Safe Active Exploration

Author
Alexander Spiridonov
aspiridonov@ethz.ch

Author
Gian Hess
gihess@ethz.ch

Author
Iason Chalas
ichalas@ethz.ch

Abstract

Experimental design is a key challenge in science and engineering. Classical experimental design associates experiments with states that can freely be chosen. We consider the rich case where the structure of a given Markov Decision Process constrains the choice of experiments. Additionally, many real-world experiments have safety constraints. Certain states or actions may be dangerous or simply off-limits. At the same time, we often want to learn about some unknown quantity in the unsafe region. This problem is relevant in many real-world settings, from robotic inspection of industrial sites to nuclear fusion. We propose a safe state-action density constraint set that guarantees safe policies. Additionally, we propose a new reweighted experimental design objective that leverages correlation between the safe and unsafe regions. This allows us to estimate an unknown quantity in the unsafe region from observations in the safe region. Finally, we provide both theoretical and experimental analysis of the convergence.

1 Introduction

Optimal experimental design is a ubiquitous challenge in science and engineering [9]. Given some fixed budget, we want to learn as much as possible about some unknown quantity. During experiments, we gather observations and use them to estimate the unknown quantity. In the Multi-Armed Bandit literature experiment, states can freely be chosen. We consider the case where one can not freely choose the states of the experiment. Instead, one traverses the possible states according to some known Markov Chain. In this setting, every experiment is associated with a policy that can be rolled out in the Markov Chain. Thus the goal of learning about the unknown quantity reduces to finding a policy that allocates the budget of observations optimally. Previous work in this direction by Tarbouriech et al. [11], and Mutný et al. [7], already established algorithms that provably converge to the optimal policy.

In this work, we extend the previously introduced setting to the important case of safety-critical experiments. For many real-world experiments, it is essential to remain in a safe region of the state-action space. Unsafe states or actions may cause harm to people or equipment and must be avoided at all costs. At the same time, we might also be interested in estimating the unknown quantity in the unsafe region. Consider the illustrative example in Fig. 1. A robot is used at an industrial site to measure a gas concentration. At the site, the gas concentration is highly correlated between pipes and compressors, respectively. Part of the site is restricted for the robot and contains many pipes. Current state-of-the-art methods do not consider the correlation between safe and unsafe pipes. Accordingly, they will visit the pipe less frequently since there are more safe compressors then pipes. This behavior is not optimal if we want to estimate the gas concentration over the whole site.

Foundations of Reinforcement Learning, FS2023. Department of Computer Science, ETH Zürich.

There are two significant challenges associated with this setting. Firstly, we must ensure that a policy never visits unsafe states or performs unsafe actions. Clearly, it is not easy to guarantee safe policies for nondeterministic dynamics. We also need to know what states and actions are safe. Secondly, we must learn about the unknown quantity at states and actions we are not allowed to visit. Without any regularity assumption on the unknown quantity, this is not possible. Accordingly, we restrict the problem to Markov Chains with deterministic dynamics and a known and fixed safe state-action set. Moreover, we assume the unknown quantity has some regularity, a known kernel structure in our case.

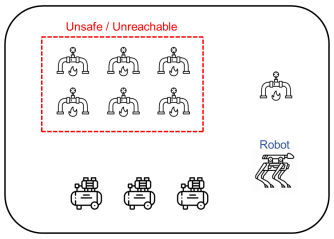


Figure 1: Illustrative Example: A robot measuring the gas concentration at an industrial site. There are pipes and compressors. The red area is restricted and contains six pipes.

We tackle the first challenge by introducing a safe state action density polytope and constraining optimization over it. We then leverage the known kernel structure in the objective to incentivize policies that estimate the unknown quantity both in safe and unsafe areas. Overall our contributions are the setting of Safe Active Exploration in MDPs, provably safe policies, and a reweighted optimization objective that considers the correlation between safe and unsafe states. Finally, we provide a theoretical convergence analysis and evaluate our findings in an experiment with an invariant correlation structure and one with a general correlation structure.

2 Preliminaries

2.1 Optimal Experimental Design

Optimal experimental design (OED) is concerned with the sequential design of experiments [5] where the unknown quantities are elements of a Hilbert space [9]. The unknown quantity is often a function f, which is linear in the parameters θ or features $\Phi(x)$ of experiments and is to be estimated using noisy observations. Given a limited budget of $n \in \mathbb{N}$, the goal of OED is to choose n experiments that minimize a particular objective, commonly the second moment of residuals $f - \hat{f}$. A brief introduction to the application of convex optimization can be found in [1]. The second moment of residuals is matrix-valued in general, and thus, one commonly uses scalarizations $s : \mathbb{S}_+ \to \mathbb{R}$, such as $s(\mathbf{\Sigma}) = -\log\det((\cdot)^{-1})$ or $s(\mathbf{\Sigma}) = Tr(\cdot)$, referred to as *D-optimal design* and *A-optimal design*, where $\mathbb{S}_+$ is the space of p.s.d. matrices [9].

2.2 Active Exploration

In optimal experimental design, we assume that the experiments are chosen from the set of possible experiments without restriction. Mutný et al. [7] and Tarbouriech et al. [11] extend experimental design to the setting where the structure of a given MDP constrains the choice of experiments. Tarbouriech et al. [11] consider independent observation distributions $\nu(s)$ per state $s \in \mathcal{S}$. In contrast, Mutný et al. [7] assume the unknown quantity to be a function f on state-action pairs possessing certain regularity, namely f belonging to a RKHS $\mathcal{H}_k$ with known kernel k.

2.3 Reproducing kernel Hilbert space (RKHS)

A RKHS is a Hilbert space where all point evaluations are bounded linear functionals. This means that for a Hilbert space $\mathcal{H}$ of functions on some domain $\mathcal{X}$, for all $x \in \mathcal{X}, f \in \mathcal{H}$, there exists $\Phi(x) \in \mathcal{H}$ such that $f(x) = f^T\Phi(x)$, where $\cdot^T\cdot$ denotes the inner product in $\mathcal{H}$ [12]. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a *kernel* if there exists a Hilbert space $\mathcal{H}$ and a map $\Phi : \mathcal{X} \to \mathcal{H}$ such that for all $x, x' \in \mathcal{X} \quad k(x, x') = \Phi(x)^T\Phi(x')$ [2]. Due to the Representer Theorem [6] we know that the minimizer of the regularized empirical risk functional $\hat{f} = \arg\min_{f\in\mathcal{H}} \sum_{i=1}^n (f^T\Phi(x) - y_i)^2 + \lambda\|f\|_{\mathcal{H}}$ can be represented as a finite linear combination of feature space mappings of points $\hat{f} = \sum_{i=1}^n \hat{\alpha}_i\Phi(x_i)$. The minimizer $\hat{\alpha}$ can be computed using the closed form solution of kernel ridge regression $\hat{\alpha} = (K + \lambda I)^{-1}y$, where $K_{ij} = \Phi(x_i)^T\Phi(x_j)$ [2].

2.4 Markov Decision Process (MDP)

An MDP [10] is a tuple $M = (\mathcal{S}, \mathcal{A}, P, H, r, s_0)$, where $\mathcal{S}$ is a state space, $\mathcal{A}$ is an action space, P denotes the transition model $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, such that $P(s'|s, a)$ is the probability to transition to state s' when choosing action a in state s, H is the episode horizon, r is a reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ where $r(s, a)$ is the reward obtained for playing action a in state s and s_0 is the initial state.
A policy π is a sequence of decision rules $(\pi_h)_{h=0}^H$ that map a state-action pair to probabilities over actions, i.e., $\pi_h : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{A})$. A policy π induces a distribution $\eta \in \Delta(\mathcal{T}_h)$ over trajectories where $\eta(\tau)$ is the probability of generating trajectory τ when following policy π and $\mathcal{T}_h$ is the set of trajectories $(s_h, a_h)_{h=0}^H$ of length h. A policy also induces a distribution over state-action pairs $d_\pi = \mathbb{E}_{\tau\sim\eta}[d]$, where $d(s, a) = \frac{1}{H} \sum_{(s_h,a_h)\in\tau} \mathbb{1}\{(s_h, a_h) = (s, a)\}$.

2.5 Convex Reinforcement Learning (RL)

Convex RL due to Hazan et al. [3] and Zahavy et al. [13] considers objectives $U(d)$, that are convex in the state-action distribution. It employs the Frank-Wolfe algorithm, which gradually builds a mixture policy π^* corresponding to the optimum $U(d^*)$. We give a quick review of the most important components of the algorithm:
Mixture Policy A mixture policy refers to a convex combination of policies. We denote this mixture policy as $\pi_{mix,n} = \{(a_i, \pi_i)\}_{i=0}^n$. A notable characteristic of mixture policies is that the induced state-action distribution follows a convex combination of the individual policies, i.e., $d_{\pi_{mix},n} = \sum_{i=0}^n a_i d_{\pi_i}$. Finally, a mixture policy can be represented by a single policy through the process of marginalization: $\pi(a \mid s) = \frac{d_{\pi_{mix}}(s,a)}{\sum_a d_{\pi_{mix}}(s,a)}$.

Density Estimation: Given an individual policy π_i, the density estimation oracle estimates the value of d_{π_i}. In the case of tabular setting, the d_{π_i} can be estimated easily by applying the transition operator K_{π_i}, see Appendix B.2. For known MPDs, the state action distribution can be estimated by simulating the policy π_i, and there is no need for interaction with the real environment.

Policy Search After estimating $d_{\pi_{mix},n}$ we can introduce an additional element π_{n+1} to the mixture policy by solving the following problem:

$$\pi_{n+1} = \arg\min_{\pi\in\Pi} \sum_{s,a} d_{\pi(s,a)} \nabla U(d_{\pi_{mix},n}(s, a)) \tag{1}$$

This is a classical reinforcement learning problem where the gradients of U serve as the reward function. As a result, it can be implemented via any of the exact solution methods, e.g., value iteration, policy iteration, and linear programming. The new policy's weight a_{n+1} can be calculated via line search. Finally, the new mixture policy is $\pi_{mix,n+1} = (1 - a_{n+1})\pi_{mix,i} \cup (a_{n+1}, \pi_{n+1})$.

3 Safe Active Exploration

Let $M = (\mathcal{S}, \mathcal{A}, P, H, f, s_0)$ be a known and deterministic MDP with episode length H. Our agent can operate in a known and fixed safe set $\mathcal{B}_{safe} \subset \mathcal{S}\times\mathcal{A}$ and is not allowed to enter $\mathcal{B}_{unsafe} \coloneqq \mathcal{B}_{safe}^{\mathsf{c}}$. There is a fixed budget of T trajectories of length H. An unknown quantity f, part of a RKHS $f \in \mathcal{H}_k$ with known kernel $k((s, a), (s', a')) = \Phi(s, a)^T \Phi(s', a')$ and bound $\|f\|_{\mathcal{H}_k} \leq \frac{1}{\lambda}$, is defined over $(s, a) \in \mathcal{S} \times \mathcal{A}$. We have access to noisy observations of the unknown quantity $y = f^T \Phi(s, a) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. The goal is to estimate a linear functional of the unknown function (potentially over the unsafe region):

$$\text{Find} \quad \mathbf{C}\hat{f}_T \quad \text{where:} \quad \hat{f}_T = \arg\min_{f\in\mathcal{H}_k} \sum_{i=1}^{T} \sum_{(s,a)\in\tau_i} (f^T\Phi(s, a) - y_{i,s,a})^2 + \lambda\|f\|_{\mathcal{H}_k}$$

4 Method

As typical in optimal experimental design, we minimize the second moment of the estimation error:

$$\mathbf{E}_T = \mathbb{E}_\epsilon[\mathbf{C}(f - \hat{f}_T)(f - \hat{f}_T)^T\mathbf{C}^T]$$

To ensure safety, we define a safe state-action distribution polytope.

$$\mathcal{D}^{safe} := \{d \,|d(s,a) \geq 0, \sum_{s,a} d(s,a) = 1, \sum_{a} d(s',a) = \sum_{s,a} d(s,a)P(s'|s,a), \\ \forall (s,a) \in \mathcal{B}_{unsafe}.\, d(s,a) = 0\}$$

Mutný et al. [7] derive a convex upper bound for the second moment of the estimation error. A first naive approach is to use this objective and simply add the safe state-action distribution polytope to the constraints:

$$\min_{d_\pi \in \mathcal{D}^{safe}} U(d_\pi) \coloneqq s\left(\mathbf{C}\left(\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \frac{H d_\pi(s,a)}{\sigma^2_{s,a}} \Phi(s,a)\Phi(s,a)^T + \frac{\lambda \mathbf{I}}{T}\right)^{-1} \mathbf{C}^T\right) \tag{2}$$

where $s(\cdot)$ is a convex scalarization function. However, note that all the features $\Phi(s,a)$ from the unsafe region cancel, since $d_\pi(s,a) = 0 \quad \forall (s,a) \in \mathcal{B}_{unsafe}$. Since Mutný et al. [7] usually set $\mathbf{C} = \mathbf{I}$, we lose the correlation information to the unsafe region. In the next two sections, we present two approaches to recover this correlation information in the objective.

4.1 Evaluation Functional Approach

The linear functional $\mathbf{C}$ is a matrix in the finite-dimensional case. We choose the rows of $\mathbf{C}$ to be the features of the state-action pairs that we want to evaluate, i.e., $(\mathbf{C})_i = \Phi(x_i)^T$, where $x_i = (s,a)_i$. Then, since f belongs to a RKHS, we have $(\mathbf{C}f)_i = \Phi(x_i)^T f = f(x_i)$. Thus, the diagonal entries of the matrix representing the second moment of residuals are

$$(\mathbf{C}(f - \hat{f}_T)(f - \hat{f}_T)^T\mathbf{C}^T)_{ii} = \left(f(x_i) - \hat{f}_T(x_i)\right)^2$$

and therefore, we have

$$\mathrm{Tr}\left[\mathbb{E}_\epsilon[\mathbf{C}(f - \hat{f}_T)(f - \hat{f}_T)^T\mathbf{C}^T]\right] = \mathbb{E}_\epsilon\left[\sum_{i=1}^{n}\left(f(x_i) - \hat{f}_T(x_i)\right)^2\right]$$

where we used linearity of expectation. By choosing $\mathbf{C}$ to be the evaluation functional, minimizing the trace of the second moment of residuals corresponds to minimizing the mean squared estimation error over the whole domain.

We then proceed by minimizing the upper bound of the trace of the second moment of the residuals, as defined in Equation 2:

$$\min_{d_\pi \in \mathcal{D}^{safe}} U(d_\pi)$$

This objective corresponds to A-optimal design, i.e., taking $s(\cdot) = \mathrm{Tr}(\cdot)$. Using this objective, we also include information about the unsafe region. See Appendix B.1 for an example aiming to provide intuition on how this information is included. Note that using the D-optimal design, i.e., taking $s(\cdot) = -\log\det((\cdot)^{-1})$, is not possible as soon as there exist state-action pairs $(s,a), (s',a')$ with $\Phi(s,a) = \Phi(s',a')$, since the matrix $\mathbf{C}$ then becomes singular and cannot be inverted.

4.2 Reweighting-Based Approach

Another option is to introduce a reweighting $w : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ to the objective. The idea is to put more weight on state-action pairs that are highly correlated with unsafe state-action pairs:

$$\min_{d_\pi \in \mathcal{D}^{safe}} U_w(d_\pi) \coloneqq s\left(\mathbf{C}\left(\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \frac{H d_\pi(s,a) w(s,a)}{\sigma^2_{s,a}} \Phi(s,a)\Phi(s,a)^T + \frac{\lambda \mathbf{I}}{T}\right)^{-1} \mathbf{C}^T\right) \tag{3}$$

The weights are defined as follows:

$$w(s,a) = \sum_{(s',a')\in\mathcal{B}_{unsafe}} \left|\frac{\Phi(s,a)^T\Phi(s',a')}{\sqrt{\Phi(s,a)^T\Phi(s,a)}\sqrt{\Phi(s',a')^T\Phi(s',a')}}\right| + 1 \tag{4}$$

We define the weight as the sum of the correlation of a state-action pair with all the state-action pairs in the unsafe state-action space plus one. If a state-action pair is entirely uncorrelated with the unsafe region, the weight is one, and we recover the naive approach. The reweighting transports information from the unsafe to the safe region. The higher the weight of a state-action pair, the higher its correlation to the unsafe region and the higher the final visitation.

4.3 Optimization

We can minimize the convex objective $\min_{d \in D_{safe}} U_w(d)$ over the polytope $d \in D_{safe}$ by using *Convex RL*. Our Algorithm 1 is essentially the same as Markov-Design by Mutný et al. [7]. However, there is a difference in the constraint set, and in the first step, we calculate the weighted objective as defined in Equation 3 and Equation 4.

We use the ADAPTIVE method proposed in Mutný et al. [7]. The key idea of this method is to incorporate information from previously executed trajectories to guide the selection of the next policy. This involves gradually estimating the empirical distribution of visited states $Z\eta_t = d_t$. So the new objective we optimize in each episode is $G_t(d) := U(Z\eta_t \frac{t}{t+1} + \frac{1}{t+1} d)$.

Algorithm 1 Safe Active Exploration

Require: known deterministic MDP, known fixed unsafe set $\mathcal{B}_{unsafe}$, Objective U, episodes T

$U_w = \textbf{Reweighting}(\mathcal{B}_{unsafe}, U)$ ▷ compute weighted objective
while $t \leq T$ **do**
  Convex RL: solving $\min_{d \in D^{safe}} G_t(d) := U_w(Z\eta_t \frac{t}{t+1} + \frac{1}{t+1} d)$
    $\pi_{mix,1} = \pi_t; i = 1; d_{\pi_{mix},1} = Z\eta_t$
    repeat
      $i = i + 1$
      $\pi_i = \arg\min_{\pi \in \Pi} \sum_{s,a} d_\pi(s,a) \nabla_{s,a} G_t(d_{\pi_{mix},i}(s,a))$ ▷ RL problem
      $d_{\pi_i} = \textbf{Density Estimation}(\pi_i)$ ▷ keep track of visitations
      $a_i = \arg\min_{a \in \mathbb{R}} G_t((1-a) d_{\pi_{mix},i} + a d_{\pi_i})$ ▷ line search
      $\pi_{mix,i+1} = (1-a_i)\pi_{mix,i} \cup (a_i, \pi_i)$ ▷ update mixture policy
      Update $d_{\pi_{mix},i+1} = (1-a_i) d_{\pi_{mix},i} + a_i d_{\pi_i}$ ▷ update mixture policy visitations
    until convergence
    $\pi_t = \textbf{Marginalize}\ \pi_{mix}$
  Interaction
    Sample trajectory from $\tau_t \sim \pi_t$ (also as $\delta_{\tau_t} \sim q_t$)
    $Z\eta_{t+1} = Z\frac{t}{t+1}\eta_t + Z\frac{1}{t+1}\delta_t$ ▷ keep track of visited states
end while

5 Convergence Theory

Since our optimization scheme is essentially the same as in Mutný et al. [7], we base our convergence results on their work. The evaluation functional approach in subsection 4.1 is already covered by the theory introduced by Mutný et al. [7]. Accordingly, we only need to show convergence for the reweighted objective. We first state a basic assumption on the naive objective without reweighting, which follows directly from Assumption 1 in Mutný et al. [7].

Assumption 1 *Let $U : \mathcal{D}^{safe} \to \mathbb{R}$ be convex, differentiable, locally Lipschitz continuous in $\|\cdot\|_\infty$, and locally smooth as:*

$$U(d') \leq U(d) + \nabla U(d)^\top (d' - d) + \frac{L'}{2} \|d' - d\|_2^2 \tag{5}$$

For $d', d \in \mathcal{D}^{safe}$ and global smoothness: $L = \max L'$

Now we show a relation between the smoothness of the reweighted and naive objective.

Proposition 1 *Let $U_w : \mathcal{D}^{safe} \to \mathbb{R}$ be the reweighted objective. The objective is convex and differentiable. With Design A scalarization and invariant features Φ the objective is also locally*

Lipschitz continuous in $\|\cdot\|_\infty$, and locally smooth as:

$$U_w(d') \leq U_w(d) + \nabla U_w(d)^\top (d' - d) + \frac{L'_w}{2}\|d' - d\|_2^2 \tag{6}$$

For $d', d \in \mathcal{D}^{safe}$ and with $L'_w = cL'$ where $c \in (0, 1]$. Also let $L_w = \max L'_w$

The reweighted objective has a smaller or equal smoothness constant than the naive objective without reweighting. For the case where the weights are all one i.e., the unweighted case, we have $c = 1$. Otherwise, higher weights result in a smaller c. This follows directly from the proof of Proposition 1, which we postpone to Appendix A.2.

Note that for η, a distribution over trajectories, Mutný et al. [7] show $U(d) = U(\mathbf{Z}\eta) = F(\eta)$. One can also easily show that $U_w(d) = U_w(\mathbf{Z}\eta) = F_w(\eta)$. We state the convergence in terms of $F_w(\eta), F_w(\eta^*)$ but it can equally be stated in terms of $U_w(d), U_w(d^*)$.

5.1 High Probability Convergence

From Proposition 1 and using the previously introduced relation, we know that $F_w(\eta)$ is smooth with global smoothness constant $L_w = cL$, where $c \in (0, 1]$. However, Mutný et al. [7] point out that Experimental Design objectives can have global smoothness $L = \frac{T}{\lambda}$. Thus global smoothness of L_w is not sufficient for convergence. One way to mitigate this is to use smoothing due to Nesterov [8]. The smoothed objective $F_{w,\mu}$ has global smoothness $L_{w,\mu} = \frac{L_w}{1+\mu L_w} \leq \frac{1}{\mu}$. Using smoothing, we can show the following convergence bound:

Theorem 1 *Under Proposition 1, with the smoothed objective using $\mu = \sqrt{\log T}/T$:*

$$F_w(\eta_T) - F_w(\eta^*) \leq \mathcal{O}\left(\frac{1}{T}\sqrt{\sum_{t=0}^{T} \|\nabla F(\eta_t)\|_\infty^2 \log(T/\delta))}\right) \tag{7}$$

With probability $1 - \delta$ over transition model and policy.

The convergence is the same as in Mutný et al. [7]. It is no surprise that we get the same convergence result here. The main difference between $F(\eta)$ and $F_w(\eta)$ is in the local and global smoothness constants. Since we use Nesterov smoothing [8], we bound away any dependency on the original smoothness constants. For a detailed proof, visit Theorem 1 in the appendix of Mutný et al. [7] and exchange L_μ with $L_{w,\mu}$. As a result, we can also relax Proposition 1 and not require any smoothness of the objective. Thus Theorem 1 holds for all scalarization designs and general correlation structures. Overall with a reasonable upper bound on the gradient of F, the convergence is $\mathcal{O}(1/\sqrt{T})$ even for non-smooth objectives.

5.2 Convergence in Expectation

From Proposition 1, we known that $F_w(\eta)$ is smooth with local smoothness constant $L'_w = cL'$, where $c \in (0, 1]$. Mutný et al. [7] conjecture that the local smoothness L' increases with $\mathcal{O}(\log T)$. Using the local smoothness, we can also show convergence in expectation:

Theorem 2 *Under Proposition 1:*

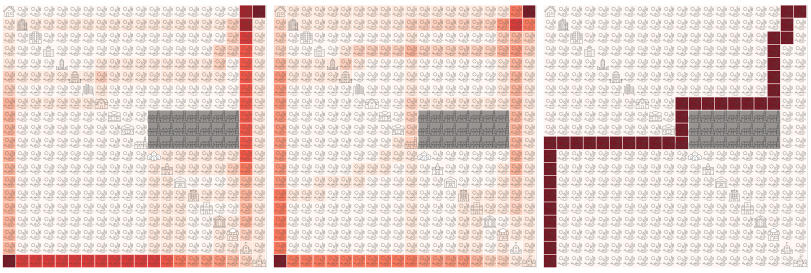
$$\mathbb{E}\left[F_w(\eta_T)\right] - F_w(\eta^*) \leq \mathcal{O}\left(\frac{c}{T}\sum_{k=0}^{T} \frac{L_k}{1+k}\right) \tag{8}$$

Where $c \in (0, 1]$

For a detailed proof, visit Theorem 4 in the appendix of Mutný et al. [7] and exchange the local smoothness $L_{\eta_k,1/k}$ with $L_{w,\eta_k,1/k} = cL_{\eta_k,1/k} = cL_k$. This convergence result is again very similar to the one in Mutný et al. [7]. The only difference is the factor $c \in (0, 1]$. The proof for Proposition 1 shows that the factor depends on the magnitude of the weights $w(s, a)$ in the objective. If there is no reweighting, we have $c = 1$, and the convergence is the same as in Mutný et al. [7]. Otherwise, the higher the weights, the smaller c, and the faster the convergence relative to Mutný et al. [7]. Overall the convergence is $\mathcal{O}(\log T/T)$, given that the local smoothness decreases with $\mathcal{O}(\log T)$.

6 Experiments

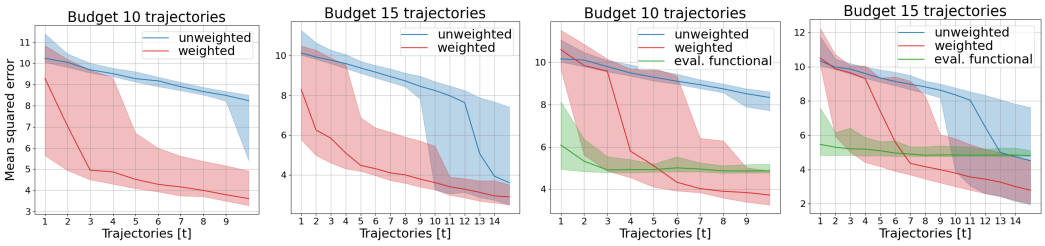
6.1 Invariant correlation - Synthetic grid world



(a) D-Design: **unweighted** objective (b) D-Design: **weighted** objective (c) A-Design: evaluation functional

Figure 2: Using the **unweighted** objective 2a, the agent does not take into account the unsafe region and uniformly visits different sectors in the diagonal. None of the trajectories go through the factory, correlated with the unsafe region. The **weighted** objective 2b considers the unsafe region and visits the safe factory. The objective with the **evaluation functional** 2c puts too much weight on the safe factory, which leads all trajectories visiting this state.

Consider the synthetic grid world Fig.2 with dimensions 20x20, where the initial state is the bottom left corner and the final is the top right corner. The environment is non-ergodic; the possible actions in each state are to move up or right. The features $\Phi(s, a)$ are defined over states. Different pictograms represent different sectors. As a result, states in the same sector have the same features and are invariant. States in different sectors are independent. We define the unknown target f to be a constant function. Safety is also defined over states, and the unsafe states can be observed with shaded grey in Fig.2. In our experiment setting, we have twenty different sectors in the diagonal. The agent can only pass through one diagonal sector in every trajectory because the environment is non-ergodic. In Fig.2, we present the heatmap after ten trajectories for the D-design of the unweighted and weighted objectives. As the evaluation function cannot be implemented for the D-design, we did such experiments only for the A-design; see Fig.2. In Appendix B.3, we cite heatmaps of the other experiments with A-design. To calculate $\hat{f}$, we simulate the observations and then perform a kernelized regression. After every run, we compute the MSE. We present the results in Fig.3. The figures do not include the initial MSE; see Appendix B.4. We conclude that both the unweighted objective and the evaluation function lead to sub-optimal solutions. The unweighted objective does not take into account the unsafe region, and the evaluation function puts too much weight on the safe factory state. The weighted objective incentivizes the visitation of the safe factory state and estimates the whole domain more consistently and efficiently.

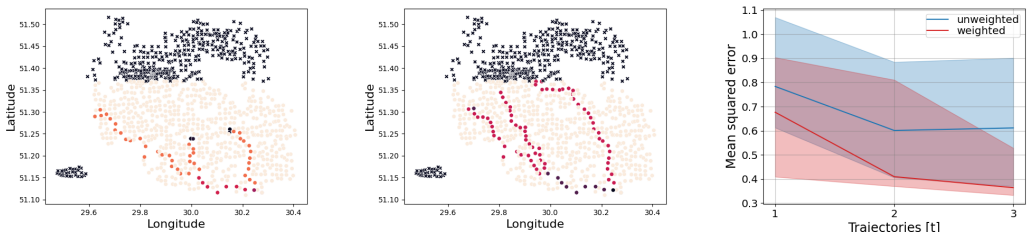


(a) **D-Design** (b) **D-Design** (c) **A-Design** (d) **A-Design**

Figure 3: The median MSE of 20 reruns with 10% and 90% quantiles. On the left, one can see the results of the **D-Design** for 10 and 15 trajectories. The weighted objective performs significantly better for ten trajectories than the unweighted one. For 15 trajectories, the MSE of the unweighted objective drops as the probability of sampling the safe factory increases. On the right, we see the results of the **A-Design**. In green, we observe the MSE of the evaluation functional. As it only samples from the safe factory state, we see a drop from the first trajectory, but it does not converge.

6.2 General correlation - Chornobyl Exclusion Zone

To test our method on an example with non-invariant kernels, we used a spatial data set containing data on radionuclide contamination in the Ukrainian Chornobyl Exclusion Zone [4]. In particular, we use data that includes measurements of radiocaesium, radiostrontium, and soil chemistry parameters. For the experiment, we assume access to all measurements except the concentration of radiostrontium in the soil, which we want to estimate. This represents a scenario where data from previous years is already available, but a new quantity previously not measured is to be estimated efficiently. To simulate data collection, we created a graph containing a node for every measurement site with edges between geographically nearest neighbors. The MDP corresponding to this graph has a state for every node and actions to transition to a chosen neighbor deterministically. Upon arrival in a state, the agent receives a noisy observation of the concentration of radiostrontium at this location. Safety or inaccessibility in this scenario can be related to a known, geographically limited safety hazard or simply to the absence of permission to enter certain areas. For this experiment, we classified two arbitrarily selected regions as unsafe, making 419 of the 1074 states inaccessible. We chose the number of trajectories to be $T = 3$, with a horizon of $H = 30$. As features, we used all 14 measured variables and a constant feature normalized to be between 0 and 1. The variance of the Gaussian observation noise is 0.1, and the regularization parameter for the kernel ridge regression is 0.1. The results of the experiment can be seen in Fig. 4. One can notice that trajectories resulting from the weighted objective visit more states to the north, where many inaccessible states are located. We achieve a lower mean squared error on average using the weighted objective, although the variance for both methods is quite large.



(a) (b) (c)

Figure 4: a) Heatmap of one run using the **unweighted** objective. The unsafe states are marked with a cross and colored dark. b) Heatmap of one run using the **weighted** objective. c) Comparison of the mean squared error of the estimated function values after each executed trajectory. We report the median of 10 runs with 10% and 90% quantiles.

7 Conclusion

In this work, we introduced a new and highly relevant setting of safety-critical experimental design. We defined a safe state-action density polytope that guarantees safe policies. We propose a new reweighted experimental design objective that leverages the correlation of the unknown function between unsafe and safe regions. We performed a theoretical analysis of the convergence of this new objective. Finally, we validate the performance of the reweighted objective experimentally. We designed a synthetic experiment with an invariant correlation structure and an experiment with real data, and a general correlation structure. One limitation we notice is a high variance for both the unweighted and weighted objectives. In the future, it may be worthwhile to investigate this and additionally consider the setting of an adaptive safe region.

References

[1] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. doi: 10.1017/CBO9780511804441.

[2] Arthur Gretton. Introduction to rkhs. `http://mlss.tuebingen.mpg.de/2015/slides/gretton/part_1.pdf`, July 2015.

[3] Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. 2019. URL `https://arxiv.org/pdf/1812.02690.pdf`.

[4] V. Kashparov, S. Levchuk, M. Zhurba, V. Protsak, Yu. Khomutinin, N.A. Beresford, and J.S. Chaplow. Spatial datasets of radionuclide contamination in the ukrainian chernobyl exclusion zone, 2017. URL `https://doi.org/10.5285/782ec845-2135-4698-8881-b38823e533bf`.

[5] J. Kiefer. Optimum experimental designs. *Journal of the Royal Statistical Society. Series B (Methodological)*, 21(2):272–319, 1959. ISSN 00359246. URL `http://www.jstor.org/stable/2983802`.

[6] George Kimeldorf and Grace Wahba. Some results on tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33(1):82–95, 1971.

[7] Mojmir Mutny, Tadeusz Janik, and Andreas Krause. Active exploration via experiment design in markov chains. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 7349–7374. PMLR, 25–27 Apr 2023. URL `https://proceedings.mlr.press/v206/mutny23a.html`.

[8] Yu. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, May 2005. ISSN 1436-4646. doi: 10.1007/s10107-004-0552-5. URL `https://doi.org/10.1007/s10107-004-0552-5`.

[9] Friedrich Pukelsheim. *Optimal Design of Experiments*. Society for Industrial and Applied Mathematics, 2006. doi: 10.1137/1.9780898719109. URL `https://epubs.siam.org/doi/abs/10.1137/1.9780898719109`.

[10] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[11] Jean Tarbouriech and Alessandro Lazaric. Active exploration in markov decision processes, 2019.

[12] Grace Wahba. An introduction to reproducing kernel hilbert spaces and why they are so useful. *IFAC Proceedings Volumes*, 36(16):525–528, 2003. ISSN 1474-6670. doi: https://doi.org/10.1016/S1474-6670(17)34815-2. URL `https://www.sciencedirect.com/science/article/pii/S1474667017348152`. 13th IFAC Symposium on System Identification (SYSID 2003), Rotterdam, The Netherlands, 27-29 August, 2003.

[13] Tom Zahavy, Brendan O'Donoghue, Guillaume Desjardins, and Satinder Singh. Reward is enough for convex mdps. *CoRR*, abs/2106.00661, 2021. URL `https://arxiv.org/abs/2106.00661`.

A Proofs

A.1 Lemmas

Lemma 1 For $w(s,a) \geq 1$, $d_\pi(s,a) \geq 0$, $\Phi(s,a) = \mathbf{e}_i$, where $\mathbf{e}_i$ is a unit vector in $\mathbb{R}^d$ and $\mathcal{B}_{\text{feature}}(i) = \{(s,a) \in \mathcal{S} \times \mathcal{A} : \Phi(s,a) = \mathbf{e}_i\}$ it holds:

$$
\begin{aligned}
&\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \frac{Hd_\pi(s,a)w(s,a)}{\sigma^2_{s,a}}\Phi(s,a)\Phi(s,a)^T + \frac{\lambda\mathbf{I}}{T} \\
&\qquad = \text{diag}\begin{bmatrix} \sum_{(s,a)\in\mathcal{B}_{\text{fwas set toeature}}(1)} \frac{Hd_\pi(s,a)w(s,a)}{\sigma^2_{s,a}} + \frac{\lambda}{T} \\ \vdots \\ \sum_{(s,a)\in\mathcal{B}_{\text{feature}}(d)} \frac{Hd_\pi(s,a)w(s,a)}{\sigma^2_{s,a}} + \frac{\lambda}{T} \end{bmatrix} \\
&\qquad = \text{diag}\begin{bmatrix} c_1 \sum_{(s,a)\in\mathcal{B}_{\text{feature}}(1)} \frac{Hd_\pi(s,a)}{\sigma^2_{s,a}} + \frac{\lambda}{T} \\ \vdots \\ c_d \sum_{(s,a)\in\mathcal{B}_{\text{feature}}(d)} \frac{Hd_\pi(s,a)}{\sigma^2_{s,a}} + \frac{\lambda}{T} \end{bmatrix} \\
&\qquad = \text{diag}\begin{bmatrix} c_1 \\ \vdots \\ c_d \end{bmatrix} \text{diag}\begin{bmatrix} \sum_{(s,a)\in\mathcal{B}_{\text{feature}}(1)} \frac{Hd_\pi(s,a)}{\sigma^2_{s,a}} + \frac{\lambda}{T} \\ \vdots \\ \sum_{(s,a)\in\mathcal{B}_{\text{feature}}(d)} \frac{Hd_\pi(s,a)}{\sigma^2_{s,a}} + \frac{\lambda}{T} \end{bmatrix} \\
&\qquad = \mathbf{C} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \frac{Hd_\pi(s,a)}{\sigma^2_{s,a}}\Phi(s,a)\Phi(s,a)^T + \frac{\lambda\mathbf{I}}{T}
\end{aligned}
$$

Importantly we have $c_1 \dots c_d \geq 1$. It is easy to see that if all our weights are 1, we have $c_1 \dots c_d = 1$, and the more large weights exist, the bigger $c_1 \dots c_d$.

A.2 Reweighted Smoothness

Proposition 1 Let $U_w : \mathcal{D}^{safe} \to \mathbb{R}$ be the reweighted objective. The objective is convex, and differentiable. With Design A scalarization and invariant features Φ the objective is also locally Lipschitz continuous in $\|\cdot\|_\infty$, and locally smooth as:

$$
U_w(d') \leq U_w(d) + \nabla U_w(d)^\top (d' - d) + \frac{L'_w}{2}\|d' - d\|_2^2 \tag{9}
$$

For $d', d \in \mathcal{D}^{safe}$ and with $L'_w = cL'$ where $c \in (0,1]$. Also let $L_w = \max L'_w$

Proof. It is easy to see that the reweighted objective is differentiable and convex. Now let us first recall the smoothness condition of the unweighted objective from Assumption 1:

$$
U(d') \leq U(d) + \nabla U(d)^\top (d' - d) + \frac{L'}{2}\|d' - d\|_2^2 \tag{10}
$$

We now show that $U_w(d) = c \cdot U(d) \quad \text{with} \quad c \in (0,1]$

$$
\begin{aligned}
U_w(d) &= \text{Tr}\left(\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \frac{Hd_\pi(s,a)w(s,a)}{\sigma^2_{s,a}}\Phi(s,a)\Phi(s,a)^T + \frac{\lambda\mathbf{I}}{T}\right)^{-1} \\
&\overset{\text{Lemma 1}}{=} \text{Tr}\left(\mathbf{C}\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \frac{Hd_\pi(s,a)}{\sigma^2_{s,a}}\Phi(s,a)\Phi(s,a)^T + \frac{\lambda\mathbf{I}}{T}\right)^{-1} \\
&= c \cdot \text{Tr}\left(\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \frac{Hd_\pi(s,a)}{\sigma^2_{s,a}}\Phi(s,a)\Phi(s,a)^T + \frac{\lambda\mathbf{I}}{T}\right)^{-1} \\
&= c \cdot U(d)
\end{aligned}
$$

Where $c \in (0, 1]$. From Lemma 1, we see that $c = 1$ if we have no reweighting i.e., $w(s, a) = 1 \forall (s, a) \in \mathcal{S} \times \mathcal{A}$. Otherwise, the more large weights, the smaller c. Now we insert the expression into relation 10 and get the final result:

$$\frac{1}{c} U_w(d') \leq \frac{1}{c} U_w(d) + \frac{1}{c} \nabla U_w(d)^\top (d' - d) + \frac{L'}{2} \|d' - d\|_2^2$$
$$U_w(d') \leq U_w(d) + \nabla U_w(d)^\top (d' - d) + \frac{cL'}{2} \|d' - d\|_2^2$$

B Additional Material

B.1 Evaluation Functional Approach - Example

For simplicity, consider a setting where the features are defined over states, and the state space consists of three disjoint sets $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$, with $\Phi(s) = \mathbf{e}_i$ the i-th unit vector for $s \in \mathcal{S}_i$. Then, for every $s \in \mathcal{S}_i$, $\Phi(s)\Phi(s)^T$ is a matrix with 1 at position ii and 0 everywhere else. We thus have

$$\sum_{s \in \mathcal{S}} d_\pi(s) \Phi(s) \Phi(s)^T + \lambda \mathbf{I} = \begin{bmatrix} \sum_{s \in \mathcal{S}_1} d_\pi(s) + \lambda & 0 & 0 \\ 0 & \sum_{s \in \mathcal{S}_2} d_\pi(s) + \lambda & 0 \\ 0 & 0 & \sum_{s \in \mathcal{S}_3} d_\pi(s) + \lambda \end{bmatrix}$$

And since this matrix is diagonal, we have

$$\left(\sum_{s \in \mathcal{S}} d_\pi(s) \Phi(s) \Phi(s)^T + \lambda \mathbf{I} \right)^{-1} = \begin{bmatrix} \frac{1}{\sum_{s \in \mathcal{S}_1} d_\pi(s) + \lambda} & 0 & 0 \\ 0 & \frac{1}{\sum_{s \in \mathcal{S}_2} d_\pi(s) + \lambda} & 0 \\ 0 & 0 & \frac{1}{\sum_{s \in \mathcal{S}_3} d_\pi(s) + \lambda} \end{bmatrix}$$

If we take $\mathbf{C}$ to be the evaluation functional over the whole domain $\mathcal{S}$, the matrix $\mathbf{C}$ will contain the feature $\mathbf{e}_i$ exactly $|\mathcal{S}_i|$ times. Using that the trace is invariant under cyclic permutations, we have

$$Tr \left(\mathbf{C} \left(\sum_{s \in \mathcal{S}} d_\pi(s) \Phi(s) \Phi(s)^T + \lambda \mathbf{I} \right)^{-1} \mathbf{C}^T \right) = Tr \left(\mathbf{C}^T \mathbf{C} \left(\sum_{s \in \mathcal{S}} d_\pi(s) \Phi(s) \Phi(s)^T + \lambda \mathbf{I} \right)^{-1} \right)$$

We have that

$$\mathbf{C}^T \mathbf{C} = \begin{bmatrix} |\mathcal{S}_1| & 0 & 0 \\ 0 & |\mathcal{S}_2| & 0 \\ 0 & 0 & |\mathcal{S}_3| \end{bmatrix}$$

Therefore

$$Tr \left(\mathbf{C} \left(\sum_{s \in \mathcal{S}} d_\pi(s) \Phi(s) \Phi(s)^T + \lambda \mathbf{I} \right)^{-1} \mathbf{C}^T \right) = \sum_{i=1}^{3} \frac{|\mathcal{S}_i|}{\sum_{s \in \mathcal{S}_i} d_\pi(s) + \lambda}$$

Now, assuming we have given a safe region $\mathcal{B}_{safe}$, we have the constraint $d_\pi(s) = 0$ for all $s \in \mathcal{B}_{unsafe} \coloneqq \mathcal{B}_{safe}^{\mathsf{c}}$. Therefore the objective is equivalent to

$$\sum_{i=1}^{3} \frac{|\mathcal{S}_i|}{\sum_{s \in \mathcal{S}_i \cap \mathcal{B}_{safe}} d_\pi(s) + \lambda}$$

Since we minimize this quantity, $d_\pi(s)$ should be bigger where the cardinality of the corresponding set is large, and fewer states of the same set are in the safe region.

B.2 Transition Operator

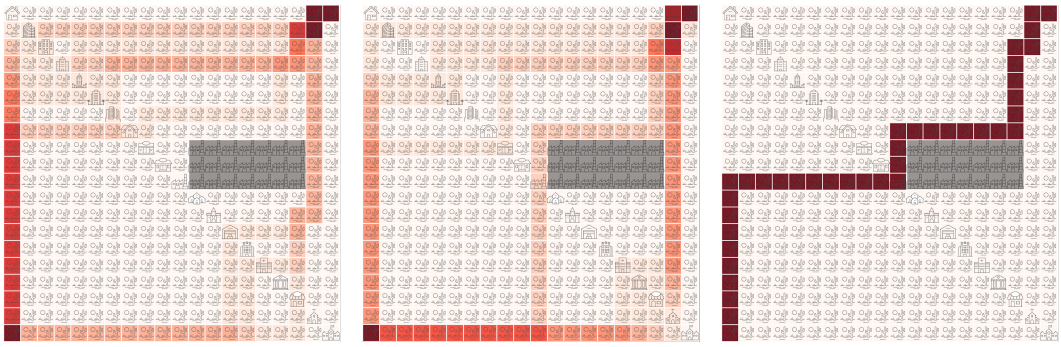
The transition operator is defined as:

$$K_\pi(x, x') := \sum_a P(x'|a, x)\pi_h(a|x)$$

We get the state-action density by applying:

$$d_h(s, a) = \left(\prod_{i=1}^{h-1} K_{\pi_i} d_0(s) \right) \pi_h(s|a)$$

B.3 A-Design



(a) A-Design: **unweighted** objective (b) A-Design: **weighted** objective (c) A-Design: evaluation functional

Figure 5: Using the **unweighted** objective 5a, the agent does not take into account the unsafe region and, for this run, does not samples from the safe factory. With the **weighted** objective 5b, the agent samples from the safe factory and reduces the overall MSE. The reweighting with the **evaluation functional** 5c puts too much weight on the safe factory and leads to a sub-optimal solution.

B.4 MSE log scale figures

As the initial MSE is too large, we excluded it from Fig.3 for better visualizations. Here we present the complete figures in log scale.

(a) **D-Design** (b) **D-Design** (c) **A-Design** (d) **A-Design**

Figure 6