Dickstein et al., 2015] is a generative framework composed of two stochastic processes: a *forward* process that progressively adds Gaussian noise to real data, and a *reverse* (or denoising) process that learns to remove this noise step by step. Formally, let $\mathbf{z}^0 \sim \mathcal{D}$ be a sample from the training dataset.[1] The forward diffusion process can be written as $q(\mathbf{z}^t \mid \mathbf{z}^{t-1}) = \mathcal{N}\big(\mathbf{z}^t;\, \mathbf{z}^{t-1},\, \beta^2\mathbf{I}\big)$, where β^2 is the noise variance at each step $t = 1, \dots, T$. As T becomes large, repeated noising transforms the data distribution into (approximately) pure Gaussian noise: $q(\mathbf{z}^T) \approx \mathcal{N}(\mathbf{0},\, T\beta^2\mathbf{I})$.

Score-based Approximation. To invert this process (i.e., to denoise and recover samples from the original data distribution), one can approximate the reverse transition $q(\mathbf{z}^{t-1} \mid \mathbf{z}^t)$ via the *score function*, $\nabla_{\mathbf{z}^t} \log q(\mathbf{z}^t)$ when β is small. Specifically,

$$q(\mathbf{z}^{t-1} \mid \mathbf{z}^t) \;\approx\; \mathcal{N}\Big(\mathbf{z}^{t-1};\, \mathbf{z}^t + \beta^2\, \nabla_{\mathbf{z}^t} \log q(\mathbf{z}^t),\, \beta^2\mathbf{I}\Big).$$

Here, $q(\mathbf{z}^t) = \int q(\mathbf{z}^0)\, q(\mathbf{z}^t \mid \mathbf{z}^0)\, d\mathbf{z}^0$, and the gradient $\nabla_{\mathbf{z}^t} \log q(\mathbf{z}^t)$ points toward regions of higher data density. In practice, we do not know $q(\mathbf{z}^t)$ in closed form, so a neural *score network* $s_\theta(\mathbf{z}^t, t)$ is trained to approximate this gradient via *denoising score matching* [Vincent, 2011, Ho et al., 2020]. Consequently, the learned reverse (denoising) transition becomes

$$p_\theta(\mathbf{z}^{t-1} \mid \mathbf{z}^t) = \mathcal{N}\Big(\mathbf{z}^{t-1};\, \mathbf{z}^t + \beta^2\, s_\theta(\mathbf{z}^t, t),\, \beta^2\mathbf{I}\Big).$$

Starting from an initial Gaussian sample $\mathbf{z}^T \sim \mathcal{N}(\mathbf{0},\, T\beta^2\mathbf{I})$, iterating this reverse process ultimately recovers samples that approximate the original data distribution.

Conditional Extension. This diffusion framework can be naturally extended to include additional context $\mathbf{c}$. In a *conditional* diffusion model [Ho and Salimans, 2021], the score network becomes $s_\theta(\mathbf{z}^t, t, \mathbf{c})$, so that at each step the denoising is informed by side information such as class labels, textual descriptions, or other relevant features. This conditional approach enables the generation of samples that match not only the learned data distribution but also the specific context $\mathbf{c}$, making it particularly useful for tasks in which external conditions strongly influence the underlying data generation process.

4 PROBLEM FORMULATION

In green security settings, a defender (e.g., a ranger) patrols a protected area to prevent resource extraction by an attacker (e.g., a poacher or illegal logger). Let K denote the number of targets—such as 1×1 km regions within the protected area—that require protection. The defender must allocate patrol effort across these targets while adhering to a total budget B. Formally, the patrol strategy is represented as $\mathbf{x} = (x_1, \dots, x_K)$, where x_k denotes the amount of effort (e.g., patrol hours) assigned to target k. The defender's strategy is constrained by: $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^K \mid x_k \geq 0, \forall k, \sum_{k=1}^K x_k \leq B\}$, which ensures that all patrol efforts are non-negative and do not exceed the available budget B.

Attacker Behavior via a Conditional Diffusion Model. Building on the diffusion-model framework in Section 3, we now focus on a poaching scenario, in which an attacker's behavior can be highly uncertain and multimodal. Let $\mathbf{z}$ denote the number of snares or traps placed in each 1×1 km region, where K is its dimensionality. Similarly, let $\mathbf{c}$ represent contextual features, including last month's patrol effort per region [Xu et al., 2021, 2020b], distance to the park boundary, elevation, and land cover. We model the attacker's behavior with a continuous *conditional diffusion model* $p_\theta(\mathbf{z} \mid \mathbf{c})$. Concretely, we treat historical poaching data as samples of $\mathbf{z}^0$, add noise in a forward process, and learn a reverse (denoising) process conditioned on $\mathbf{c}$. Once trained, this diffusion model captures how poachers respond to different patrol allocations and environmental factors. By sampling from $p_\theta(\mathbf{z} \mid \mathbf{c})$ for new contexts, we can generate realistic, diverse poaching scenarios that inform patrol strategy design. Table 1 shows the forecasting results on the real-world poaching data and we can see the **diffusion model can outperform existing approaches used in green security**. Experimental details are in Appendix. H.

Model	MSE
Linear regression	24.40
Gaussian process	24.21 ± 0.04
Diffusion model	**23.46** ± 0.07

Table 1: Forecasting performance in terms of mean squared error (MSE) on poaching data.

Robust Patrol Optimization. In practice, the learned diffusion model may be imperfect due to data noise, limited training samples, or suboptimal network training. Consequently, the learned distribution might not accurately capture the true underlying behavior. To ensure robustness in patrol strategy design, we assume the true distribution lies within a bounded KL-divergence from the learned distribution. We then optimize for the worst-case expected utility over all distributions in that KL-divergence ball, leading to the following formulation:

$$\max_{\pi(\mathbf{x}) \in \Delta(\mathcal{X})} \min_{\tau(\mathbf{z}) \in \mathcal{T}} \quad \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{\tau(\mathbf{z})} [u(\mathbf{x}, \mathbf{z})]$$
$$\mathcal{T} = \{\tau(\mathbf{z}) \mid D_{\mathrm{KL}}(\tau(\mathbf{z}) \parallel p_\theta(\mathbf{z} \mid \mathbf{c})) \leq \rho\}\,, \qquad (1)$$

where ρ is a tolerance parameter specifying how far the true distribution may deviate from the learned distribution. $u(\mathbf{x}, \mathbf{z})$ represents the defender's utility (e.g., the number of animals in the park) for strategy $\mathbf{x}$ given the adversary's choice $\mathbf{z}$ and is assumed to be bounded in $[0, M]$.

Eq. (1) can be interpreted as a two-player zero-sum game in which the *defender* seeks a robust mixed strategy $\pi(\mathbf{x})$, while a *nature adversary* (representing model misspecification) selects $\tau(\mathbf{z})$ within the KL-divergence ball to minimize

[1] We use $\mathbf{z}^0$ and $\mathbf{z}$ interchangeably when there is no ambiguity.

the defender's expected utility. The defender's pure strategy space is $\mathcal{X}$, and the adversary's pure strategy space is $\mathcal{Z} = \mathrm{Support}(p_\theta(\mathbf{z}|\mathbf{c}))$. The defender's mixed strategy $\pi(\mathbf{x})$ is a probability distribution over $\mathcal{X}$, with the corresponding space denoted by $\Delta(\mathcal{X})$. In contrast, the adversary's mixed strategy $\tau(\mathbf{z})$ is in the constained space $\mathcal{T}$.

For notational simplicity, when both players use mixed strategies, the defender's expected utility is denoted by $U(\pi, \tau)$. If one player employs a pure strategy and the other a mixed strategy, we write $U(\mathbf{x}, \tau) := U(\delta_{\mathbf{x}}, \tau)$.

Note that, unlike standard DRO, where the goal is typically to find a single strategy $\mathbf{x}$, here we aim to identify a *mixed* strategy for the defender. This approach is particularly important in green security settings, as adopting a randomized policy helps prevent predictability. A deterministic patrol strategy could be exploited by adversaries, such as poachers, who can adapt their behavior to bypass predictable patterns. By introducing randomness into the patrol strategy, we increase the difficulty for adversaries to anticipate the ranger's actions, thereby enhancing the overall security and effectiveness of the patrol.

5 ROBUST OPTIMIZATION WITH DIFFUSION MODEL

In this section, we propose DIFFORACLE to solve the robust optimization problem in Eq. 1. In Section 5.1, we introduce a mixed strategy over mixed strategies to ensure the applicability of the double oracle approach. Section 5.2 details the overall workflow of the algorithm. In Section 5.3, we present twisted SMC sampler to estimate the expected utility. Finally, in Section 5.4, we provide a convergence analysis of DIFFORACLE.

5.1 MIXED STRATEGY OVER MIXED STRATEGIES

Eq. (1) requires solving for mixed strategies in a continuous game with infinitely many strategies. A common approach for such problems is the double oracle method [Adam et al., 2021], which iteratively expands both players' strategy sets and computes the equilibrium of the resulting subgame. This procedure is guaranteed to converge to an equilibrium in any two-player zero-sum continuous game. However, during the double oracle process, the mixed strategy it produces is necessarily a discrete distribution, whereas $p_\theta(\mathbf{z}|\mathbf{c})$ is a continuous distribution. As a result, the KL divergence between these two distributions is ill-defined, making it difficult to include the KL-divergence constraint in the subgame-equilibrium computation.

To address this limitation, we note that given a fixed $\pi(\mathbf{z})$, the inner constrained minimization problem admits a closed-form solution that can be sampled using the diffusion model. This procedure can be interpreted as computing a best-response pure strategy in the double oracle framework. Consequently, we propose viewing the original mixed strategy $\tau(\mathbf{z})$ as a "pure" strategy and introducing a *mixed strategy over mixed strategies*. This reformulation enables the application of the double oracle method while preserving the desired constraints.

Definition 5.1 (Mixed Strategy over Mixed Strategies)**.** *Let $\mathcal{T}$ denote the space of mixed strategies, where each $\tau \in \mathcal{T}$ represents a probability distribution over pure strategies. A mixed strategy over mixed strategies, σ, is a probability distribution over $\mathcal{T}$, formally expressed as $\sigma \in \Delta(\mathcal{T})$. This implies that σ satisfies the following conditions: (1) $\sigma(\tau) \geq 0$ for all $\tau \in \Delta$, and (2) $\int_{\mathcal{T}} \sigma(\tau)\, d\tau = 1$.*

We provide concrete examples in Appendix B to help readers understand Definition 5.1. By introducing this concept of a mixed strategy over mixed strategies, σ, we can reformulate our objective as follows:

$$\max_{\pi(\mathbf{x})\in\Delta(\mathcal{X})} \min_{\sigma(\tau)\in\Delta(\mathcal{T})} \mathbb{E}_{\pi(\mathbf{x})}\mathbb{E}_{\sigma(\tau)} \left(\mathbb{E}_{\tau(\mathbf{z})}\left[u(\mathbf{x}, \mathbf{z})\right]\right)$$
$$\mathcal{T} \;=\; \{\, \tau(\mathbf{z}) \mid D_{\mathrm{KL}}(\tau(\mathbf{z}) \,\|\, p_\theta(\mathbf{z} \mid \mathbf{c})) \leq \rho\}, \quad (2)$$

In this reformulation, the adversary's pure strategy is no longer a single value but instead a full distribution $\tau(\mathbf{z})$. Consequently, the adversary's pure strategy space becomes $\mathcal{T}$ and the corresponding mixed strategy space is the set of distributions over these distributions, $\Delta(\mathcal{T})$. Under this framework, the defender's utility function takes the form $\mathbb{E}_{\tau(\mathbf{z})}\big[u(\mathbf{x}, \mathbf{z})\big]$, while the attacker's utility becomes $-\mathbb{E}_{\tau(\mathbf{z})}\big[u(\mathbf{x}, \mathbf{z})\big]$.

Crucially, this reformulation shifts the KL-divergence constraint from the adversary's mixed strategy space to its pure strategy space. As we will show in Section 5.2, the best response for such a constrained pure strategy can be written in closed form. Hence, Eq. (2) can be solved efficiently using the double oracle algorithm.

Proposition 5.1. *The reformulated objective in Eq.* (2) *yields the same defender mixed strategy $\pi(\mathbf{x})$ as the original formulation in Eq.* (1)*.*

Proof. See Appendix. C.

By Proposition 5.1, solving Eq. (2) is equivalent to solving Eq. (1). Therefore, applying the double oracle algorithm to Eq. (2) recovers the optimal defender mixed strategy for the original problem (Eq. (1)).

Since we have reformulated the problem, we will henceforth refer to the adversary's pure strategy as $\tau(\mathbf{z})$ and the mixed strategy as $\sigma(\tau)$.

5.2 DOUBLE ORACLE FLOW

The overall double oracle algorithm is outlined in Algorithm 1 and illustrated in Figure 1. We begin by initializing the adversary's strategy as $\tau_0 = p_\theta(\mathbf{z}|\mathbf{c})$ and selecting a random initial defender strategy $\mathbf{x}_0$ from $\mathcal{X}$ (lines 2-3), forming

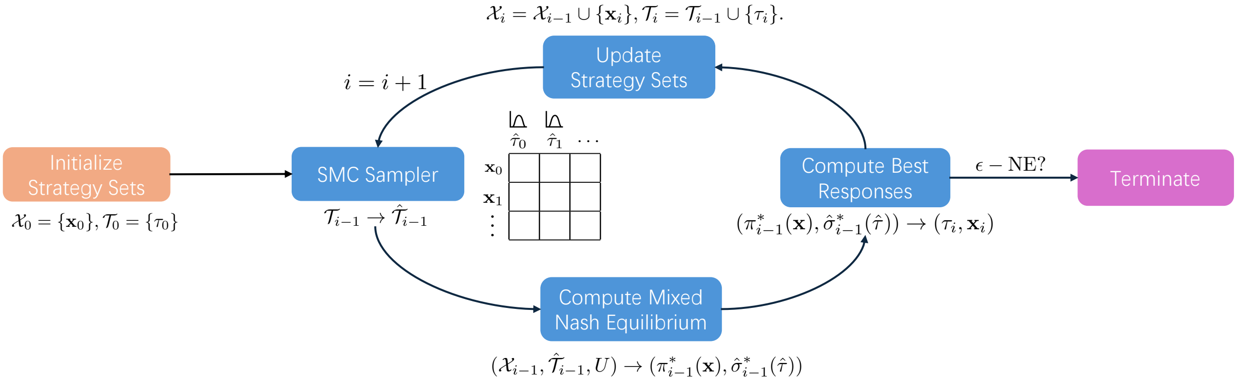

$\mathcal{X}_i = \mathcal{X}_{i-1} \cup \{\mathbf{x}_i\}, \mathcal{T}_i = \mathcal{T}_{i-1} \cup \{\tau_i\}.$
Update Strategy Sets
$i = i + 1$
Initialize Strategy Sets
$\mathcal{X}_0 = \{\mathbf{x}_0\}, \mathcal{T}_0 = \{\tau_0\}$
SMC Sampler
$\mathcal{T}_{i-1} \to \hat{\mathcal{T}}_{i-1}$
$\hat{\tau}_0$ $\hat{\tau}_1$ $\cdots$
$\mathbf{x}_0$
$\mathbf{x}_1$
$\vdots$
Compute Best Responses
$(\pi^*_{i-1}(\mathbf{x}), \hat{\sigma}^*_{i-1}(\hat{\tau})) \to (\tau_i, \mathbf{x}_i)$
$\epsilon-$NE?
Terminate
Compute Mixed Nash Equilibrium
$(\mathcal{X}_{i-1}, \hat{\mathcal{T}}_{i-1}, U) \to (\pi^*_{i-1}(\mathbf{x}), \hat{\sigma}^*_{i-1}(\hat{\tau}))$

Figure 1: Overview of DiffOracle. We begin by initializing the strategy set for each player. At the i-th iteration, we use SMC sampler to obtain a set of empirical distributions $\hat{\mathcal{T}}_{i-1}$. Next, a mixed Nash solver computes the equilibrium π^*_{i-1} and $\hat{\sigma}^*_{i-1}$. We then compute each player's best response against the opponent's mixed strategy and update the players' strategy sets. This procedure is repeated until convergence.

the initial strategy sets $\mathcal{T}_0$ and $\mathcal{X}_0$. These serve as the foundation for the iterative process. In each iteration, we first sample from each distribution in $\mathcal{T}_{i-1} = \{\tau_0, \ldots, \tau_{i-1}\}$ to obtain a set of empirical distributions $\hat{\mathcal{T}}_{i-1} = \{\hat{\tau}_0, \ldots, \hat{\tau}_{i-1}\}$ (line 6). These empirical distributions are used to estimate expected utilities, which are then input into the *Mixed Nash Equilibrium solver* to compute an equilibrium $(\pi^*_{i-1}, \hat{\sigma}^*_{i-1})$ of the subgame $\{\mathcal{X}_{i-1}, \hat{\mathcal{T}}_{i-1}, U\}$ (lines 7). Next, the *defender oracle* and *attacker oracle* compute their respective best responses to the mixed strategy, yielding new strategies $\mathbf{x}_i$ and $\tau_i(\mathbf{z})$ (lines 8-9). These best response strategies are then added to the strategy sets, expanding them to $\mathcal{X}_i$ and $\mathcal{T}_i$ (line 10).

This iterative procedure alternates between the oracles and the solver until convergence (lines 11–13). The parameters prob and tolerance ϵ are user-defined and guarantee that the algorithm converges to an 4ϵ-equilibrium with probability $1 -$ prob, as detailed in Theorem 5.2. In practice, to manage runtime, we cap the number of double oracle iterations to a fixed limit—a common strategy also employed in Lanctot et al. [2017], Xu et al. [2021].

We introduce the details of the three key components defender oracle, adversary oracle and Mixed Nash equilibrium solver as below.

Adversary Oracle At the i-th iteration, given the defender's mixed strategy π^*_{i-1}, the adversary oracle computes the best response by solving:

$$\tau_i(\mathbf{z}) = \arg\max_{\tau \in \mathcal{T}} U(\pi^*_{i-1}, \tau)$$
$$\mathcal{T} = \{\tau_i(\mathbf{z}) \mid D_{\mathrm{KL}}(\tau_i(\mathbf{z}) \parallel p_\theta(\mathbf{z} \mid \mathbf{c})) \leq \rho\}\,, \tag{3}$$

Proposition 5.2. *The optimal solution $\tau_i(\mathbf{z})$ of 3 has a closed-form:*

$$\tau_i(\mathbf{z}) \propto p_\theta(\mathbf{z}|\mathbf{c}) \exp\left(-\gamma U(\pi^*_{i-1}, \mathbf{z})\right)\,, \tag{4}$$

where γ is the Lagrange multiplier associated with the KL-divergence constraint.

Proof. See Appendix D

As shown in Eq. 4, $\tau_i(\mathbf{z})$ is an unnormalized distribution obtained by reweighting the original diffusion model distribution according to the utility function. Computing expected utilities under $\tau_i(\mathbf{z})$ requires sampling from this high-dimensional unnormalized distribution, which is challenging in practice. To address this, we employ twisted Sequential Monte Carlo (SMC) techniques [Chopin et al., 2020, Wu et al., 2023], detailed in Section 4.2, which provide asymptotically exact utility estimates. We denote the resulting empirical distribution as $\hat{\tau}_i(\mathbf{z})$.

Defender Oracle At the i-th epoch, given the attacker's mixed strategy $\hat{\sigma}^*_{i-1}$, the defender oracle computes the best response by solving:

$$\mathbf{x}_i = \arg\max_{\mathbf{x} \in \mathcal{X}} U(\mathbf{x}, \hat{\sigma}^*_{i-1}). \tag{5}$$

Since $\hat{\sigma}^*_{i-1}$ represents a mixed strategy over a set of empirical distributions $\hat{\mathcal{T}}_{i-1}$, we can directly compute the expected utility, reducing the problem to a standard deterministic optimization. To handle the budget constraint in our setting, we employ mirror ascent [Nemirovski, 2012].

Mixed Nash Equilibrium Solver At i-th iteration, the Mixed Nash Equilibrium solver computes a mixed Nash equilibrium $(\pi^*_{i-1}, \hat{\sigma}^*_{i-1})$ over the players' current strategy sets $\mathcal{X}_{i-1}$ and $\hat{\mathcal{T}}_{i-1}$. The equilibrium can be found using linear programming [Nisan et al., 2007], and in our work, we utilize the PuLP implementation [COIN-OR PuLP, 2024] for this purpose.

Algorithm 1 Double Oracle with Diffusion Models

```
Require: Pretrained diffusion model $p_\theta(\mathbf{z} \mid \mathbf{c})$, utility function $U(\mathbf{x}, \tau)$, probability threshold $\mathsf{prob} > 0$
Initialize $i \leftarrow 0$
$\mathbf{x}_0 \leftarrow$ random strategy,   $\tau_0 \leftarrow p_\theta(\mathbf{z} \mid \mathbf{c})$
$\mathcal{X}_0 \leftarrow \{\mathbf{x}_0\}$,   $\mathcal{T}_0 \leftarrow \{\tau_0\}$
repeat
    $i \leftarrow i + 1$
    $\hat{\mathcal{T}}_{i-1} \leftarrow$ Empirical distributions from $\mathcal{T}_{i-1}$ using Alg. 2
    $(\pi^*_{i-1}, \hat{\sigma}^*_{i-1}) \leftarrow \textsc{MixedNashSolver}(\mathcal{X}_{i-1}, \hat{\mathcal{T}}_{i-1}, U)$
    $\mathbf{x}_i \leftarrow \arg\max_{\mathbf{x} \in \mathcal{X}} U(\mathbf{x}, \hat{\sigma}^*_{i-1})$ //Adversary Oracle
    $\tau_i(\mathbf{z}) \propto p_\theta(\mathbf{z}|\mathbf{c}) \exp\big(-\gamma\, U(\pi^*_{i-1}, \mathbf{z})\big)$ // Defender Oracle
    $\mathcal{X}_i \leftarrow \mathcal{X}_{i-1} \cup \{\mathbf{x}_i\}$,   $\mathcal{T}_i \leftarrow \mathcal{T}_{i-1} \cup \{\tau_i\}$
    $\hat{\tau}_i \leftarrow$ sample from $\tau_i$ using Alg 2
    $\underline{v}_i \leftarrow U(\pi^*_{i-1}, \hat{\tau}_i)$,   $\bar{v}_i \leftarrow U(\mathbf{x}_i, \hat{\sigma}^*_{i-1})$
until $(\bar{v}_i - \underline{v}_i \in (-2\epsilon, 2\epsilon)) \wedge (i > 1/(16\,\mathsf{prob}))$
Output: Final defender strategy $\pi^*_{i-1}$
```

5.3 SAMPLING WITH TWISTED SEQUENTIAL MONTE CARLO

To efficiently sample from the unnormalized distribution in Eq. 4 while ensuring correctness, we leverage Twisted Sequential Monte Carlo (Twisted SMC) [Chopin et al., 2020], an adaptive importance sampling technique that improves sampling through sequential proposal and weighting. Wu et al. [2023] applied it to sampling from a conditional distribution with diffusion model; here, we adapt it to sample from the unnormalized reweighted distribution in Eq. 4.

Twisted SMC operates with a collection of N weighted particles $\{(w_n^t, \mathbf{z}_n^t)\}_{n=1}^N$ that evolve iteratively over T steps. At each step t, particles are propagated using an adjusted score function, similar to Chung et al. [2023]:

$$\hat{p}_\theta(\mathbf{z}^{t-1} \mid \mathbf{z}^t, \mathbf{c}) = \mathcal{N}\big(\mathbf{z}^{t-1};\, \mathbf{z}^t + \sigma^2 \hat{s}_\theta(\mathbf{z}^t, \mathbf{c}, t),\, \hat{\beta}^2\big),$$

where the adjusted score function is:

$$\hat{s}_\theta(\mathbf{z}^t, \mathbf{c}, t) = s_\theta(\mathbf{z}^t, \mathbf{c}, t) + \gamma \log \Phi_t(\mathbf{z}^t).$$

The twisting function Φ_t is defined as:

$$\Phi_t(\mathbf{z}_n^t) = \exp\left(-\gamma U(\pi^*_{i-1}, \hat{\mathbf{z}}_\theta^0(\mathbf{z}_n^t))\right). \tag{6}$$

Here, $\hat{\mathbf{z}}_\theta^0(\mathbf{z}^t)$ estimates the original state $\mathbf{z}^0$ using Tweedie's formula [Robbins, 1992, Efron, 2011]:

$$\hat{\mathbf{z}}_\theta^0(\mathbf{z}^t) = \mathbf{z}^t + t\beta^2\, s_\theta(\mathbf{z}^t, \mathbf{c}, t).$$

At $t = 0$, we set $\hat{\mathbf{z}}_\theta^0(\mathbf{z}^0) := \mathbf{z}^0$. The correction term in $\hat{s}_\theta$ reconstructs $\mathbf{z}^0$ and incorporates the reweighted term from Eq. 4, ensuring proper adaptation of the sampling process.

To account for discrepancies between the proposal and target distributions, Twisted SMC assigns a weight to each particle:

$$w_n^t = \frac{p_\theta(\mathbf{x}_n^t | \mathbf{x}_n^{t+1}, \mathbf{c}) \Phi_t(\mathbf{x}_n^t)}{\hat{p}_\theta(\mathbf{x}_n^t | \mathbf{x}_n^{t+1}, \mathbf{c}) \Phi_{t+1}(\mathbf{x}_n^{t+1})}.$$

Algorithm 2 Twisted SMC for Diffusion Model

```
Require: Pretrained diffusion model, number of particles $N$, time horizon $T$, $\Phi(\mathbf{z})$ (Eq. 6)
Initialize $\mathbf{z}_n^T \sim p_\theta(\mathbf{z}^T)$,  $w_n \leftarrow \Phi(\mathbf{z}_n^T)$
for $t = T, \ldots, 1$ do
    Resample:
      $\{\mathbf{z}_n^t\}_{n=1}^N \sim \text{Multinomial}\big(\{\mathbf{z}_n^t\}_{n=1}^N;\ \{w_n^t\}_{n=1}^N\big)$
    for $k = 1 \ldots K$ do
        $\hat{s}_k \leftarrow s_\theta(\mathbf{z}_k^t, \mathbf{c}, t) \ -\ \gamma \nabla_{\mathbf{z}_k^t} \Big[U(\pi^*_{i-1}, \mathbf{z})\Big]$
        $\mathbf{z}_k^{t-1} \sim \mathcal{N}\big(\mathbf{z}_k^t + \sigma^2 \hat{s}_k,\ \hat{\beta}^2\big)$
        $w_k^{t-1} \leftarrow \dfrac{p_\theta\big(\mathbf{z}_k^{t-1} \mid \mathbf{z}_k^t, \mathbf{c}\big)\, \Phi(\mathbf{z}_k^{t-1})}{\hat{p}_\theta\big(\mathbf{z}_k^{t-1} \mid \mathbf{z}_k^t, \mathbf{c}\big)\, \Phi(\mathbf{z}_k^t)}$
    end for
end for
Output: Weighted particles $\{\mathbf{z}_k^0,\ w_k^0\}_{k=1}^K$
```

This reweighting step ensures unbiased estimation.

To mitigate variance and prevent particle degeneracy over long horizons, we apply multinomial resampling at each step based on normalized weights [Douc and Cappé, 2005]. The final approximation of the target distribution is: $\hat{\tau} = \sum_{n=1}^N \frac{w_n^0}{\sum_{n'=1}^N w_{n'}^0} \delta_{\mathbf{z}_n^0}$.

A full description of Twisted SMC is provided in Algorithm 2.

Proposition 5.3. *(Informal) Under regularity conditions on the score function, as the number of particles $N \to \infty$, we have*

$$U(\mathbf{x}, \hat{\tau}(\mathbf{z})) \to U(\mathbf{x}, \tau(\mathbf{z})) \quad \textit{almost surely},$$

where $\hat{\tau}$ is the empirical distribution returned by Algorithm 2.

Proof. See Appendix. E.

5.4 CONVERGENCE ANALYSIS

In Section 5.4, we analyze the convergence properties of our framework. For theoretical analysis, we introduce two mild assumptions.

Assumption 1. *We assume that the utility function is twice differentiable and concave with respect to* $\mathbf{x}$.

Assumption 1 implies there is diminishing marginal return in ranger effort, which is a common assumption in economics models [Mankiw, 1998] and reflects the intuition that initial patrol efforts contribute more significantly to wildlife protection than additional increments in effort. Under assumption 1, Eq. 5 is a convex optimization problem and existing optimization solvers [Diamond and Boyd, 2016] can accurately find the defender's best response.

Assumption 2. *We assume that the distribution $p_\theta(\mathbf{z} \mid \mathbf{c})$ places its mass on a compact space.*

In practice, the attacker's action at each target must lie in a bounded interval, e.g. $[0, z_{\max}]$. For instance, the number of snares at any region cannot exceed a practical upper limit. Consequently, it is reasonable to treat the action space as compact, ensuring that $p_\theta(\mathbf{z} \mid \mathbf{c})$ has compact support.

For each $\hat{\sigma}_i^*$, we denote the corresponding mixed strategy on the underlying true adversary strategy distribution as σ_i^*. Formally, $\sigma_i^*(\tau_l) = \hat{\sigma}_i^*(\hat{\tau}_l)\ \forall l \in [i]$. Without the terminating condition, Algorithm 1 produces two sequences of mixed strategies: $(\pi_i^*)_{i=0}^{\infty}$ and $(\sigma_i^*)_{i=0}^{\infty}$. Proposition 5.3 says if we use infinite samples to estimate expected utilities, then there is no estimation error and Theorem 5.1 follows from the original double oracle algorithm's proof [Adam et al., 2021].

Theorem 5.1. *Without terminating conditions, under assumptions 1, 2, if we use $N \to \infty$ samples for all iterations, every weakly convergent subsequence of Alg. 1 converges to an exact equilibrium in possibly infinite iterations. Such a weakly convergent subsequence always exists.* [2]

However, in practical scenarios where only a finite number of samples is available, the estimation of the expected utility is imprecise. Consequently, estimation errors will appear in the following steps within each iteration of our algorithm: (1) solving the subgame, (2) computing the defender oracle, and (3) evaluating the terminating condition.

Theorem 5.2. *Under assumptions 1 and 2, with finite number of samples at the i-th iteration*

$$N_i = \left\lceil CM^2(i+1)^2 i^{1+\delta}/\epsilon^2 \right\rceil,$$

for each adversary distribution, where C is a constant, M is the upper bound of utility function, ϵ is the approximation error, and δ is any positive number.

- ***Item 1:** Without terminating condition, every weakly convergent subsequence of Alg. 1 converges to an ϵ-equilibrium in a possibly infinite number of iterations. Such a weakly convergent subsequence always exist.*
- ***Item 2:** With the terminating condition, Alg. 1 terminates in a finite number of iterations. Also, it converges to a finitely supported 4ϵ-equilibrium with probability at least $1 - \mathsf{prob}$.*

Proof. We provide a sketch of the proof here and defer the full details to Appendix G. The key steps for proving Item 1 are as follows:

- **Step 1:** We bound the utility estimation error for any mixed strategy pair at iteration i by the maximum estimation error over all entries in the payoff matrix.
- **Step 2:** We show that, under our finite sampling scheme, the probability that the maximum cell-wise error exceeds $\epsilon/4$ is nonzero only during the first i_r iterations, for some finite i_r.
- **Step 3:** We treat the strategies generated in the first i_r rounds as the initial strategy set in the standard Double Oracle (DO) algorithm [Adam et al., 2021]. We then adapt the original convergence proof to account for the error introduced by finite sampling, which is now bounded by $\epsilon/4$.

By relaxing the error bound in Item 1, we obtain convergence within a finite number of iterations. The additional approximation error in Item 2 stems from two sources: (1) enforcing finite termination, and (2) using estimated utilities of mixed strategy pairs when evaluating the stopping condition.

□

In practice, we use a fixed number of samples across iterations, and experiments in Section 6 shows our framework still achieves robust performance.

6 EXPERIMENTS

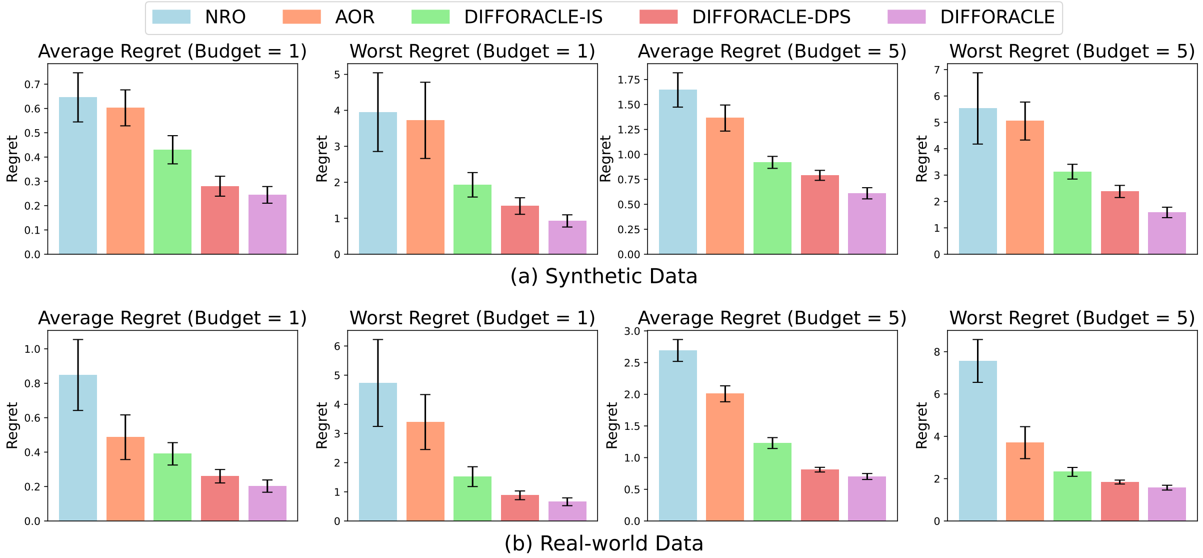
6.1 EXPERIMENTAL SETUP

Datasets We conduct experiments on both synthetic and real-world datasets which we describe below. We use a graph based dataset [Nguyen et al., 2016] to reflect geospatial constraints in the poaching domain for patrollers.

Synthetic data. Poaching counts are sampled from a Gamma distribution parameterized by shape and scale values. To determine the shape parameter, we randomly select one of two Graph Convolutional Networks (GCNs) [Kipf and Welling, 2022] with randomly initialized weights to map the node's feature vector to a continuous value, which serves as the shape parameter. The scale parameter is set to 1 if the first GCN is chosen and 0.9 if the second is selected. Finally, adversarial noise inversely proportional to the poaching count is added, ensuring that nodes with lower poaching counts receive higher noise levels.

Real-world Data. We use poaching data from Murchison Falls National Park (MFNP) in Uganda, collected between 2010 and 2021. The protected area is discretized into 1×1 km grid cells. For each cell, we measure ranger patrol effort (in kilometers patrolled) as the conditional variable for the diffusion model, while the monthly number of detected illegal activity instances serves as the adversarial behavior. Following Basak et al. [2016], we represent the park as a graph to capture geospatial connectivity among these cells. To focus on high-risk regions, we subsample 20 subgraphs from the entire graph. Specifically, at each month we identify the 20 cells with the highest poaching counts. Each of

[2] We include the definition of weak convergence in Appendix F.



(a) Synthetic Data

(b) Real-world Data

Figure 2: Experimental Results on both synthetic and real-world datasets. Following [Ho et al., 2020], we average the results over 5 random seeds.

these cells is treated as a central node, and we iteratively add the neighboring cell with the highest poaching count until the subgraph reaches 20 nodes. This process generates 532 training, 62 validation, and 31 test samples.

Baselines We compare the following methods:

Non-robust Optimization (NRO). We use a baseline that directly maximizes the expected utility under the pre-trained diffusion model. The stochastic optimization is solved via sample average approximation, using samples from the diffusion model in conjunction with mirror ascent. Since this approach yields only a pure strategy, we repeat the process with different initializations to obtain five distinct pure strategies. These pure strategies are then combined into a mixed strategy by assigning them equal probability.

Alternate Optimization with Random Reinitialization (AOR). This method solves the DRO problem using alternating optimization without employing the double oracle framework. It iterates between optimizing the defender's strategy and sampling from the worst-case distribution using twisted SMC. Similar to NRO, we construct a mixed strategy by running the procedure five times with different initializations, generating multiple pure strategies that are then combined with equal probability.

We also compare against three variations of DIFFORACLE:

DIFFORACLE with Importance Sampling (DIFFORACLE-IS). This variant replaces the twisted SMC sampler in Section 5.3 with importance sampling to sample from Eq. 4. As the proposal distribution, we directly use the pre-trained diffusion model, $p_\theta(\mathbf{z}|\mathbf{c})$.

DIFFORACLE with Diffusion Posterior Sampling (DIFFORACLE-DPS). This variant employs the diffusion posterior sampler [Chung et al., 2023] instead of the twisted SMC sampler in Section 5.3.

DIFFORACLE. This version retains the default twisted SMC sampler in Section 5.3 to sample from Eq. 4.

Evaluation metrics We evaluate the methods using decision *regret*, defined as the difference between the defender's best possible utility under the true adversarial behavior and the expected utility under the optimized mixed strategy. We report both the average regret on the test set and the worst-case regret on the test set.

Implementation details We employ a three-layer GCN with a hidden dimension of 128 as the backbone of the diffusion model. The optimizer used is Adam [Kingma, 2014] with a learning rate of 10^{-3}. We use 500 samples to estimate the expected utility for all the menthods. γ is selected on the validation set based on the average regret. More details of the implementation details are provided in Appendix H.

6.2 EXPERIMENTAL RESULTS

Main Results. We evaluate our method against baselines on both synthetic and real-world poaching datasets under different patrol budgets ($B = 1$ and $B = 5$). The results, presented in Fig. 2, show that DIFFORACLE consistently achieves the lowest average regret and worst-case regret across all settings.

Compared to NRO, DIFFORACLE reduces average regret by 62.2%, 62.9%, 73.3%, and 74.0% across different datasets and budgets. Similarly, worst-case regret is significantly reduced by 59.1%, 64.9%, 71.3%, and 79.0%. These improvements highlight the robustness of our approach, which is particularly crucial in green security domains, where minimizing worst-case outcomes is essential for high-stakes decision-making.

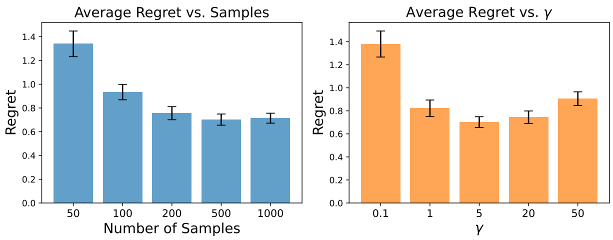


Figure 3: Parameter Study on DIFFORACLE-SMC on poaching data under Budget 5.

The double-oracle framework provides substantial benefits, as all three variants of DIFFORACLE significantly outperform the naive robust optimization approach, AOR. This is because AOR relies on a simple heuristic to solve the minimax problem and construct the mixed strategy, lacking convergence guarantees. Consequently, AOR exhibits greater variance and instability, further underscoring the advantages of employing game-theoretic methods for robust optimization.

Among the DIFFORACLE methods employing different sampling strategies, DPS emerges as the strongest alternative to twisted SMC. However, SMC consistently outperforms DPS, demonstrating statistically significant improvements in five out of eight cases. Furthermore, DPS cannot exactly sample from the target distribution in Eq.4 [Lu et al., 2023], a critical requirement for ensuring the convergence of the double-oracle framework, as analyzed in Section 5.4.

Parameter Study. Fig. 3 presents the parameter study of DIFFORACLE using the twisted SMC sampler. As shown, varying the number of samples in the sampler reveals that once the sample size exceeds 200, performance stabilizes. Additionally, when adjusting the value of γ, we observe that performance drops significantly as γ approaches 0, since it effectively reduces to non-robust optimization. Conversely, when γ is too large, performance also declines because the nature adversary may select a worst-case distribution that deviates too far from the learned distribution, making it non-informative.

7 CONCLUSION

We introduced a conditional diffusion model for adversary behavior modeling in green security, overcoming the limitations of traditional Gaussian process and linear models. To the best of our knowledge, this is the first application of diffusion models in this domain. To integrate diffusion models into game-theoretic optimization, we proposed a mixed strategy of mixed strategies and leverage a twisted Sequential Monte Carlo (SMC) sampler for efficient sampling from unnormalized distributions. We established theoretical convergence to an ϵ-equilibrium with high probability using finite samples and finite iterations and demonstrated empirical effectiveness on both synthetic and real-world poaching datasets. Future work could explore extensions to sequential-decision-making.

ACKNOWLEDGEMENTS

We are thankful to the Uganda Wildlife Authority for granting us access to incident data from Murchison Falls National Park. We also thank Charles Emogor for the insightful discussions and the anonymous reviewers for their valuable feedback. This work was supported by ONR MURI N00014-24-1-2742.

REFERENCES

Lukáš Adam, Rostislav Horčík, Tomáš Kasl, and Tomáš Kroupa. Double oracle algorithm for computing equilibria in continuous games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5070–5077, 2021.

Cecil Ash, Claver Diallo, Uday Venkatadri, and Peter VanBerkel. Distributionally robust optimization of a canadian healthcare supply chain to enhance resilience during the covid-19 pandemic. *Computers & Industrial Engineering*, 168:108051, 2022.

Anjon Basak, Fei Fang, Thanh Hong Nguyen, and Christopher Kiekintveld. Abstraction methods for solving graph-based security games. In *Autonomous Agents and Multiagent Systems: AAMAS 2016 Workshops, Visionary Papers, Singapore, Singapore, May 9-10, 2016, Revised Selected Papers*, pages 13–33. Springer, 2016.

Patrick Billingsley. *Convergence of probability measures*. John Wiley & Sons, 2013.

Nicolas Chopin, Omiros Papaspiliopoulos, et al. *An introduction to sequential Monte Carlo*, volume 4. Springer, 2020.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=OnD9zGAGT0k`.

COIN-OR PuLP. PuLP: A Linear Programming Toolkit for Python, 2024. URL `https://coin-or.github.io/pulp/main/index.html`. Accessed: 2024-02-05.

Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Randal Douc and Olivier Cappé. Comparison of resampling schemes for particle filtering. In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.*, pages 64–69. Ieee, 2005.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496): 1602–1614, 2011.

Fei Fang, Peter Stone, and Milind Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, July 2015.

Shahrzad Gholami, Sara Mc Carthy, Bistra Dilkina, Andrew Plumptre, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, Mustapha Nsubaga, Joshua Mabonga, et al. Adversary models account for imperfect crime data: Forecasting and planning against real-world poachers. International Conference on Autonomous Agents and Multiagent Systems, 2018.

Hugo Gilbert and Olivier Spanjaard. A double oracle approach to minmax regret optimization problems with interval data. *European Journal of Operational Research*, 262(3):929–943, 2017.

Nate Gruver, Samuel Stanton, Nathan Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. *Advances in neural information processing systems*, 36, 2024.

Swaminathan Gurumurthy, Lantao Yu, Chenyan Zhang, Yongchao Jin, Weiping Li, Xiaodong Zhang, and Fei Fang. Exploiting data and human knowledge for predicting wildlife poaching. In *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies*, pages 1–8, 2018.

Dewan Tariq Hasan, Md Mosaddek Khan, Muhammad Ibrahim, and Ibrahem Almansour. On evaluation of patrolling and signalling schemes to prevent poaching in green security games. *Intelligent Systems with Applications*, 14:200083, 2022.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021.

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

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.

Debarun Kar, Benjamin Ford, Shahrzad Gholami, Fei Fang, Andrew Plumptre, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, Mustapha Nsubaga, et al. Cloudy with a chance of poaching: Adversary behavior modeling and forecasting with real-world poaching data. International Conference on Autonomous Agents and Multiagent Systems, 2017.

Jackson A Killian, Lily Xu, Arpita Biswas, and Milind Tambe. Restless and uncertain: Robust policies for restless bandits via deep multi-agent reinforcement learning. In *Uncertainty in Artificial Intelligence*, pages 990–1000. PMLR, 2022.

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2022.

Ken Kobayashi, Yuichi Takano, and Kazuhide Nakata. Cardinality-constrained distributionally robust portfolio optimization. *European Journal of Operational Research*, 309(3):1173–1182, 2023.

Lingkai Kong, Jiaming Cui, Haotian Sun, Yuchen Zhuang, B Aditya Prakash, and Chao Zhang. Autoregressive diffusion model for graph generation. In *International conference on machine learning*, pages 17391–17408. PMLR, 2023a.

Lingkai Kong, Harshavardhan Kamarthi, Peng Chen, B Aditya Prakash, and Chao Zhang. Uncertainty quantification in deep learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5809–5810, 2023b.

Lingkai Kong, Yuanqi Du, Wenhao Mu, Kirill Neklyudov, Valentin De Bortoli, Dongxia Wu, Haorui Wang, Aaron Ferber, Yi-An Ma, Carla P Gomes, et al. Diffusion models as constrained samplers for optimization with unknown constraints. *arXiv preprint arXiv:2402.18012*, 2024.

Lingkai Kong, Haichuan Wang, Tonghan Wang, Guojun Xiong, and Milind Tambe. Composite flow matching for reinforcement learning with shifted-dynamics data. *arXiv preprint arXiv:2505.23062*, 2025.

Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Pérolat, David Silver, and Thore Graepel. A unified game-theoretic approach to multiagent reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Yinghao Li, Lingkai Kong, Yuanqi Du, Yue Yu, Yuchen Zhuang, Wenhao Mu, and Chao Zhang. Muben: Benchmarking the uncertainty of molecular representation models. *arXiv preprint arXiv:2306.10060*, 2023.

Cheng Lu, Huayu Chen, Jianfei Chen, Hang Su, Chongxuan Li, and Jun Zhu. Contrastive energy prediction for exact energy-guided diffusion sampling in offline reinforcement learning. In *International Conference on Machine Learning*, pages 22825–22855. PMLR, 2023.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

N Gregory Mankiw. *Principles of microeconomics*, volume 1. Elsevier, 1998.

Andrew Mastin, Patrick Jaillet, and Sang Chin. Randomized minmax regret for combinatorial optimization under uncertainty. In *International symposium on algorithms and computation*, pages 491–501. Springer, 2015.

H Brendan McMahan, Geoffrey J Gordon, and Avrim Blum. Planning in the presence of cost functions controlled by an adversary. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 536–543, 2003.

Jennifer F Moore, Felix Mulindahabi, Michel K Masozera, James D Nichols, James E Hines, Ezechiel Turikunkiko, and Madan K Oli. Are ranger patrols effective in reducing poaching-related threats within protected areas? *Journal of Applied Ecology*, 55(1):99–107, 2018.

Arkadi Nemirovski. Tutorial: Mirror descent algorithms for large-scale deterministic and stochastic convex optimization. In *Conf. Learn. Theory*, 2012.

Thanh H Nguyen, Arunesh Sinha, Shahrzad Gholami, Andrew Plumptre, Lucas Joppa, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, and Rob Critchlow. Capture: A new predictive anti-poaching tool for wildlife protection. 2016.

Noam Nisan, Tim Roughgarden, Éva Tardos, and Vijay V. Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, 2007. ISBN 978-0-521-87282-9. URL https://doi.org/10.1017/CBO9780511800481.

Hamed Rahimian and Sanjay Mehrotra. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

Herbert E Robbins. An empirical bayes approach to statistics. In *Breakthroughs in Statistics: Foundations and basic theory*, pages 388–394. Springer, 1992.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. 2022 ieee. In *CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, 2021.

Shiori Sagawa*, Pang Wei Koh*, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Chitwan Saharia, William Chan, Huiwen Chang, Chris A Lee, Jonathan Ho, Tim Salimans, David J Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. 10.48550. *arXiv preprint arXiv.2111.05826*, 2021.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Lifeng Shen and James Kwok. Non-autoregressive conditional diffusion models for time series prediction. In *International Conference on Machine Learning*, pages 31016–31029. PMLR, 2023.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Bryan Wilder. Equilibrium computation and robust optimization in zero sum games with submodular structure. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Bryan Wilder, Amulya Yadav, Nicole Immorlica, Eric Rice, and Milind Tambe. Uncharted but not uninfluenced: Influence maximization with an uncertain network. In *Proceedings of the 16th conference on autonomous agents and multiagent systems*, pages 1305–1313, 2017.

Luhuan Wu, Brian L. Trippe, Christian A Naesseth, John Patrick Cunningham, and David Blei. Practical and asymptotically exact conditional sampling in diffusion models. In *Thirty-seventh Conference on Neural*

Information Processing Systems, 2023. URL https://openreview.net/forum?id=eWKqr1zcRv.

Haifeng Xu, Benjamin Ford, Fei Fang, Bistra Dilkina, Andrew Plumptre, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, Mustapha Nsubaga, and Joshua Mabonga. Optimal patrol planning for green security games with black-box attackers. In *Decision and Game Theory for Security*, pages 458–477, Cham, 2017. Springer International Publishing. ISBN 978-3-319-68711-7.

Lily Xu, Elizabeth Bondi-Kelly, Fei Fang, A. Perrault, Kai Wang, and Milind Tambe. Dual-mandate patrols: Multi-armed bandits for green security. In *AAAI Conference on Artificial Intelligence*, 2020a. URL https://api.semanticscholar.org/CorpusID:221655680.

Lily Xu, Shahrzad Gholami, Sara McCarthy, Bistra Dilkina, Andrew Plumptre, Milind Tambe, Rohit Singh, Mustapha Nsubuga, Joshua Mabonga, Margaret Driciru, et al. Stay ahead of poachers: Illegal wildlife poaching prediction and patrol planning under uncertainty with field test evaluations (short version). In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 1898–1901. IEEE, 2020b.

Lily Xu, Andrew Perrault, Fei Fang, Haipeng Chen, and Milind Tambe. Robust reinforcement learning under minimax regret for green security. In Cassio de Campos and Marloes H. Maathuis, editors, *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pages 257–267. PMLR, 27–30 Jul 2021.

Yiyuan Yang, Ming Jin, Haomin Wen, Chaoli Zhang, Yuxuan Liang, Lintao Ma, Yi Wang, Chenghao Liu, Bin Yang, Zenglin Xu, et al. A survey on diffusion models for time series and spatio-temporal data. *arXiv preprint arXiv:2404.18886*, 2024.

Appendix for Robust Optimization with Diffusion Models for Green Security

A MORE DETAILS ON DIFFUSION MODELS

A diffusion model [Sohl-Dickstein et al., 2015] is a generative framework composed of two stochastic processes: a *forward* process that progressively adds Gaussian noise to real data, and a *reverse* (or denoising) process that learns to remove this noise step by step. Formally, let $\mathbf{z}^0 \sim \mathcal{D}$ be a sample from the training dataset.[3] The forward diffusion process can be written as $q(\mathbf{z}^t \mid \mathbf{z}^{t-1}) = \mathcal{N}\big(\mathbf{z}^t;\, \mathbf{z}^{t-1},\, \beta^2\mathbf{I}\big)$, where β^2 is the noise variance at each step $t = 1, \ldots, T$. As T becomes large, repeated noising transforms the data distribution into (approximately) pure Gaussian noise: $q(\mathbf{z}^T) \approx \mathcal{N}(\mathbf{0},\, T\beta^2\mathbf{I})$.

Score-based Approximation. To invert this process (i.e., to denoise and recover samples from the original data distribution), one can approximate the reverse transition $q(\mathbf{z}^{t-1} \mid \mathbf{z}^t)$ via the *score function*, $\nabla_{\mathbf{z}^t} \log q(\mathbf{z}^t)$ when β is small. Specifically,

$$q(\mathbf{z}^{t-1} \mid \mathbf{z}^t) \;\approx\; \mathcal{N}\Big(\mathbf{z}^{t-1};\, \mathbf{z}^t + \beta^2\, \nabla_{\mathbf{z}^t} \log q(\mathbf{z}^t),\, \beta^2\mathbf{I}\Big).$$

Here, $q(\mathbf{z}^t) = \int q(\mathbf{z}^0)\, q(\mathbf{z}^t \mid \mathbf{z}^0)\, d\mathbf{z}^0$, and the gradient $\nabla_{\mathbf{z}^t} \log q(\mathbf{z}^t)$ points toward regions of higher data density. In practice, we do not know $q(\mathbf{z}^t)$ in closed form, so a neural *score network* $s_\theta(\mathbf{z}^t, t)$ is trained to approximate this gradient via *denoising score matching* [Vincent, 2011, Ho et al., 2020]. Consequently, the learned reverse (denoising) transition becomes

$$p_\theta(\mathbf{z}^{t-1} \mid \mathbf{z}^t) = \mathcal{N}\Big(\mathbf{z}^{t-1};\, \mathbf{z}^t + \beta^2\, s_\theta(\mathbf{z}^t, t),\, \beta^2\mathbf{I}\Big).$$

Starting from an initial Gaussian sample $\mathbf{z}^T \sim \mathcal{N}(\mathbf{0},\, T\beta^2\mathbf{I})$, iterating this reverse process ultimately recovers samples that approximate the original data distribution.

Conditional Extension. This diffusion framework can be naturally extended to include additional context $\mathbf{c}$. In a *conditional* diffusion model [Ho and Salimans, 2021], the score network becomes $s_\theta(\mathbf{z}^t, t, \mathbf{c})$, so that at each step the denoising is informed by side information such as class labels, textual descriptions, or other relevant features. This conditional approach enables the generation of samples that match not only the learned data distribution but also the specific context $\mathbf{c}$, making it particularly useful for tasks in which external conditions strongly influence the underlying data generation process.

Rather than directly estimating the score function $s_\theta(\mathbf{z}^t, t)$, Denoising Diffusion Probabilistic Models (DDPM) [Ho et al., 2020] reformulate the learning objective as a *noise prediction* task. This reparameterization leverages the closed-form expression of the forward process:

$$\mathbf{z}^t = \sqrt{\bar{\alpha}_t}\mathbf{z}^0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}),$$

[3] We use $\mathbf{z}^0$ and $\mathbf{z}$ interchangeably when there is no ambiguity.

where $\bar{\alpha}_t$ denotes the cumulative product of noise schedules. The training objective becomes recovering the noise $\boldsymbol{\epsilon}$ that perturbed $\mathbf{z}^0$ to form $\mathbf{z}^t$. A neural network $\boldsymbol{\epsilon}_\theta(\mathbf{z}^t, t)$ is trained to approximate this noise, which corresponds to learning the score function up to a time-dependent scaling:

$$s_\theta(\mathbf{z}^t, t) = -\frac{\boldsymbol{\epsilon}_\theta(\mathbf{z}^t, t)}{\sqrt{1-\bar{\alpha}_t}}.$$

Training then reduces to minimizing a simple mean squared error (MSE) loss between the true and predicted noise:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{\mathbf{z}_0,\, \boldsymbol{\epsilon},\, t}\big[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{c})\|^2\big].$$

By training this conditional diffusion model on historical poaching data—augmented with contextual features $\mathbf{c}$—we learn $p_\theta(\mathbf{z} \mid \mathbf{c})$, a powerful and expressive model of poacher behavior. This enables us to capture complex, multimodal patterns of attacker responses, thereby supporting the development of robust patrol strategies discussed earlier.

B EXAMPLES OF MIXED STRATEGY OVER MIXED STRATEGIES

Let us consider a national park with 3 target regions to protect, and poachers' pure strategies specify how many snares to put in each target region. Two examples of poacher pure strategies could be $\mathbf{z}_1 = (3, 4, 3)$ and $\mathbf{z}_2 = (0, 0, 10)$. Each entry in the pure strategy determines the number of snares a poacher will place in the corresponding target region. Let us denote poachers' pure strategy space as $\mathcal{Z} = \{\mathbf{z}_1, \mathbf{z}_2\}$.

A mixed strategy τ is a distribution on the pure strategy space, i.e., $\tau \in \Delta(\mathcal{Z})$. Denote the subset of mixed strategies which satisfy the constraint $D_{\text{KL}}(\tau(\mathbf{z})||p_\theta(\mathbf{z}|\mathbf{c})) \leq \rho$ as $\mathcal{T}$. One such example τ_1 could be $P(\mathbf{z}_1) = 0.1$ and $P(\mathbf{z}_2) = 0.9$. Another degenerate example of mixed strategy τ_2 could be $P(\mathbf{z}_1) = 0$ and $P(\mathbf{z}_2) = 1$.

A mixed strategy over mixed strategies σ is a distribution on the constrained mixed strategy space, i.e., $\sigma \in \Delta(\mathcal{T})$. One example of mixed strategy over mixed strategies σ_1 could be $P(\tau_1) = 0.1$ and $P(\tau_2) = 0.9$. Another degenerate example σ_2 could be $P(\tau_1) = 0$ and $P(\tau_2) = 1$.

A mixed strategy over mixed strategies is still a distribution on the original pure strategy space, i.e., $\sigma \in \Delta(\mathcal{Z})$. For example, an alternative way to view σ_1 could be

$$P(\mathbf{z}_1) = P(\sigma_1(\tau_1)) \cdot P(\tau_1(\mathbf{z}_1)) + P(\sigma_1(\tau_2)) \cdot P(\tau_2(\mathbf{z}_1)) = 0.01$$

and

$$P(\mathbf{z}_2) = P(\sigma_1(\tau_1)) \cdot P(\tau_1(\mathbf{z}_2)) + P(\sigma_1(\tau_2)) \cdot P(\tau_2(\mathbf{z}_2)) = 0.99$$

However, it is proven in Proposition 5.1 that all mixed strategy over mixed strategies σ satisfy $D_{\text{KL}}(\sigma||p_\theta(\mathbf{z}|\mathbf{c})) \leq \rho$, which is not generally true for elements in $\Delta(\mathcal{Z})$.

From Section 5.2 onward, readers can interpret $\mathcal{T}$ as the pure strategy space and σ as a standard mixed strategy. Despite each pure strategy $\tau \in \mathcal{T}$ being a distribution, all standard terminologies of game theory remain applicable.

C PROOF OF PROPOSITION 5.1

We now show that for any $\pi(\mathbf{x}) \in \Delta(\mathcal{X})$,

$$\min_{\tau(\mathbf{z})} \left\{\mathbb{E}_{\pi(\mathbf{x})}\mathbb{E}_{\tau(\mathbf{z})}\left[u(\mathbf{x}, \mathbf{z})\right] : D_{\text{KL}}(\tau(\mathbf{z}) \,\|\, p_\theta(\mathbf{z}|\mathbf{c})) \leq \rho\right\} =$$
$$\min_{\sigma(\tau)} \left\{\mathbb{E}_{\pi(\mathbf{x})}\mathbb{E}_{\sigma(\tau)}\left(\mathbb{E}_{\tau(\mathbf{z})}\left[u(\mathbf{x}, \mathbf{z})\right]\right) : D_{\text{KL}}(\tau(\mathbf{z}) \,\|\, p_\theta(\mathbf{z}|\mathbf{c})) \leq \rho\right\}.$$

From this, the original theorem follows.

Consider any solution $\tau'(\mathbf{z})$ that attains the minimum on the left-hand side. Define a degenerate distribution over strategies $\sigma'(\tau) = \delta[\tau = \tau']$, i.e., it places all its mass on τ'. Note that τ' satisfies the divergence constraint on the left, so $\sigma'(\tau)$ will also satisfy the corresponding constraint on the right-hand side. Since the expected value under $\sigma'(\tau)$ matches the value attained by τ', we have the left side is not smaller than the right side.

Now take any solution $\sigma'(\tau)$ that attains the minimum on the right side. Define $\tau'(\mathbf{z}) = \mathbb{E}_{\sigma'(\tau)}[\tau(\mathbf{z})]$. Because a mixture over mixed strategies is itself a valid mixed strategy in $\Delta(\mathcal{Z})$, $\tau'(\mathbf{z})$ is admissible on the left side.

By the convexity of the divergence measure D, we have:

$$D_{\mathrm{KL}}(\tau'(\mathbf{z}) \parallel p_\theta(\mathbf{z}|\mathbf{c})) = D_{\mathrm{KL}}\big(\mathbb{E}_{\sigma'(\tau)}\tau(\mathbf{z}) \parallel p_\theta(\mathbf{z}|\mathbf{c})\big) \leq \mathbb{E}_{\sigma'(\tau)}[D_{\mathrm{KL}}(\tau(\mathbf{z}) \parallel p_\theta(\mathbf{z}|\mathbf{c})] \leq \rho.$$

Here, the first inequality follows from the convexity of D, and the second inequality is by the construction of $\sigma'(\tau)$, which satisfies the original constraint on the right side.

Thus, $\tau'(\mathbf{z})$ satisfies the left side constraint and attains the same expected value as $\sigma'(\tau)$. We then obtain that the left side is not larger than the right side.

Combining both parts, we conclude the proof.

D PROOF OF PROPOSITION 5.2

Proof. We introduce a Lagrange multiplier α for the KL-divergence constraint and another multiplier λ for the normalization constraint. The Lagrangian is

$$\mathcal{L}(\tau, \alpha, \lambda) \;=\; \int \tau(\mathbf{z})\Big(U(\pi^*_{i-1}, \mathbf{z})\Big)\, d\mathbf{z} \;-\; \alpha\Big(D_{\mathrm{KL}}(\tau(\mathbf{z})||p_\theta(\mathbf{z}|\mathbf{c})) - \rho\Big) \;+\; \lambda\Big(\int \tau(\mathbf{z})\, d\mathbf{z} - 1\Big).$$

By taking the functional derivative of $\mathcal{L}$ with respect to $\tau(\mathbf{z})$ and setting it to zero, one obtains

$$\tau(\mathbf{z}) \;\propto\; p_\theta(\mathbf{z} \mid \mathbf{c}) \exp\Big(\tfrac{1}{\alpha}\, U(\pi^*_{i-1}, \mathbf{z}))\Big).$$

Defining $\gamma = -\frac{1}{\alpha}$ (where $\gamma > 0$ absorbs constants and signs from the Lagrange approach) gives the closed-form solution

$$\tau_i(\mathbf{z}) \;\propto\; p_\theta(\mathbf{z} \mid \mathbf{c}) \exp\Big(-\gamma U(\pi^*_{i-1}, \mathbf{z})\Big),$$

which matches Eq. (4). This completes the proof. □

E PROOF OF PROPOSITION 5.3

We first provide the full statement of Proposition 5.3 as below.

Proposition 5.3 (Twisted SMC)**.** *Suppose the following conditions hold:*

1. $\Phi_t(\mathbf{z}^T)$ *and* $\Phi_t(\mathbf{z}^t)/\Phi_{t-1}(\mathbf{z}^{t-1})$ *are positive and bounded.*
2. *For* $t > 0$, $\log \Phi_t(\mathbf{z}^t)$ *is continuous and has bounded gradients with respect to* $\mathbf{z}^t$.
3. $\hat{\beta}^2 > \beta^2$.

Almost Sure Convergence: *Under these assumptions, as the number of particles* $N \to \infty$*, we have*

$$U(\mathbf{x}, \hat{\tau}(\mathbf{z})) \to U(\mathbf{x}, \tau(\mathbf{z})) \quad \textit{almost surely},$$

where $\hat{\tau}$ *is the empirical distribution returned by Algorithm 2.*

Error Bound under Finite Samples: *Under these assumptions, the mean squared error of the twisted SMC sampler satisfies the bound:*

$$\mathbb{E}\left[|U(\mathbf{x}, \tau(\mathbf{z})) - U(\mathbf{x}, \hat{\tau}(\mathbf{z}))|^2\right] \leq \frac{C'M^2}{N},$$

where C' *is a constant and* M *is the maximum value of the utility function.*

Justification of Assumptions:

- **Assumption (1):** This holds if $\exp(-\gamma U(\pi, \mathbf{z}))$ is positive and bounded away from zero. In our green security domain, U is always positive (as introduced in Section X), ensuring this condition is automatically satisfied.

- **Assumption (2):** This is justified by the Appendix A.5 of Wu et al. [2023].
- **Assumption (3):** This can be ensured by selecting a sufficiently large $\hat{\beta}$.

Proof. Recall that $p_\theta(\mathbf{z}|\mathbf{c})$ serves as the prior, while the likelihood is given by $\exp\left(-\gamma U(\pi, \mathbf{z})\right)$.

We first prove that the marginal distribution of the sampler is $\tau(\mathbf{z})$:

$$\begin{aligned}
\hat{p}(\mathbf{z}^{0:T}) &= \frac{1}{Z}\left[p_\theta(\mathbf{z}^T)\prod_{t=1}^{T-1}\hat{p}(\mathbf{z}^t|\mathbf{z}^{t-1})\right]\left[\Phi_T(\mathbf{z}^T)\prod_{t=1}^{T-1}\frac{p_\theta(\mathbf{z}^t|\mathbf{z}^{t+1})\Phi_t(\mathbf{z}^t)}{\hat{p}_\theta(\mathbf{z}^t|\mathbf{z}^{t+1})\Phi_{t+1}(\mathbf{z}^{t+1})}\right] \\
&= \frac{1}{Z}\left[p_\theta(\mathbf{z}^T)\prod_{t=1}^{T-1}p(\mathbf{z}^t|\mathbf{z}^{t-1})\right]\left[\Phi_T(\mathbf{z}^T)\prod_{t=1}^{T-1}\frac{\hat{p}_\theta(\mathbf{z}^t|\mathbf{z}^{t+1})\Phi_t(\mathbf{z}^t)}{\hat{p}_\theta(\mathbf{z}^t|\mathbf{z}^{t+1})\Phi_{t+1}(\mathbf{z}^{t+1})}\right] \\
&= \frac{1}{Z}p_\theta(\mathbf{z}^{0:T})\left[\prod_{t=0}^{T-1}\frac{\Phi_t(\mathbf{z}^t)}{\Phi_{t+1}(\mathbf{z}^{t+1})}\right]\Phi_T(\mathbf{z}^T) \\
&= \frac{1}{Z}p_\theta(\mathbf{z}^{0:T})\Phi_0(\mathbf{z}^0).
\end{aligned} \tag{7}$$

Since $\Phi_0(\mathbf{z}^0) = \exp(-\gamma U(\pi, \mathbf{z}^0))$, marginalizing out $\mathbf{z}^{1:T}$ yields

$$\hat{p}(\mathbf{z}^0) = \tau(\mathbf{z}^0) \propto p_\theta(\mathbf{z}|\mathbf{c})\exp\left(-\gamma U(\pi, \mathbf{z})\right),$$

as desired.

Next, according to Appendix A.5 in Wu et al. [2023], under the given assumptions, the importance weights w^t remain bounded. Consequently, applying Propositions 11.5 and 11.3 from Chopin et al. [2020] establishes almost sure convergence and the error bound under finite samples.

□

F DEFINITION OF WEAK CONVERGENCE

We directly cite the definition of weak convergence provided in Adam et al. [2021], and a more detailed discussion of the convergence of probability measures can be seen in Billingsley [2013].

Definition F.1. *A sequence of mixed strategy $(\pi_i^*)_{i=1}^\infty$ in $\Delta(\mathcal{X})$ weakly converges to $\pi^* \in \Delta(\mathcal{X})$ if*

$$\lim_{i\to\infty}\int_{\mathcal{X}} f(x)d\pi_i = \int_{\mathcal{X}} f(x)d\pi^*$$

for every continuous function $f : \mathcal{X} \to R$. We use $\pi_i \Rightarrow \pi^*$ to denote weak convergence.

G PROOF OF THEOREM 5.2

Theorem 5.2. *Under assumptions 1 and 2, with finite number of samples at the i-th iteration*

$$N_i = \left\lceil CM^2(i+1)^2 i^{1+\delta}/\epsilon^2\right\rceil,$$

for each adversary distribution, where C is a constant, M is the upper bound of utility function, ϵ is the approximation error, and δ is any positive number.

- ***Item 1:*** *Without terminating condition, every weakly convergent subsequence of Alg. 1 converges to an ϵ-equilibrium in a possibly infinite number of iterations. Such a weakly convergent subsequence always exist.*
- ***Item 2:*** *With the terminating condition, Alg. 1 terminates in a finite number of iterations. Also, it converges to a finitely supported 4ϵ-equilibrium with probability at least $1 - \mathsf{prob}$.*

The constant C in N_i can be expressed as $16C'$, where C' is the constant in **Error Bound under Finite Sample** discussed in Appendix E. To prove Theorem 5.2, we first prove the utility estimation error bound for the twisted diffusion sampler.

Lemma 1. *Under the same assumptions as Proposition 5.3, the utility estimation error of the twisted SMC sampler satisfies the bound:*

$$P(|U(\mathbf{x}, \tau(\mathbf{z})) - U(\mathbf{x}, \hat{\tau}(\mathbf{z}))| \geq \epsilon) \leq \frac{C'M^2}{N\epsilon^2},$$

*where C' is the constant in **Error Bound under Finite Sample** of Appendix E and M is the maximum value of the utility function.*

Proof. Consider the random variable $U(\mathbf{x}, \tau(\mathbf{z})) - U(\mathbf{x}, \hat{\tau}(\mathbf{z}))$, whose variance is upper bounded by $E|(U(\mathbf{x}, \tau(\mathbf{z})) - U(\mathbf{x}, \hat{\tau}(\mathbf{z}))^2|$. By **Error Bound under Finite Sample**, we know that this variance is upper bounded by $\frac{C'M^2}{N}$.

Applying Chebyshev's inequality to the random variable $U(\mathbf{x}, \tau(\mathbf{z})) - U(\mathbf{x}, \hat{\tau}(\mathbf{z}))$, we have

$$P(|U(\mathbf{x}, \tau(\mathbf{z})) - U(\mathbf{x}, \hat{\tau}(\mathbf{z})| \geq \epsilon) \leq \frac{C'M^2}{N\epsilon^2}$$

□

Notation We introduce notations used in the proof of Theorem 5.2. Recall at the i-th iteration of algorithm 1, we use an empirical distribution $\hat{\tau}_i$ with N_i samples to approximate each adversary strategy (distribution) $\tau_i \in \mathcal{T}_i$. Because of the finite sample approximation, the utility estimation for each pure strategy pair in the payoff matrix is imprecise. Define the estimation error in row j, column k of payoff matrix at iteration i as

$$\Delta_i^{j,k} = U(x_j, \tau_k) - U(x_j, \hat{\tau}_k).$$

Let Δ_i denote the absolute value of the largest utility estimation error for any cell in the payoff matrix at the i-th iteration of the algorithm, i.e., $\Delta_i = \max_{j,k} |\Delta_i^{j,k}|$. At step 7 of algorithm 1, when we apply linear programming to solve the subgame $(\mathcal{X}_i, \hat{\mathcal{T}}_i, U)$, we obtain a mixed strategy for adversary $\hat{\sigma}_i^*$ defined on $\hat{\mathcal{T}}_i$. Recall $\sigma_i^* \in \Delta(\mathcal{T}_i)$ is the mixed strategy on the underlying true adversary distribution that shares the same weight as $\hat{\sigma}_i^*$. Formally, $\sigma_i^*(\tau_l) = \hat{\sigma}_i^*(\hat{\tau}_l)\ \forall l \in [i]$.

Proof of Item 1

Proof. We first show that for any $\pi_i \in \Delta(\mathcal{X}_i)$ and $\sigma_i \in \Delta(\mathcal{T}_i)$, we have $|U(\pi_i, \hat{\sigma}_i) - U(\pi_i, \sigma_i)| \leq \Delta_i$. We write

$$|U(\pi_i, \sigma_i) - U(\pi_i, \hat{\sigma}_i)| = \sum_{\mathbf{x} \in \mathcal{X}_i} \sum_{\tau \in \mathcal{T}_i} \pi_i(\mathbf{x}) \cdot \sigma_i(\tau) \cdot |U(\mathbf{x}, \tau) - U(\mathbf{x}, \hat{\tau})| \tag{8}$$

$|U(\mathbf{x}, \tau) - U(\mathbf{x}, \hat{\tau})|$ denotes the sample estimation error for the pure strategy pair $(\mathbf{x}, \tau)$ in the payoff matrix. The maximum on the right-hand side of Equation 8 is obtained when putting all the probability mass on the strategy pair with the largest sample estimation error, which is Δ_i.

Then we bound Δ_i for each i. For any cell (j, k) in the matrix, we apply Lemma 1:

$$P(|\Delta_i^{j,k}| \geq \frac{\epsilon}{4}) \leq \frac{16C'M^2}{N\epsilon^2}.$$

Since at i-th iteration, there are $(i+1)^2$ cells in the payoff matrix, we apply the union bound and obtain:

$$P(\Delta_i \geq \frac{\epsilon}{4}) \leq \frac{16C'M^2(i+1)^2}{N_i\epsilon^2}.$$

By setting $N_i = \lceil 16C'M^2(i+1)^2 i^{1+\delta}/\epsilon^2 \rceil$, we have

$$P(\Delta_i \geq \frac{\epsilon}{4}) \leq \frac{1}{i^{1+\delta}}.$$

Here we consider the events $A_i = \{\Delta_i \geq \frac{\epsilon}{4}\}$. From the step above, we have

$$P(A_i) \leq \frac{1}{i^{1+\delta}}.$$

Because $\delta > 0$, $\sum_{i=1}^{\infty} \frac{1}{i^{1+\delta}}$ is a convergent series. Therefore, $\sum_{i=1}^{\infty} P(A_i) < \infty$. From Borel-Cantelli Lemma, we then obtain $P(\limsup_{i\to\infty} A_i) = 0$, which implies A_i only happens for finite times. There exists i_r that for any $i > i_r$,

$$P(\Delta_i \geq \frac{\epsilon}{4}) = 0.$$

Because of assumption 2, the pure strategy space $\mathcal{X}$ and $\mathcal{T}$ are both compact and U is continuous. Hence, $(\mathcal{X}, \mathcal{T}, U)$ is a two-player zero-sum continuous game, and here are several results that are already proven in Adam et al. [2021] for two-player zero-sum continuous games.

- Sequences $(\pi_i^*)_{i=1}^{\infty}$ and $(\sigma_i^*)_{i=1}^{\infty}$ have a weakly convergent subsequence, which for simplicity, will be denoted by the same indices. Therefore, $\pi_i^* \Rightarrow \pi^*$ for some π^* and $\sigma_i^* \Rightarrow \sigma^*$ for some σ^*, where $\Rightarrow$ denotes weak convergence.
- If $\pi_i \Rightarrow \pi$ in $\Delta(\mathcal{X})$ and $\sigma_i \Rightarrow \sigma$ in $\Delta(\mathcal{T})$, then $U(\pi_i, \sigma_i) \to U(\pi, \sigma)$. If $\pi_i \Rightarrow \pi$ in $\Delta(\mathcal{X})$ and $\tau_i \to \tau$ in $\mathcal{T}$, then $U(\pi_i, \tau_i) \to U(\pi, \tau)$.
- For any $\pi \in \Delta(\mathcal{X})$ we have
$$\min_{\tau \in \mathcal{T}} U(\pi, \tau) = \min_{\sigma \in \Delta(\mathcal{T})} U(\pi, \sigma)$$
- The size of initial subset X_1 and Y_1 can be any finite number.

From the proof above, A_i only happens for finite times. Assume i_r is the largest number satisfying that A_i happens. We then treat $(\mathcal{X}_{i_r}, \mathcal{T}_{i_r})$ as the initial set of strategies for both players. Then our sampling scheme ensures that for any strategy pair (π, σ) and iteration i, we have $|U(\pi_i, \sigma_i) - U(\pi_i, \hat{\sigma}_i)| \leq \Delta_i \leq \epsilon/4$.

Consider any $\mathbf{x}$ such that $\mathbf{x} \in \mathcal{X}_{i_0}$ for some i_0. Take an arbitrary $i \geq i_0$, which implies $\mathbf{x} \in \mathcal{X}_i$. Since $(\pi_i^*, \hat{\sigma}_i^*)$ is an equilibrium of the subgame $(\mathcal{X}_i, \hat{\mathcal{T}}_i, U)$, we get

$$U(\pi_i^*, \hat{\sigma}_i^*) \geq U(\mathbf{x}, \hat{\sigma}_i^*)$$

Since $U(\pi_i^*, \sigma_i^*)$ and $U(\pi_i^*, \hat{\sigma}_i^*)$ differ by at most $\frac{\epsilon}{4}$, and $U(\mathbf{x}, \sigma_i^*)$ and $U(\mathbf{x}, \hat{\sigma}_i^*)$ differ by at most $\frac{\epsilon}{4}$, we have

$$U(\pi_i^*, \sigma_i^*) + \frac{\epsilon}{2} \geq U(\mathbf{x}, \sigma_i^*) \to U(\mathbf{x}, \sigma^*).$$

Since $U(\pi_i^*, \sigma_i^*) \to U(\pi^*, \sigma^*)$, we have

$$U(\pi^*, \sigma^*) + \frac{\epsilon}{2} \geq U(\mathbf{x}, \sigma^*) \tag{9}$$

for all $\mathbf{x} \in \cup \mathcal{X}_i$. Since U is continuous, the previous inequality holds for all $\mathbf{x} \in cl(\cup \mathcal{X}_i)$.

Fix now an arbitrary $\mathbf{x} \in \mathcal{X}$. Note $\mathbf{x}_{i+1}$ is the best response to $U(\cdot, \hat{\sigma}_i^*)$ (since ranger oracle uses finite sample estimation of payoff matrix), and we have

$$U(\mathbf{x}_{i+1}, \hat{\sigma}_i^*) \geq U(\mathbf{x}, \hat{\sigma}_i^*)$$

Because $U(\mathbf{x}_{i+1}, \sigma_i^*)$ and $U(\mathbf{x}_{i+1}, \hat{\sigma}_i^*)$ differ by at most $\epsilon/4$, and $U(\mathbf{x}, \sigma_i^*)$ and $U(\mathbf{x}, \hat{\sigma}_i^*)$ differ by at most $\epsilon/4$, we have

$$U(\mathbf{x}_{i+1}, \sigma_i^*) + \frac{\epsilon}{2} \geq U(\mathbf{x}, \sigma_i^*) \to U(\mathbf{x}, \sigma^*)$$

Since $\mathbf{x}_{i+1} \in \mathcal{X}_{i+1}$ and by compactness of $\mathcal{X}$, we can select a convergence subsequence $\mathbf{x}_i \to \tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}} \in cl(\cup \mathcal{X}_i)$. This allows us to use 9 to obtain

$$U(\mathbf{x}_{i+1}, \sigma_i^*) \to U(\tilde{\mathbf{x}}, \sigma^*) \leq U(\pi^*, \sigma^*) + \frac{\epsilon}{2}.$$

Therefore, for any $\mathbf{x} \in X$,

$$U(\mathbf{x}, \sigma^*) \leq U(\pi^*, \sigma^*) + \epsilon.$$

Similarly, we have for any $\tau \in \mathcal{T}$,

$$U(\pi^*, \tau) \geq U(\pi^*, \sigma^*) - \frac{\epsilon}{2}.$$

The two sides are not symmetrical because the best response for the poacher doesn't use the finite sample approximation of payoff matrix, thus having a smaller error. We then conclude the proof. □

We then show that adding the terminating condition, for any $\epsilon > 0$, algorithm 1 can terminate in a finite number of iterations. Also, when it stops, it converges to a 4ϵ-equilibrium with high probability.

Proof of Item 2

Proof. Consider now an infinite game, from Item 1 in Theorem 5.2, we know that $\bar{v}_i - \underline{v}_i \to \epsilon$. Also, our sampling scheme ensures that for any strategy pair (π, σ) and iteration i after some finite rounds i_r, we have $|U(\pi_i, \sigma_i) - U(\pi_i, \hat{\sigma_i})| \leq \Delta_i \leq \frac{\epsilon}{4}$. This indicates that with $\epsilon > 0$, the terminating condition will be satisfied after a finite number of iterations. Assume that the algorithm ends at the j-th iteration. This implies

$$U(\mathbf{x}_{j+1}, \hat{\sigma}_j^*) - U(\pi_j^*, \hat{\tau}_{j+1}) \in (-2\epsilon, 2\epsilon)$$

Then we have

$$\begin{aligned}
U(\pi_j, \sigma_j) &\leq U(\pi_j, \hat{\sigma}_j) + \Delta_j \\
&\leq U(\mathbf{x}_{j+1}, \hat{\sigma}_j) + \Delta_j \\
&\leq U(\pi_j, \hat{\tau}_{j+1}) + \Delta_j + 2\epsilon \\
&\leq U(\pi_j, \tau_{j+1}) + 2\Delta_j + 2\epsilon \\
&= \arg\min_{\tau} U(\pi_j, \tau) + 2\Delta_j + 2\epsilon.
\end{aligned}$$

Here the first and fourth relation follows from the estimation error of U. The second one is because $\mathbf{x}_{j+1}$ is the best response for the ranger. The third one is from the terminating condition and the fifth one comes from the best response for the poacher. Similarly,

$$\begin{aligned}
U(\pi_j, \sigma_j) &\geq U(\pi_j, \sigma_{j+1}) \\
&\geq U(\pi_j, \hat{\sigma}_{j+1}) - \Delta_j \\
&\geq U(\mathbf{x}_{j+1}, \hat{\sigma}_j) - \Delta_j - 2\epsilon \\
&\geq U(\mathbf{x}^*, \hat{\sigma}_j) - \Delta_j - 2\epsilon, \text{ where } \mathbf{x}^* = \arg\max_{\mathbf{x}} U(\mathbf{x}, \sigma_j) \\
&\geq \arg\max_{\mathbf{x}} U(\mathbf{x}, \sigma_j) - 2\Delta_j - 2\epsilon.
\end{aligned}$$

Here the second and fifth relation comes from the estimation error of U. The first one is from the best response of the poacher and the fourth one is from the best response of the ranger. The third one follows from the terminating condition.

For Δ_j, we have

$$P(\Delta_j \geq \epsilon) \leq \frac{1}{16 j^{1+\delta}} \leq \mathsf{prob},$$

where the second relation comes from $j > \frac{1}{16\mathsf{prob}}$. Therefore, we show that (π_j, σ_j) is a 4ϵ-equilibrium with probability at least $1 - \mathsf{prob}$.

□

H EXPERIMENTAL DETAILS

H.1 DATASETS

Synthetic Dataset To better reflect real-world conditions, regions are connected based on a predefined topology. We randomly generate 5,100 graphs, each with 30 nodes and 20 edges. The first 4,800 graphs are used for training, the next 200 for validation, and the remaining 100 for testing. Each node is assigned a randomly generated 10-dimensional feature vector. Next, we establish a stochastic mapping from a node's features to its poaching count, capturing the complex relationships observed in real-world scenarios. Poaching counts are sampled from a Gamma distribution parameterized by shape and scale values. We randomly initialize two Graph Convolutional Networks (GCNs). For each node, one of the two GCNs is selected with equal probability to map the node's features to a continuous value, which is then scaled by a factor of 20. This value serves as the shape parameter of the Gamma distribution. The poaching count is then drawn from the Gamma distribution, where the scale parameter is set to 1 if the first GCN is chosen and 0.9 if the second is chosen. To incorporate adversarial noise, we apply perturbations inversely proportional to the poaching count—nodes with lower poaching counts receive higher noise levels. Finally, the poaching count for each node is capped within the range $[0, 40]$ and scaled by 0.2 to align the overall distribution with real-world data.

Real-world Dataset We use poaching data from Murchison Falls National Park (MFNP) in Uganda, collected between 2010 and 2021. The protected area is discretized into 1 × 1 km grid cells. For each cell, we measure ranger patrol effort (in kilometers patrolled) as the conditional variable for the diffusion model, while the monthly number of detected illegal activity instances of each cell serves as the adversarial behavior. Following [Basak et al., 2016], we represent the park as a graph to capture geospatial connectivity among these cells. To focus on high-risk regions, we subsample 20 subgraphs from the entire graph. Specifically, at each time step we identify the 20 cells with the highest poaching counts. Each of these cells is treated as a central node, and we iteratively add the neighboring cell with the highest poaching count until the subgraph reaches 20 nodes. This procedure yields 532 training samples, 62 validation samples, and 31 test samples.

H.2 IMPLEMENTATION DETAILS

We use a three-layer Graph Convolutional Network (GCN) [Kipf and Welling, 2022] with a hidden dimension of 128 as the backbone of the diffusion model. The diffusion process follows the DDPM framework [Ho et al., 2020] with $T = 1000$ time steps and a variance schedule from 10^{-4} to 0.02. Optimization is performed using Adam [Kingma, 2014] with a learning rate of 10^{-3}, and the model is trained for 5000 epochs. To estimate the expected utility, we draw 500 samples from the diffusion model. All comparison methods run for 30 iterations. The mirror ascent oracle uses a step size of 0.1 and runs for 100 iterations. The step size in the mirror ascent step for the baselines is also 0.1.

The actions of the poacher and ranger in grid j, represented by z_j and x_j respectively, influence the wildlife population in the area. We model the wildlife population in grid j as follows:

$$\max(N_0(j)e^r - \alpha e^{\psi \mathbf{z}_j - \theta \mathbf{x}_j}, 0),$$

where $N_0(j)$ is the initial wildlife population in the area and r denotes the natural growth rate of the wildlife. The parameter α captures the impact of both the ranger's and poacher's actions on the wildlife population, ψ reflects the strength of poaching, and θ measures the effectiveness of patrol effort. The utility for the ranger is then represented as the sum of wildlife population across all grids:

$$U(\mathbf{x}, \mathbf{z}) = \sum_{j=1}^{K} \max(N_0(j)e^r - \alpha e^{\psi \mathbf{z}_j - \theta \mathbf{x}_j}, 0)$$

.

Forecasting Experiments. We use the poaching dataset described in Appendix H.1. Following Xu et al. [2021], linear regression and Gaussian processes predict the poaching count for each 1 × 1 km cell individually, using two features: the previous month's patrol effort in the current cell and the aggregated patrol effort from neighboring cells. For linear regression, we employ the scikit-learn implementation, while for Gaussian processes, we use the GPy library with both the RBF and Matérn kernels. The training procedure for the diffusion model follows Appendix H.2, with its support constrained to $[0, 3]$. For each test instance, we generate 500 samples and use the mean prediction. We also attempted to impose constraints on the baseline's output but found that this only degraded its performance.