

---

# Active Exploration via Experiment Design in Markov Chains

---

**Mojmír Mutný**

mojmir.mutny@inf.ethz.ch  
ETH Zürich

**Tadeusz Janik**

tjanik@student.ethz.ch  
ETH Zürich

**Andreas Krause**

krausea@ethz.ch  
ETH Zürich

## Abstract

A key challenge in science and engineering is to design experiments to learn about some unknown quantity of interest. Classical experimental design optimally allocates the experimental budget into measurements to maximize a notion of utility (e.g., reduction in uncertainty about the unknown quantity). We consider a rich setting, where the experiments are associated with states in a *Markov chain*, and we can only choose them by selecting a *policy* controlling the state transitions. This problem captures important applications, from exploration in reinforcement learning to spatial monitoring tasks. We propose an algorithm – MARKOV-DESIGN – that efficiently selects policies whose measurement allocation *provably converges to the optimal one*. The algorithm is sequential in nature, adapting its choice of policies (experiments) using past measurements. In addition to our theoretical analysis, we demonstrate our framework on applications in ecological surveillance and pharmacology.

## 1 INTRODUCTION

The optimal design of experiments (Pukelsheim, 2006; Chaloner and Verdinelli, 1995) is a ubiquitous challenge in science and engineering. The key goal is to utilize the limited budget (time, resources) to gain as much information about some unknown quantity of interest. Classical experiment design assumes that an experiment measures a single value with specific conditions. Motivated by applications illustrated in more detail below, we assume that experiments are associated with *policies* executed in a *known Markov chain*. We seek to find the best selection of policies that allow us to extract as much information from the environment as our budget allows. There are three major challenges in finding the set of best policies. Firstly, the space of policies can be *combinatorial* in

the size of the state-action space, and searching over it directly would lead to intractable optimization problems with classical methods. Secondly, the feedback from a policy is *stochastic*, due to the randomness of the Markov chain, which needs to be taken into account. Lastly, it is unclear how classical experimental design *objectives* can be formulated as a function of the policies. We address all these challenges with convex optimization techniques and utilize the structure of Markov decision processes.

**Quantity of interest** We want to estimate some aspects of an unknown quantity  $f$ . Here, we assume  $f$  is a function of states and actions of a Markov chain  $f : \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R}$ . We further assume  $f$  possess certain regularity, in particular, belongs to a reproducing kernel Hilbert space (RKHS) ( $f \in \mathcal{H}_k$ ) with a *known kernel*  $k((x, a), (x', a')) = \Phi(x, a)^\top \Phi(x', a')$ <sup>1</sup>, where  $x, x' \in \mathcal{X}$  and  $a, a' \in \mathcal{A}$ , and a *known bound*  $\|f\|_{\mathcal{H}_k} \leq \frac{1}{\lambda}$ . As a concrete example, the function  $f$  can model the distribution of species over a certain geographical location, where the kernel incorporates spatial features such as access to water, soil salinity, etc., and  $x$  corresponds to the spatial location. While  $f$  is unknown altogether, we are often interested in estimating only a *linear functional* of it  $\mathbf{C}f$ , where  $\mathbf{C} : \mathcal{H}_k \rightarrow \mathbb{R}^p$ . In the context of biological surveillance, this can be, e.g., a spatial average over certain locations. Another example are values at specific locations, where the rows of  $\mathbf{C}_i = \Phi(x_i, \cdot)^\top$ , with  $\{x_i\}_i$  being the locations of interest. For simpler models (finite-dimensional  $f$ ),  $\mathbf{C}$  may even be the identity (i.e., the goal is to estimate  $f$  completely). We observe  $f$  via noisy evaluations at specific states  $x$  while performing an action  $a$ ,

$$y = f^\top \Phi(x, a) + \epsilon \quad (1)$$

where  $\epsilon$  is random noise realization such that  $\mathbb{E}[\epsilon] = 0$ .

**Exploration** Consider an example, where we want to deploy an agent (e.g., a drone) that explores the environment efficiently to learn  $\mathbf{C}f$  from observations at  $(x, a)$ . Due to the kernel regularity assumption (RKHS), we know that similar  $(x, a)$  lead to similar values of  $f$ , and to understand  $\mathbf{C}f$ , intuitively, we should explore diverse landscape features instead of evaluating similar  $(x, a)$ . In the species distribution example, these are illustrated via the pictograms in Fig. 1a and

<sup>1</sup>.<sup>†</sup> denotes the Hilbert space inner product.

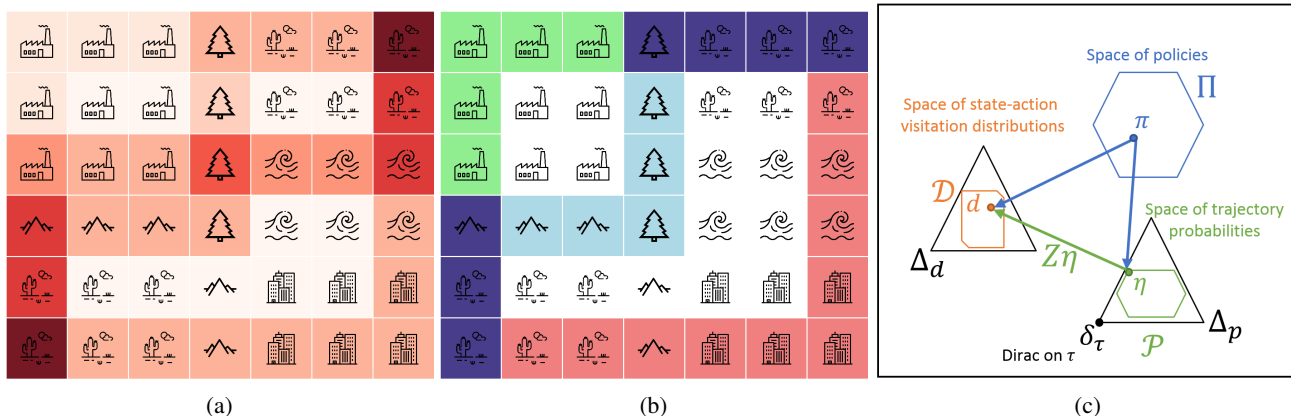


Figure 1: a) Optimal state-action visitation distribution that maximizes the information about species distribution as a function of state features (indicated via pictograms). Different pictograms represent sectors where the unknown function  $f$  has different values. Note that some sectors are not visited by the optimal policy, as the information about the function  $f$  in these sectors is already extracted from different states with the same pictogram. In essence, showing that the correlation of the model is utilized. b) Displayed are *three trajectories* from *three different policies* in *three episodes* using three different colors. The average of the visitations is converging to the optimal state-action distribution depicted in a). Note that the trajectories move from the initial (lower left) to the terminal state (upper right) and overlap in the purple-colored states. c) This figure provides a diagram of the relation between the sets we work with. Given a policy  $\pi$ , we can calculate state-action visitations and probabilities over trajectories it generates. Also, state-action visitations  $d$  and probability over trajectories  $\eta$  are related via a linear *conversion map*  $\mathbf{Z}$ . A single executed trajectory defines a Dirac delta on the probability space  $\mathcal{P}$ ,  $\delta_\tau$ .

Fig. 1b. We would like to pick a small subset of states covering all pictograms there, however, we cannot choose states arbitrarily: the only way we can choose them is by following the Markov transition rule, which then generates trajectories as in Fig. 1b. In fact, to learn  $\mathbf{C}f$  optimally, we need to visit the states in proportion to the heat map in Fig. 1a - *optimal visitation of the states*. In this work, we develop a method that picks policies sequentially such that their trajectories (as in Fig. 1b) converge to the optimal visitation distribution over states and actions as in Fig. 1a, detailed below.

**Contribution** We design a *novel* algorithm: MARKOV-DESIGN for adaptive experiment design, where we collect data by deploying policies in a *known* Markov chain for a fixed horizon. We notice that classical *experimental design* objectives can be reformulated such that they depend on state-action visitations and show how they can be efficiently optimized using *convex reward reinforcement learning* which reduces to a sequence of classical planning problems. The resulting algorithm relies on convex optimization techniques and is simple to implement. We prove that our algorithm converges to the optimal allocation over trajectories generated by selected policies and propose several variants of the algorithm with varying levels of approximation and analyze their performance. Additionally, we demonstrate our algorithm on two real-world experimental design problems with Markov chain structure.

## 2 BACKGROUND AND RELATED WORK

**Environment** We assume a known set of environment states  $\mathcal{X}$  and actions  $\mathcal{A}$ , and a known Markov transition

model  $p(x'|x, a) : \mathcal{X} \times \mathcal{A} \rightarrow \mathcal{X}$ . We interact with the environment by executing a *policy*, where actions are sampled  $a_h \sim \pi_h(a|x_h)$ , for horizon  $H$ , resulting in trajectories  $\tau = \{x_0, a_0, x_1, a_1, \dots, x_{H-1}, a_{H-1}\}$ , where  $\tau \in \mathcal{T}$  is the set of all possible trajectories with horizon  $H$ . The space of *all policies* is denoted  $\Pi$ . A policy induces a probability distribution over trajectories  $\eta$ , belonging to the feasible distribution set  $\mathcal{P} \subseteq \Delta_p$ , a subset of all possible distributions over trajectories. A distribution supported only on one trajectory  $\tau$  is denoted  $\delta_\tau$ . As in Fig. 1c, note that a policy  $\pi$  is related to a specific  $\eta \in \mathcal{P}$ , but it is generally challenging if at all possible to associate an arbitrary distribution over trajectories to a policy.

### 2.1 Related Work

**Experiment Design** Sequential design of experiments has a very rich history spanning back to Fisher (1926) and Kiefer (1959). In this work, we focus on a specific case where the unknown elements are members of a Hilbert space, usually referred to as optimal experimental design (OED) (Pukelsheim, 2006). Modern applications of OED range from protein design (Romero et al., 2013), Poisson sensing (Mutný and Krause, 2022b) to machine calibration Kirschner et al. (2019) among many others. It is closely related to active learning (Settles, 2009) and bandit optimization (Szepesvari and Lattimore, 2020). It can be studied with the frequentist assumption as done in this work, or within a Bayesian framework (cf., Chaloner and Verdinelli, 1995). Traditionally, in OED, the action (experiment) can be chosen deterministically, and the space of actions is not large – a scenario which we cover when the Markov chain

is *trivial*. Scaling to many actions has been addressed in the bandit literature in the context of *combinatorial bandits* (Zou et al., 2014; Talebi et al., 2013; Jourdan et al., 2021), where the action structures are, e.g., paths in a graph, or spanning trees. While the goal is optimization in their line of work (i.e., finding  $\arg \max_x f(x)$ ), the techniques used in these works are similar in spirit to the ones used here. They circumvent the large action set problem by having an efficient oracle that incrementally increases the allocation, e.g., the Dijkstra algorithm. This incremental increase slowly converges to the optimal allocation. Similarly, our problem – experiment design over policies – also spans a large set of actions. We utilize the Markov chain structure to design a sequence of Markov policies to incrementally change the visitations of states and actions. The oracle we rely upon is the Bellman optimality solver – planning.

**Convex RL** Our approach utilizes the recent framework of reinforcement learning with convex reward functions due to Hazan et al. (2019) and Zahavy et al. (2021). These works propose a way to find a stationary policy  $\pi^*$  which minimizes a certain convex reward function, which, in light of our work, can be uncertainty about  $f$ . Similarly to this work, exploration with convex rewards in the context of MDPs is investigated by Tarbouriech and Lazaric (2019) in a non-episodic context with ergodicity assumption. Different from their work, we consider more complex objectives which incorporate the structure of RKHS – similarity between states, exploiting a connection with the field of experiment design. In addition, we do not need to assume ergodicity. Note that a large body of works, addresses exploration in Markov chains in order to learn the parameters of the Markov chain (cf. Sato et al., 1981; Duff, 2002). However, this is not the topic of this work as we assume the Markov chain is known.

Shortly after the publication of the first version of this work, Wagenmaker and Jamieson (2022) introduced a similar experiment design problem on Markov chains with the aim of developing instance-optimal RL algorithms. Unlike their work, we focus also on how the optimal policy  $\pi^*$  should be executed. Replicating the policy  $\pi^*$  for multiple episodes leads to potentially the same states being visited and a sub-optimal convergence to the optimal choice of states and actions as we show. Instead, we incrementally construct a sequence of policies that take the history of executed trajectories into account such that convergence of the empirical visitation distribution to the optimal visitation distribution of policy  $\pi^*$  is faster than resampling.

If  $f$  is interpreted as a reward function, our work can be seen as a reward learning problem similar to Lindner et al. (2021). The difference here is that our exploration is planned for an entire episode instead of assuming an access to arbitrary state-action oracle (simulator) as in their work, which makes our approach more widely applicable. Episodic planning for reward learning with different assumptions on  $f$  and methods is considered by Belogolovskiy et al. (2021).

## 2.2 Estimation

We estimate  $\mathbf{C}f$  using a regularized least squares estimator, where the estimated  $\hat{f}_t$  after  $t$  episodes is the solution to  $\hat{f}_t = \arg \min_{f \in \mathcal{H}_k} \sum_{i=1}^t \sum_{(x,a) \in \tau_i} (f^\top \Phi(x,a) - y_{a,x,i})^2 + \lambda \|f\|_{\mathcal{H}_k}^2$ . In order to estimate the functional  $\mathbf{C}f$ , we simply use  $\mathbf{C}\hat{f}$ . This estimator is motivated by the famed Gauss-Markov theorem, as it minimizes the second moment of residuals (see Mutný and Krause, 2022a, for details). Due to the representer theorem (cf., Schölkopf et al., 2001), the above problem can be solved even if the RKHS is infinite-dimensional.

We aim to understand the uncertainty of the estimate  $\mathbf{C}\hat{f}$  as a function of trajectories taken. To do so, as common in optimal experiment design, we consider the second moment of the residuals  $\mathbf{C}(\hat{f} - f)$  denoted as  $\mathbf{E}_T = \mathbb{E}_\epsilon[\mathbf{C}(\hat{f} - f)(\hat{f} - f)^\top \mathbf{C}^\top]$ , where the expectation is understood over the noise realization  $\epsilon$  (see Eq. (1)). Evaluating it, we can see its dependence on the executed trajectories  $\tau_i, i \in [T]$ :

$$\mathbf{E}_T \preceq \mathbf{C} \left( \left( \sum_{i=1}^T I(\tau_i) + \lambda \mathbf{I} \right)^{-1} \right) \mathbf{C}^\top \quad (2)$$

where we call  $I(\tau) : \mathcal{H}_k \rightarrow \mathcal{H}_k \times \mathcal{H}_k$  the *information matrix* for a trajectory  $\tau$ ,

$$I(\tau) = \sum_{(a,x) \in \tau} \frac{1}{\sigma_{a,x}^2} \Phi(x,a) \Phi(x,a)^\top. \quad (3)$$

and  $\sigma_{a,x}$  denotes the variance of random variable  $\epsilon$ . The derivation follows a straightforward application of optimality conditions, representer theorem, and use of bounded norm assumption  $\|f\|_{\mathcal{H}_k} \leq \frac{1}{\lambda}$ . A very detailed derivation is given by Mutný and Krause (2022a).

By appropriate choice of trajectories  $\tau_i$ , we can minimize the second moment of residuals in Eq. (2). As alluded to in the beginning, unless the system is completely deterministic, we cannot directly choose trajectories  $\tau_i$ . Even if we could, this space of trajectories may grow *exponentially* in  $|S|$  and  $|A|$ . Instead, we first reformulate the objective via a fractional allocation over trajectories  $\eta \in \Delta_p$ , essentially by inserting  $1/T$  into Eq. (2),

$$\mathbf{E}_T(\eta_T) \preceq \frac{1}{T} \mathbf{C} \underbrace{\left( \left( \sum_{\tau \in \mathcal{T}} I(\tau) \eta_T(\tau) + \frac{\lambda}{T} \mathbf{I} \right)^{-1} \right)}_{\text{covariance matrix } \Sigma(\eta_T)} \mathbf{C}^\top \quad (4)$$

where we identified  $\eta_T = \frac{1}{T} \sum_{i=1}^T \delta_{\tau_i}$ . Now the goal of the experiment design is to invest the  $T$  samples (trajectories) in such a way that the matrix in Eq. (4) is as small as possible.

Since the RHS is matrix-valued, a classical approach is to scalarize the objective by one of the well-known scalarization functions (see 1). A scalarization function  $s : \mathbb{S}^{p \times p} \rightarrow$

Table 1: Selected design objectives with their interpretations and name (Fedorov and Hackl, 1997).

Design	Represents	$s(\Sigma)$
D	information	$-\log \det(\Sigma^{-1})$
A	parameter error <sup>2</sup>	$\text{Tr}[\Sigma]$
E	worst projection error <sup>2</sup>	$\lambda_{\max}[\Sigma]$

$\mathbb{R}$  acts on the space of PSD matrices, and is convex. These objectives usually correspond to a specific type of uncertainty, such as squared error, entropy, or predictive error. Upon scalarization we have  $F(\eta) = \frac{1}{T} s(\Sigma(\eta))$ , which upper bounds  $s(\mathbf{E}_T)$ . The constant  $s(\Sigma(\eta))$  represents the optimality with which the budget of  $T$  policies is invested - the smaller it is the more information we extract. With experiment design, we seek to have the lowest possible constant with  $\eta^* = \arg \min_{\eta \in \Delta_p} F(\eta)$ .

### 3 EXPERIMENT DESIGN PROBLEM

Our goal is to design and execute a sequence of  $T$  policies  $\pi_i, i \in [T]$ , each one for a single episode, for a fixed number of steps  $H$  (episode length), such that we reduce the uncertainty of  $\mathbf{C}f$  efficiently. Directly optimizing probability distributions over trajectories in  $F(\eta)$  is intractable as their description size is  $(|\mathcal{X}||\mathcal{A}|)^H$  in the worst case. Instead, we focus on the building blocks of trajectories: states and actions, and their visitations. We design policies that visit the states and actions such that uncertainty is maximally reduced as measured by the function  $F$  and show that for experiment design objectives the complexity of the problem can be reduced to optimization over distributions with size  $|\mathcal{X}||\mathcal{A}|H$ .

#### 3.1 State-action polytope

To optimize the choice of policies (and hence trajectories), we work with an object that accumulates the information about state visitations: the *state-action visitation distribution*. For ease of exposition, assume that  $\mathcal{X}$  and  $\mathcal{A}$  are discrete sets. The space of all possible state-action visitation distributions with a known initial distribution  $d_0$  is the *state-action visitation polytope*,

$$\mathcal{D}_h := \left\{ d_h \mid d(x, a) \geq 0, \sum_{a,x} d_h(a, x) = 1, \sum_a d_h(x', a) = \sum_{x,a} d_{h-1}(x, a) p(x' | x, a) \right\}.$$

For reference see Puterman (2014) or Neu and Pike-Burke (2020). The *average state-action visitation polytope* over a time horizon  $H$  is the central constraint set used in this work, which we denote by  $\mathcal{D} := \{d = \frac{1}{H} \sum_{h=1}^H d_h \mid d_i \in \mathcal{D}_i \forall i \in [H]\}$ . Notice that for the fixed horizon, the state-action visitation distribution  $d_h$  for each timestep  $h$  is different. A useful approximation is to let  $H$  be very large. In that case, a reasonable approximation to this polytope is the *average-case polytope*, where  $\bar{\mathcal{D}} = \{d \mid d(x, a) \geq 0, \sum_{a,x} d(a, x) = 1, \sum_a d(x', a) =$

$\sum_{x,a} d(x, a) p(x' | x, a)\}$ , and the distribution  $d$  for each  $h$  is the same. The distributions  $d_h$  can be generated via transition operator  $P_{\pi_h}$  and a (Markov) policy  $\pi$ ,

$$P_{\pi_h}(x, x') := \sum_a p(x' | a, x) \pi_h(a | x), \quad (5)$$

to get  $d_h(x, a) = \left( \prod_{i=1}^{h-1} P_{\pi_i} d_0(x) \right) \pi_h(x | a)$ . Conversely, to match the state-action visitation by executing a policy, we can obtain a policy by marginalization:

$$\pi_h(a | x) = \frac{d_h(a, x)}{\sum_a d_h(a, x)} \text{ and } \bar{\pi}(a | x) = \frac{d(a, x)}{\sum_a d(a, x)}, \quad (6)$$

where the second case corresponds to the average case. In the latter case, the induced policy  $\pi$  is stationary, while the former is non-stationary. We will drop the subscript  $h$  from  $d$  and  $\pi$ , and instead refer to them as  $d_\pi$  and  $\pi$ , as the treatment for average and fixed horizon polytopes is essentially the same - they differ only in the form of marginalization.

**Visitations and Trajectories** Executing a policy  $\pi$  for  $H$  steps leads to a trajectory  $\tau = \{x_0, a_0, \dots, x_{H-1}, a_{H-1}\}$ . Since both the policy and/or the environment can be stochastic,  $\pi$  induces a distribution over trajectories of length  $H$ , which we denote by  $\eta_\pi \in \mathcal{P}$ . We see that  $\pi$  induces both  $\eta_\pi$  and  $d_\pi$ , and in turn,  $d_\pi$  can be matched by a Markov policy  $\pi$ . The distributions  $d_\pi$  and  $\eta_\pi$  can be related using a linear map  $\mathbf{Z}$  that we refer to as a *conversion map* (see Fig. 1c) as follows,

$$\begin{aligned} d_\pi(a, x) &= \sum_{\tau \in \mathcal{T}} \eta_\pi(\tau) \frac{1}{H} \sum_{a', x' \in \tau} \delta_{a=a', x=x'} \\ &= \frac{1}{H} \sum_{\tau \in \mathcal{T}} \#_{(a,x \in \tau)} \eta_\pi(\tau) = \mathbf{Z}_{a,x} \eta_\pi, \end{aligned} \quad (7)$$

where  $\#_{(a,x \in \tau)}$  refers to the number of times a state combination  $(x, a)$  appears in  $\tau$ , and the normalized sum with  $H$  is empirical visitation of a trajectory  $\tau$ . The relation in connection to additivity of the *information matrix* allows us to directly optimize the distribution over trajectories (and hence policies) by optimizing state-action visitations  $d_\pi$ .

#### 3.2 Loss Function

Using the *conversion map*  $\mathbf{Z}$ , we can relate the objectives  $F$  which is in terms of the distribution of trajectories  $\eta \in \mathcal{P}$  with objective  $U$  depending only on state-action visitations:

$$\min_{\eta_\pi \in \mathcal{P}} F(\eta_\pi) := \min_{\eta_\pi \in \mathcal{P}} U(\mathbf{Z}\eta_\pi) = \min_{d_\pi \in \mathcal{D}} U(d_\pi), \quad (8)$$

where  $d_\pi$  corresponds to the state-action visitation of the policy  $\pi$  (which needs to be neither Markovian nor stationary). However, there exists a Markovian policy achieving this due to the considerations. Functions  $U$  and  $F$  are essentially the same; the only difference is in terms of the decision variables involved. This simplification of the objective  $F$  to  $U$ , is possible due to the definition of

$F(\eta_\pi)$  via the information matrix  $I(\tau)$ , which is additive in terms of state-action pairs in a trajectory  $\tau$ . In particular,

$$U(d_\pi) = s \left( \sum_{a,x} \frac{d_\pi(a,x)H}{\sigma_{a,x}^2} \Phi(x,a)\Phi(x,a)^\top + \frac{\lambda \mathbf{I}}{T} \right),$$

where  $d_\pi \in \mathcal{D}$ . In general, this is not true for any  $F(\eta)$ . For a formal derivation please see Lemma 1 in Appendix A.1.

**Optimum** We want that the empirical distribution over executed trajectories  $\eta_T = \frac{1}{T} \sum_{t=1}^T \delta_{\tau_t}$  converges to the optimum of (8),  $\eta^* \in \arg \min_{\eta \in \mathcal{P}} F(\eta)$ , where  $\delta_{\tau_t}$  is a delta-function supported on the trajectory  $\tau_t$ . The optimum distribution over trajectories  $\eta^*$  corresponds to a *fixed* policy which comes from the set of optimal policies  $\Pi^*$  - equivalent in terms of their value of  $F$  (any one of them can be chosen).

By picking multiple different policies, each different in each episode, one might hope that one can perform better than a single optimal policy  $\pi^*$ , that induces  $\eta^*$ . However, in expectation, this benchmark is in fact stronger due to the convexity of  $U$ . Suppose  $\pi_i^*$  are set of policies maximizing the following first term,

$$\begin{aligned} \mathbb{E}_{\tau_i \sim \pi_i^*} [F(\eta_T)] &\stackrel{(8)}{=} \mathbb{E}_{\tau_i \sim \pi_i^*} [U(\mathbf{Z}\eta_T)] \stackrel{\text{Jen.}}{\geq} U(\mathbf{Z}\mathbb{E}_{\tau_i \sim \pi_i^*} [\eta_T]) \\ &= U\left(\frac{\mathbf{Z}}{T} \sum_{t=1}^T \mathbb{E}_{\tau_t \sim \pi_t^*} [\delta_{\tau_t}]\right) = U\left(\frac{\mathbf{Z}}{T} \sum_{t=1}^T q_t\right) \stackrel{\text{opt.}}{\geq} U(\mathbf{Z}\eta^*), \end{aligned}$$

where  $q_t = \mathbb{E}[\delta_{\tau_t}]$  is the induced probability over trajectories due to the policy  $\pi_t^*$ . The last inequality follows as  $\frac{1}{T} \sum_{t=1}^T \mathbf{Z}q_t$  is a convex combination of elements inside the state-action polytope, which is convex, and hence there exists a unique  $\mathbf{Z}\eta$  which can replicate it. Lastly, as  $\eta^*$  is the optimum over  $\mathcal{P}$ ,  $U(\mathbf{Z}\eta)$  is larger than  $U(\mathbf{Z}\eta^*)$ .

Hence, bounding the performance of the algorithm to the expected performance of the optimal set of policies  $\{\pi_i^*\}_{i=1}^T$  for  $T$  episodes can be done by bounding the performance with respect to the  $F(\eta^*)$  induced by a fixed policy  $\pi^*$ :

$$F(\eta_T) - \mathbb{E}_{\tau_i \sim \pi_i^*} [F(\eta_T)] \leq F(\eta_T) - F(\eta^*).$$

By bounding this performance, we get a bound on the other, perhaps more intuitive, objective. This is true regardless whether analyzing expected loss  $\mathbb{E}_{\tau_i \sim \pi_i} [F(\eta_T)]$  or high probability error for  $F(\eta_T)$  as in Sec 6.

## 4 CONVEX RL: NON-ADAPTIVE DESIGN

Given a convex objective  $U(d)$  (related to  $F(\eta) = U(\mathbf{Z}\eta)$ ) over polytope  $d \in \mathcal{D}$  (either the average or fixed horizon average polytope), we can solve for a policy  $\pi^*$  which achieves the optimum of  $d^*$  easily due to seminal works of Hazan et al. (2019) and Zahavy et al. (2021). We refer to it as *convex Reinforcement Learning (RL)*. It proceeds by solving a sequence of classical RL problems with a linear reward function that corresponds to the gradient of  $U(d)$ . We use it

as a subroutine in our adaptive algorithm that we explain in Section 5. Using this method, we can find a fixed policy  $\pi^*$  that matches the optimum value  $U(d^*) = F(\eta^*)$ . It does so by constructing a convex combination of base policies.

**Mixture Policy** We refer to a convex combination of policies as *mixture policy*. It is a tuple consisting of a set  $n$  base policies, and set of  $n$  positive weights  $\alpha_i$  that sum to one,  $\pi_{\text{mix},n} = \{(\alpha_i, \pi_i)\}_{i=1}^n$ . Such a mixture can be executed by first sampling an index  $j \in [n]$  with probability equal to  $\alpha_j$ , and then evaluating policy  $\pi_j$  for  $H$  rounds. A property of mixture policies is that the state-action probabilities follow the convex combination of the policies such as  $d_{\pi_{\text{mix},n}} = \sum_{i=1}^n \alpha_i d_{\pi_i}$ . Note that like any other policy, a mixture policy can be summarized by a single policy via marginalization as in Eq. (6).

### 4.1 Convex RL as Frank-Wolfe

Convex RL can be solved via the Frank-Wolfe algorithm – it incrementally constructs a mixture policy whose state-action distribution  $d_{\pi_{\text{mix},n}}$  converges to  $d^*$  as the number of mixture components  $n$  increases. There are two distinct steps. The first is called *density estimation*, corresponding to the evaluation of the gradient in Frank-Wolfe, and the second is *policy search*, corresponding to the linear minimization oracle in Frank-Wolfe.

**Density Estimation** Given a mixture policy, the density estimation oracle needs to estimate  $d_{\pi_{\text{mix}}}$ . For discrete Markov chains (i.e., tabular MDPs), this amounts to a straightforward application of the operator  $P_\pi(x', x)$  to  $d_0$  as in (5) for every component of the mixture policy. In particular for the fixed horizon setting  $d_{\pi_i}(x, a) = \frac{1}{H} \sum_{h=1}^H \prod_{j=1}^h P_{(\pi_i)_j} d_0(x) \pi_i(x|a)$  where subscript  $j$  denotes the iteration within the episode as  $\pi$  is non-stationary for a fixed horizon. The overall mixture state-action visitation is the convex combination of all mixture components. Beyond discrete Markov chains, any state-visitation density can be estimated via sampling. As we know the transition operator  $P$ , this means simulating the policy  $\pi$ , we do not require interaction with the real environment.

**Policy Search** Having estimated  $d_{\pi_{\text{mix},n}}$  with  $n$  elements, we can now add an element  $\pi_{n+1}$  such that the objective  $U$  decreases. We linearize the objective  $\nabla U(d)$  and solve the linear minimization oracle:

$$d_{\pi_{n+1}} = \arg \min_{d_\pi \in \mathcal{D}} \sum_{x,a} \nabla U(d_{\pi_{\text{mix}}})(x, a) d(x, a)$$

This is a classical reinforcement learning problem, where  $\nabla U(d_{\pi_{\text{mix}}})$  plays the role of the reward function (Puterman, 2014). Hence, it can be solved by any RL solver such as value/policy iteration or linear programming with a Markov policy. The newly found  $d_{\pi_{n+1}}$  defines  $\pi_{n+1}$  or vice versa depending on the RL solver used. The weight of new policy  $\pi_{n+1}$ ,  $\alpha_{n+1}$ , is found via a line search as in Alg. 1 or any other convergent step-size scheme for the Frank-Wolfe algorithm (Jaggi, 2013). The new mixture policy is then

$\pi_{\text{mix},n+1} = \{((1 - \alpha_{n+1})\alpha_i, \pi_i)\}_{i=1}^n \cup \{\alpha_{n+1}, \pi_{n+1}\}$ . This algorithm is summarized in Alg. 1 as **Convex RL**. Hazan et al. (2019) prove that in order to converge to  $\epsilon$  optimality in terms of  $U$ , under the regularity conditions as in Assumption 1 (see Sec. 6), one needs  $n \geq \mathcal{O}(\frac{L}{\epsilon} \log(1/\epsilon))$  steps with step size  $\alpha_n = \frac{\epsilon}{L}$  for all  $n$ .<sup>2</sup>

## 5 MARKOV-DESIGN: ADAPTIVE DESIGN

In the previous section, we discussed how to find a *single policy* such that its state-action visitation probability minimizes a certain convex functional  $U$ . However, ultimately our objective depends on the empirical distribution  $\eta_T = \frac{1}{T} \sum_{t=1}^T \delta_{\tau_t}$  of the executed trajectories. We now consider ways to generate trajectories: first, the straightforward NON-ADAPTIVE methods that execute a single policy multiple times, and then we describe the ADAPTIVE methods, which lie at the core of the contribution.

### 5.1 Resampling from mixture $\pi^*$

**VARIANT: NON-ADAPTIVE** As our final density is in the form of a mixture policy, we can either sample a component for each episode or summarize the policy by marginalization and execute it multiple times resulting in empirical  $\eta_T$ . The value  $F(\eta_T) \rightarrow F(\eta^*)$  with probability  $1 - \delta$ , depending on the gradient norm  $B = \|\nabla F(\eta_T)\|_\infty$ , as  $\mathcal{O}(B \log(1/\delta)/\sqrt{T})$ . In general, this strategy suffers from the *coupon collector problem*, where the same trajectories are resampled with non-zero probability. Namely, this manifests itself in the number of  $T$  needed such that  $B$  is well-behaved. One can construct instances of the problem where the coupon collector problem is clearly manifested where  $T \approx \mathcal{O}(p \log(p))$  before  $B$  is not  $T$  dependent. While for the adaptive variant, that we are about to present in the next subsection, it is  $T \approx \mathcal{O}(p)$ ; for more details and formal statements, see Appendix B.2 and B.3.

**VARIANT: TRACKING** The NON-ADAPTIVE variant is wasteful in that there exists a non-zero probability that the same base policy is executed multiple times, leading to similar trajectories (and hence redundant experiments). This can be avoided via *tracking* – closely following the empirical distribution of executed policies to the mixture found by convex RL. Namely, we choose the base policy  $\pi_j$  such that  $j = \arg \max_i (\alpha_i - \hat{\alpha}_i)$ , where  $\hat{\alpha}_i$  corresponds to the empirical distribution of executed policies. This, however, can be wasteful, as it depends on how the mixture policy is decomposed. If the mixture contains a lot of dissimilar policies, then this method can be very competitive. On the other hand, if the components are all very similar, then this is as wasteful as the NON-ADAPTIVE variant.

### 5.2 Adaptive Optimization on $\mathcal{P}$

A more elegant way to avoid resampling is to inform the choice of the next policy with the information about the

<sup>2</sup>A large smoothness parameter  $L$  of  $U$  is not a pressing issue as  $n$  does not correspond to actual samples of the environment only the computational effort

executed trajectories from previous steps. To do this, we incrementally estimate the empirical distribution of the visited states from past trajectory distributions as  $\mathbf{Z}\eta_t$ , where  $\eta_t = \frac{1}{t} \sum_{i=1}^t \delta_{\tau_i}$ . We then seek an addition to this empirical measure  $d = \mathbf{Z}q_t$  as  $\frac{1}{1+t}d + \frac{t}{1+t}\mathbf{Z}\eta_t$  which minimizes the objective  $G_t(d) = U(\frac{1}{1+t}d + \frac{t}{1+t}\mathbf{Z}\eta_t)$ . Notice that the weighting  $\frac{1}{1+t}$  is chosen such that the new empirical distribution over trajectories  $\eta_{t+1} = \frac{1}{1+t}\delta_t + \frac{t}{1+t}\eta_t$  where  $\delta_t \sim q_t$  remains still the average allocation over the executed trajectories. It is in a sense a greedy one-step change in the allocation that brings us closer to the optimal allocation.

**VARIANTS: EXACT and ONE-STEP** Depending on how we identify the distribution over trajectories  $q_t$  (and the associated  $d$ ), we distinguish two variants. The EXACT variant finds the exact element such that  $q_t = \arg \min_{q \in \mathcal{P}} U(\frac{1}{1+t}\mathbf{Z}q_t + \frac{t}{1+t}\mathbf{Z}\eta_t)$ . The ONE-STEP variant simply runs the Convex MDP framework for one step and plays the first component in the mixture policy  $\pi_{\text{mix}}$  as summarized in Alg. 1. The ONE-STEP can be seen as a computationally frugal version of EXACT, although there seem to be empirical benefits for the former with deterministic environments as we will see.

### 5.3 Uncertain Objectives

In some cases, the objective  $F$  depends on an unknown quantity  $\gamma \in \Gamma$  such as unknown variance  $\sigma_{x,a}^2$ , we denote such objective  $F_\gamma$ . The set  $\Gamma$  can be also time-dependent. To deal with this complication, there are two approaches one can take:

- *Robust design*: take a supremum over the set  $\Gamma$ , and have a design that takes into account any possible values of  $\gamma$ . If  $F$  is convex then so is the supremum over the compact set.
- *Sequential design*: amend the objective  $F$  with the new value of the supremum if by executing an episode one can reduce the size of the set  $\Gamma_t$ . In the context of the example with unknown variance, we can estimate the variance from repeated samples, and the confidence sets, coincides with  $\Gamma_t$ . The confidence set shrinks over time as more information is gathered eventually containing only values very close to the true  $\gamma$ .

We give examples for both of these design approaches in the experimental Section 7. Further details along with theoretical convergence guarantees are given in Appendix C.

## 6 CONVERGENCE THEORY

The ONE-STEP variant is closely related to the Frank-Wolfe algorithm on the space of trajectory distributions and the theory we develop for its convergence is largely based on it. The convergence cannot be linear unless step sizes are adjusted (Lacoste-Julien and Jaggi, 2015), however, our  $\frac{1}{1+t}$  step sizes are determined by the one-step update specific to this setting, and cannot be changed.

**Algorithm 1** MARKOV-DESIGN

**Require:** known Markov chain,  $p(x'|x, a)$ , Objective  $F(U)$ , Number of episodes  $T$

- 1: **while**  $t \leq T$  **do**
- 2:     **Convex RL:** solving  $\min_{d \in \mathcal{D}} G_t(d) := U(\mathbf{Z}\eta_t \frac{t}{t+1} + \frac{1}{t+1}d)$
- 3:      $\pi_{\text{mix},1} = \pi_t; i = 1, d_{\pi_{\text{mix},1}} = \mathbf{Z}\eta_t$
- 4:     **repeat**    $i = i + 1$
- 5:          $\varpi_i = \arg \min_{\pi \in \Pi} \sum_{x,a} d_{\pi}(x, a) \nabla_{x,a} G_t(d_{\pi_{\text{mix},i}}(x, a))$  ▷ RL problem
- 6:          $d_{\varpi_i} = \text{Density Estimation}(\varpi_i)$  ▷ keep track of visitations
- 7:          $\alpha_i = \arg \min_{\alpha \in \mathbb{R}} G_t(\alpha d_{\pi_{\text{mix},i}} + (1 - \alpha)d_{\varpi_i})$  ▷ line search
- 8:          $\pi_{\text{mix},i+1} = (1 - \alpha_i)\pi_{\text{mix},i} \cup \{(\alpha_i, \varpi_i)\}$  ▷ update mixture policy
- 9:         Update  $d_{\pi_{\text{mix},i+1}} = \alpha_i d_{\pi_{\text{mix},i}} + (1 - \alpha_i)d_{\varpi_i}$  ▷ update mixture policy visitations
- 10:     **until** convergence
- 11:     **Choose**  $\pi_t = \begin{cases} \text{Marginalize } \pi_{\text{mix}} & \text{if VARIANT = EXACT} \\ \varpi_1 & \text{if VARIANT = ONE-STEP} \end{cases}$
- 12:     **Interaction**
- 13:     Sample trajectory from  $\tau_t \sim \pi_t$  (also as  $\delta_{\tau_t} \sim q_t$ )
- 14:      $\mathbf{Z}\eta_{t+1} = \mathbf{Z} \frac{t}{t+1} \eta_t + \mathbf{Z} \frac{1}{t+1} \delta_{\tau_t}$  ▷ keep track of visited states
- 15: **end while**

We utilize the same convergence proof for the EXACT and ONE-STEP variants, although the two have different convergence behavior on the real problems. The regularity conditions under which we show convergence are summarized below.

**Assumption 1** (Regularity). *Let  $F : \Delta_p \rightarrow \mathbb{R}$  (and likewise  $U$ ) be convex, differentiable, locally Lipschitz continuous in  $\|\cdot\|_\infty$ , and locally smooth as,*

$$F(\eta + \alpha h) \leq F(\eta) + \nabla F(\eta)^\top h + \frac{L_{\eta,\alpha}}{2} \|h\|_2^2. \quad (9)$$

for  $\alpha \in (0, 1)$  and  $\eta, h \in \Delta_p$ ,  $L := \max_{\eta,\alpha} L_{\eta,\alpha}$ .

Note that the above differs from the classical smoothness assumption, which we refer to as global smoothness. On its own global smoothness is not sufficient to prove the desired rate of convergence we observe in practice. The problem is that the experiment design objectives can have global smoothness  $L$  proportional to  $\frac{T}{\lambda}$ , which does not suffice to prove convergence – an issue not addressed in prior works (Wagenmaker and Jamieson, 2022). However, the usual behavior is that after a few initial steps (eg. ca.  $\mathcal{O}(p)$ ), the local smoothness constants drop to a small number. On top of that some objectives might not be smooth at all like E-design. To remedy both of these, one can apply the classical smoothing technique due to Nesterov (2005), where the smoothed function  $F_\mu$  is  $\mu$  close to  $F$ , but with smoothness  $L_\mu = L/(1 + \mu L) \leq \frac{1}{\mu}$ . Applying our algorithm on objective  $F_\mu$  gives order  $T$  optimal convergence rate with high probability.

**Theorem 1** (Convergence EXACT and ONE-STEP). *Under Assumption 1, with the smoothed objective using  $\mu =$*

$\sqrt{\log T/T}$ , the EXACT and ONE-STEP variants satisfy,

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

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

Proofs are postponed to Appendix B.4, and Nesterov (2005) smoothing technique is reviewed in Appendix A.2 for completeness. With a reasonable upper bound on the gradient of  $F$ , the convergence is as  $\mathcal{O}(\frac{1}{\sqrt{T}})$  even for non-smooth objectives.

However, smoothing is not necessary as our experimental results point out. Particularly its use hampers the potentially fast convergence in expectation, which is the driving term in practice, especially for deterministic Markov chains. Therefore, we state the next result in terms of the local Lipschitz constant  $L_{\eta_t,1/t}$ , which we conjecture, depend only logarithmically on  $T$ , leading to convergence  $\mathcal{O}(\frac{\log T}{T})$ .

**Theorem 2** (Convergence EXACT and ONE-STEP). *Under Assumption 1, the EXACT and ONE-STEP variants satisfy*

$$\mathbb{E}[F(\eta_t)] - F(\eta^*) \leq \mathcal{O} \left( \frac{1}{t} \sum_{k=1}^T \frac{L_{\eta_k,1/k}}{1+k} \right)$$

for  $t \leq T$  with expectation over transition model and policy.

Showing non-trivial bounds on the smoothness and gradient remains a challenging problem not addressed even in the context of classical experiment design (Zhao and Freund, 2022). The only existing approaches that address this follow a particular initialization scheme which is infeasible for the kernelized regime and specialized for D-design (Todd,

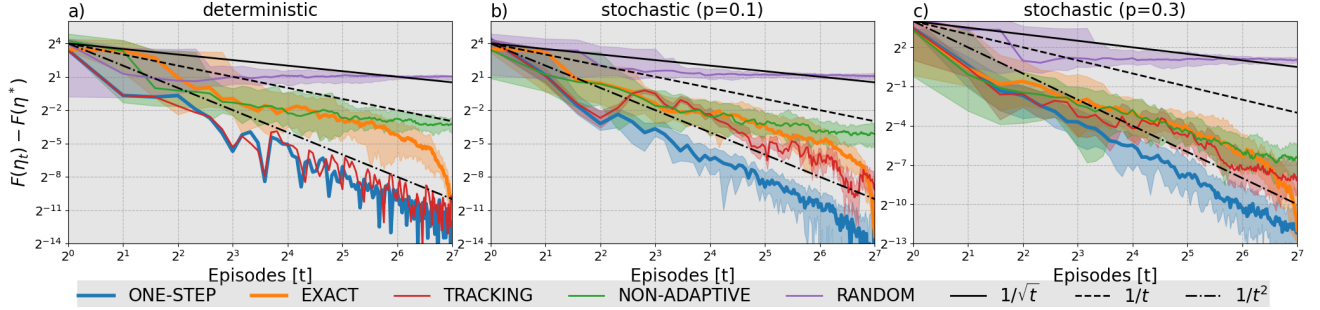


Figure 2: Gridworlds experiment: From a) to c) we increase the stochasticity of the environment. We plot the median with 10% and 90% quantiles over 20 reruns of the method. Notice that the adaptive methods perform much better than the non-adaptive ones, and very quickly optimize the objective. For deterministic systems, the ONE-STEP method is deterministic as it coincides with the greedy method over a set of deterministic policies. Surprisingly, the ONE-STEP method dominates the EXACT method for short horizons despite EXACT demonstrating linear convergence in later stages. The non-adaptive method exhibits slow convergence, while random selection does not converge. Notice that we plot suboptimality from the optimal constant; the average uncertainty decreases in all these cases as  $1/t$  on top of the decrease in this plot.

2016). Nevertheless, we conjecture that for example for D-design, the gradient and the smoothness are quickly of order  $\text{polylog}(T)$ . Even with a proper initialization, one of the main challenges hampering the analysis of the algorithm is that the algorithm has fixed step-size  $1/(1+t)$  and is not monotonically decreasing like other analyses of Frank-Wolfe (Carderera et al., 2021).

It is worth pausing, and seeing that the consequence of the above theorems is that the suboptimality in the *constant* decreases as  $F(\eta_T) \leq F(\eta^*) + T^{-1/2}$ , which in turn means that the second moment of residuals in scalarization (see Eq. 4),  $s(\mathbf{E}_L) \leq \frac{F(\eta^*)}{T} + \mathcal{O}(T^{-3/2})$ , where the leading term in terms of  $T$  depends on the optimal constant as wanted. Analog statements hold in expectation albeit with  $\mathcal{O}(T^{-2})$ .

## 7 APPLICATIONS & EXPERIMENTS

We present three applications of the proposed framework. We plot the suboptimality gap  $F(\eta_t) - F(\eta_*)$ . Further details of the experiments can be found in Appendix E. Overall, empirical convergence tends to have two phases. First slow, but then quickly, the rate of convergence is of the order  $\mathcal{O}(\frac{1}{T})$  to  $\mathcal{O}(\frac{1}{T^2})$ . In limited cases, the EXACT variant exhibits linear convergence (see Figs. 2 and 3). Note that for statistical estimation the suboptimality at  $t = T$  is decisive, as this is what we compare to. Notice that all adaptive algorithms achieve a very similar value at  $t = T$ , but this is not true for non-adaptive variants. Nevertheless, to investigate our theoretical convergence results, we plot the whole trajectory  $t \in [0, T]$ .

**Synthetic Gridworlds** Consider a grid of a fixed height and width as in Fig. 1, where possible actions are to move up, down, left, and right by one cell. We consider three levels of increasing stochasticity. We assume that with probability  $1 - p$ , the action we take is executed as expected, and with probability  $p$  an arbitrary valid action (up, down, left, right) is played instead. The unknown target

$f$  corresponds to a linear function and the features  $\Phi(x, a)$  correspond to a unit vector in a different direction for each state type (cf., pictogram). Thus, the states of the same type are completely correlated while different ones are not at all correlated. Fig. 2 shows that the convergence of our algorithm is of order  $\mathcal{O}(1/T^2)$  faster than our theory predicts, and  $\mathcal{O}(1/\sqrt{T})$  for the NON-ADAPTIVE variant as we expect. With increasing stochasticity, the performance of the TRACKING variant reduces as it cannot adapt to executed trajectories from prior steps, while our adaptive methods can.

**Species monitoring: path planning** Suppose we want to estimate the rate of occurrence of a particular species. To model this application, we use occurrence data of *Beilschmiedia*, a tree genus native to Africa, from Baddeley et al. (2015). We model the occurrence rate  $f$  as a positive valued RKHS function that determines the rate of the spatio-temporal Poisson point process. The states in this case are sectors  $X$  which are sensed. (See the map Fig. 4a in Appendix E). We adopt the approach from Mutný and Krause (2021), where  $f$  is estimated with a penalized least squares estimator from count observations. The number of counts in the region  $X$  is distributed according to  $y|X \sim \text{Poisson}(\int_{x \in X} f(x) dx)$ . A peculiar property of the Poisson distribution is that its variance and mean are the same. Hence, the values of  $\sigma_{a,X}^2 = \int_{x \in X} f(x) dx$ , are unknown due to the unknown  $f$ , and this makes  $F$  unknown.

To model  $f$ , we use a squared exponential kernel that takes the slope  $s_{x,y}$  and height  $h_{x,y}$  of a point  $(x, y)$  as inputs, as these are predictive of the habitat of *Beilschmiedia*. We assume that a drone can cover a certain trajectory of length  $H$  before it has to return to the starting position. We use the D-design objective to maximize the information about  $f$  everywhere in the domain. As  $\sigma_{X,a}$  is unknown, we either run our algorithm with a known absolute upper bound on  $\sigma_{X,a}$  (due to norm bound on  $\|f\|_{\mathcal{H}_k}$ ) as in Fig. 3a) or, alternatively, use confidence sets from Mutný and Krause



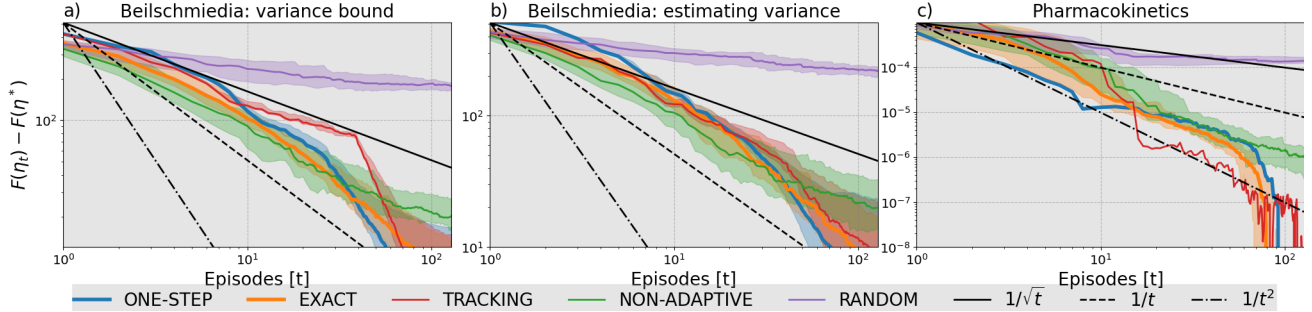


Figure 3: Beilschmiedia, and pharmacokinetics experiment: We report median of 20 reruns with 10% and 90% quantiles. In a) and b) we report the objective value as a function of episodes  $t$  for  $\sigma_{x,a}$  with a global upper bound, and one where we update the confidence set over  $\sigma_{x,a}$  after each episode, respectively. Notice that we can converge to the objective faster in b) than a) due to better exploration policy (as it is more informed about values of variance). Note that ONE-STEP and EXACT are dominating the convergence. In c) we report the same objective for the pharmacokinetic example. TRACKING works very well for this problem, most likely due to a favorable mixture decomposition. Notice that we plot suboptimality from the optimal constant; the average uncertainty decreases in all these cases as  $1/t$  on top of the decrease in this plot.

(2021) on  $f$  for adaptively collected data points. These allow us to construct confidence sets on  $\sigma_{X,a}$  after each episode, and we take the upper bound of those. Results are shown in Fig.3b). In both cases, the adaptive variants work well and their convergence rate is consistent with the theory.

**Group pharmacokinetics** The goal of pharmacokinetics is to identify the rate of drug transport between the digestion system and other organ systems such as the circulatory system (blood). Such studies are designed for any new drug candidates to understand their absorption rates. The experiments are performed by drawing blood at specific time intervals and inferring the medication concentrations over time. From this, the experimenters find the corresponding parameters  $\gamma$  for the differential equation, which generates these concentration trajectories. We assume that the more accurately we can estimate the drug concentration trajectory over time for each patient, the more accurately we can estimate the parameters of the differential equation that generated these drug concentrations – and hence focus our design on estimating the trajectories. More realistically, we also assume that the concentration in the blood has two components  $f_i(t) = c_b(t) + g_i(t) + \epsilon$  where  $\epsilon \sim \mathcal{N}(0, \sigma^2)$  and  $t$  is time. Hereby,  $c_b(t)$  is the blood concentration of interest following a differential equation with parameters  $\gamma$ , and  $g_i(t)$  is a patient-specific random variation that contaminates our measurements and we are not interested in inferring it per se.

The differential equation is linear and hence forms a linear constraint on the estimation as  $L_\gamma(c_b) = 0$ , where  $L_\gamma$  is the linear differential operator. Note that this means that  $c_b$  is in the null space of operator  $L_\gamma$ , which we denote  $\mathbf{C}_\gamma$  and the uncertainty of  $c_b$  for a fixed  $\gamma$  is then due to initial conditions only. Paradoxically, while we set the initial concentration in our experiment by specifying dosage, we will model the initial conditions as unknown with very little uncertainty, and then design an objective that reduces this

uncertainty for a fixed  $\gamma$ . This type of reasoning is only a means to an end to achieve an efficient design. Since  $\gamma$  is unknown, we will use a *robust designs* of the experimental design objective, considering the supremum (resp. infimum if negative) over reasonable values of  $\gamma \in \Gamma$  as  $F(\eta) = \sup_{\gamma \in \Gamma} \text{Tr} \left[ (\mathbf{C}_\gamma (\sum_{\tau \in \mathcal{T}} \eta(\tau) I(\tau) + (\lambda/T) \mathbf{I})^{-1} \mathbf{C}_\gamma^\top) \right]$  to define  $F$ . The policy, in this case, is a medical plan when to draw blood with a constraint that blood can be drawn only 5 times per patient (episode) with reasonable separations (three time steps). Each patient corresponds to an episode. We report the results in Fig. 3c). The patient-specific variation enters as heteroscedastic noise inside the definition of  $I(\tau)$ .

## 8 CONCLUSION

We introduced a novel algorithm MARKOV-DESIGN for experiment design in Markov chains, capable of finding a set of exploratory policies that converge to an optimal allocation over trajectories to learn an unknown function of the states and actions in a known Markov chain. The algorithm solves a sequence of convex RL problems, which are informed by the previous trajectories of the agent. We proved the convergence rate of the method and compared it to other non-adaptive approaches. We demonstrated its superior empirical performance in real-world problems and hope that this work will open a new avenue to study controlled Markov chains from an experimental design perspective.

## Acknowledgement

This project has received funding Swiss National Science Foundation through NFP75 and this publication was created as part of NCCR Catalysis (grant number 180544), a National Centre of Competence in Research funded by the Swiss National Science Foundation.

## References

- Baddeley, A., E., R., and R., T. (2015). *Spatial Point Patterns: Methodology and Applications with R*. Chapman and Hall/CRC Press.
- Beck, A. and Teboulle, M. (2012). Smoothing and first order methods: A unified framework. *SIAM Journal on Optimization*, 22(2):557–580.
- Belogolovsky, S., Korsunsky, P., Mannor, S., Tessler, C., and Zahavy, T. (2021). Inverse reinforcement learning in contextual mdps. *Machine Learning*.
- Berthet, Q. and Perchet, V. (2017). Bandit optimization with upper-confidence frank-wolfe. *arXiv preprint arXiv:1702.06917*.
- Borsos, Z., Mutný, M., and Krause, A. (2020). Coresets via bilevel optimization for continual learning and streaming. *Neural and Information Processing Systems (NeurIPS) 2020*.
- Borwein, J. and Zhu, Q. (2005). *Techniques of Variational Analysis*. Springer-Verlag New York.
- Carderera, A., Besançon, M., and Pokutta, S. (2021). Simple steps are all you need: Frank-wolfe and generalized self-concordant functions. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*, volume 34, pages 5390–5401. Curran Associates, Inc.
- Chaloner, K. and Verdinelli, I. (1995). Bayesian experimental design: A review. *Statist. Sci.*, 10(3):273–304.
- Duff, M. O. (2002). *Optimal learning: Computational procedures for Bayes-adaptive Markov decision processes*. PhD thesis, University of Massachusetts Amherst.
- Fedorov, V. V. and Hackl, P. (1997). *Model-Oriented Design of Experiments | Valerii V. Fedorov | Springer*. Springer-Verlag New York.
- Fisher, R. A. (1926). *The Arrangement of Field Experiments*. New York, NY.
- Foster, A., Ivanova, D. R., Malik, I., and Rainforth, T. (2021). Deep adaptive design: Amortizing sequential bayesian experimental design. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 3384–3395. PMLR.
- Gabrielsson, J. and Weiner, D. (1995). Pharmacokinetic and pharmacodynamic data analysis. *Trends in Pharmacological Sciences*, 16(4):143.
- Golovin, D. and Krause, A. (2011). Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *J. Artif. Int. Res.*, 42(1):427–486.
- Hazan, E., Kakade, S., Singh, K., and Van Soest, A. (2019). Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pages 2681–2691. PMLR.
- Jaggi, M. (2013). Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In Dasgupta, S. and McAllester, D., editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 427–435, Atlanta, Georgia, USA. PMLR.
- Jin, C., Yang, Z., Wang, Z., and Jordan, M. I. (2020). Provably efficient reinforcement learning with linear function approximation. In Abernethy, J. and Agarwal, S., editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 2137–2143. PMLR.
- Jourdan, M., Mutný, M., Kirschner, J., and Krause, A. (2021). Efficient pure exploration for combinatorial bandits with semi-bandit feedback. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory (ALT)*.
- Kiefer, J. (1959). Optimum experimental designs. *Journal of the Royal Statistical Society. Series B (Methodological)*, 21(2):272–319.
- Kirschner, J. and Krause, A. (2018). Information directed sampling and bandits with heteroscedastic noise. *COLT*.
- Kirschner, J., Mutný, M., Hiller, N., Ischebeck, R., and Krause, A. (2019). Adaptive and safe bayesian optimization in high dimensions via one-dimensional subspaces. *ICML 2019*.
- Krause, A. and Golovin, D. (2014). Submodular function maximization.
- Krause, A. and Guestrin, C. (2005). Optimal nonmyopic value of information in graphical models: efficient algorithms and theoretical limits. *AAAI 2005*.
- Lacoste-Julien, S. and Jaggi, M. (2015). On the global linear convergence of frank-wolfe optimization variants. *Advances in neural information processing systems*, 28.
- Lindner, D., Turchetta, M., Tschitschek, S., Ciosek, K., and Krause, A. (2021). Information directed reward learning for reinforcement learning. In *Proc. Neural Information Processing Systems (NeurIPS)*.
- Lugosi, G. (2009). Concentration of measure inequalities. *Lecture Notes*.
- Mutný, M. and Krause, A. (2018). Efficient high dimensional bayesian optimization with additivity and quadrature fourier features. In *Neural and Information Processing Systems (NeurIPS) 2018*.

- Mutný, M. and Krause, A. (2021). No-regret algorithms for capturing events in poisson point processes. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning (ICML 2021)*, volume 139 of *Proceedings of Machine Learning Research*, pages 7894–7904. PMLR.
- Mutný, M. and Krause, A. (2022a). Experimental design for linear functionals in reproducing kernel hilbert spaces: Methods, confidence sets and applications. *submitted*.
- Mutný, M. and Krause, A. (2022b). Sensing cox processes via posterior sampling and positive bases. *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS 2022)*.
- Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294.
- Nesterov, Y. (2005). Smooth minimization of non-smooth functions. *Math. Program.*, 103:127–152.
- Neu, G. and Pike-Burke, C. (2020). A unifying view of optimism in episodic reinforcement learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1392–1403. Curran Associates, Inc.
- Pukelsheim, F. (2006). *Optimal Design of Experiments*. SIAM.
- Puterman, M. L. (2014). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley Sons.
- Romero, P. A., Krause, A., and Arnold, F. H. (2013). Navigating the protein fitness landscape with gaussian processes. *Proceedings of the National Academy of Sciences*, 110(3):E193–E201.
- Sato, M., Abe, K., and Takeda, H. (1981). Learning control of finite markov chains. *IFAC Proceedings Volumes*, 14(2):1047–1052. 8th IFAC World Congress on Control Science and Technology for the Progress of Society, Kyoto, Japan, 24-28 August 1981.
- Schölkopf, B., Herbrich, R., and Smola, A. (2001). A generalized representer theorem. In *Computational learning theory*, pages 416–426. Springer.
- Settles, B. (2009). *Active learning literature survey*. University of Wisconsin-Madison Department of Computer Sciences.
- Szepesvari, C. and Lattimore, T. (2020). *Bandit Algorithms*. Cambridge University Press.
- Talebi, M. S., Zou, Z., Combes, R., Proutiere, A., and Johansson, M. (2013). Stochastic online shortest path routing: The value of feedback. *arXiv preprint arXiv:1309.7367*.
- Tarbouriech, J. and Lazaric, A. (2019). Active exploration in markov decision processes. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 974–982. PMLR.
- Todd, M. J. (2016). *Minimum-Volume Ellipsoids: Theory and Algorithms*. SIAM, mos-siam series on optimization edition.
- Wagenmaker, A. and Jamieson, K. (2022). Instance-dependent policy learning for linear mdps via online experiment design. In *NeurIPS*.
- Wainwright, M. J. (2019). *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press.
- Zahavy, T., O’Donoghue, B., Desjardins, G., and Singh, S. (2021). Reward is enough for convex mdps. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*.
- Zhang, F. (2011). *Matrix Theory: Basic Results and Techniques*. Springer Science & Business Media.
- Zhao, R. and Freund, R. (2022). analysis of the frank–wolfe method for convex composite optimization involving a logarithmically-homogeneous barrier. *Math. Program.*
- Zou, Z., Proutiere, A., and Johansson, M. (2014). Online shortest path routing: The value of information. In *American Control Conference (ACC), 2014*, pages 2142–2147. IEEE.

## Supplementary Material: Active Exploration via Experiment Design in Markov Chains

### A Additional Results

#### A.1 Relating state-visitations and trajectories

We constructively prove that  $F(\eta)$  and  $U(d)$  can be related easily for scalarized information matrix objectives.

**Lemma 1** (Conversion Mapping). *Let  $F(\eta) = s(\mathbf{S}(\eta))$ , where  $\mathbf{S}(\eta) \in \mathbb{S}_+$ , positive semi-definite cone of operators  $\mathcal{H}_k \rightarrow \mathcal{H}_k$ ,  $s$  is scalarization,  $s : \mathbb{S}_+ \rightarrow \mathbb{R}$ , and the matrix is s.t.*

$$\mathbf{S}(\eta) = \sum_{\tau \in \mathcal{T}} \eta(\tau) I(\tau) + \lambda \mathbf{I},$$

then there exists  $U(d) = U(\mathbf{Z}\eta) = F(\eta)$  equal to

$$U(d) = s \left( \sum_{a,x \in \mathcal{A} \times \mathcal{X}} d(a,x) \frac{H}{\sigma_{a,x}^2} \Phi(x,a) \Phi(x,a)^\top + \lambda \mathbf{I} \right).$$

*Proof.* The proof goes by construction, where we first construct  $U$  and then verify it satisfies the desired property. The whole proof relies only on the additive property of information operator  $I(\tau)$ .

$$U(d_\pi) = s \left( \sum_{a,x \in \mathcal{A} \times \mathcal{X}} d_\pi(a,x) \frac{H}{\sigma_{a,x}^2} \Phi(x,a) \Phi(x,a)^\top + \lambda \mathbf{I} \right) \quad (10)$$

$$\stackrel{(7)}{=} s \left( \sum_{a,x \in \mathcal{A} \times \mathcal{X}} \sum_{\tau \in \mathcal{T}} \mathbf{Z}(a,s,\tau) \eta_\pi(\tau) \frac{H}{\sigma_{a,x}^2} \Phi(x,a) \Phi(x,a)^\top + \lambda \mathbf{I} \right) \quad (11)$$

$$= s \left( \sum_{a,x \in \mathcal{A} \times \mathcal{X}} \sum_{\tau \in \mathcal{T}} \frac{1}{H} \#(a,x \in \tau) \eta_\pi(\tau) \frac{H}{\sigma_{a,x}^2} \Phi(x,a) \Phi(x,a)^\top + \lambda \mathbf{I} \right) \quad (12)$$

$$= s \left( \sum_{\tau \in \mathcal{T}} \eta_\pi(\tau) \sum_{a,x \in \mathcal{A} \times \mathcal{X}} \#(a,x \in \tau) \frac{H}{\sigma_{a,x}^2} \Phi(x,a) \Phi(x,a)^\top + \lambda \mathbf{I} \right) \quad (13)$$

$$\stackrel{(4)}{=} s \left( \sum_{\tau \in \mathcal{T}} \eta_\pi(\tau) I(\tau) + \lambda \mathbf{I} \right) = F(\eta_\pi) \quad (14)$$

□

#### A.2 Smoothing Technique for Convex Optimization

In this section, we will briefly review, by now, the classical technique of convex optimization for non-smooth functions by [Nesterov \(2005\)](#). The development outlined here is inspired by [Beck and Teboulle \(2012\)](#). Suppose the function  $F(\eta)$  is either smooth with a large constant  $L$  or non-smooth. In what follows, the non-smooth case is recovered by letting  $L \rightarrow \infty$ . A central object component in defining the smoothing is the convex conjugate of  $F$ ,  $F^*$ ,

$$F^*(\zeta) := \max_{\eta \in \mathcal{P}} \eta^\top \zeta - F(\eta)$$

In order to define a smoothed function  $F_\eta$ , we perform the reverse operation with added regularization

$$F_\mu(\eta) := \max_{\zeta} \zeta^\top \eta - F^*(\zeta) - \frac{\mu}{2} \|\zeta\|^2, \quad (15)$$

where the domain of  $\zeta$  is everywhere where  $F^*$  is finite. Equivalently it can be represented via dual reformulation as

$$F_\mu(\eta) = \inf_{\zeta \in \mathcal{P}} F(\zeta) + \frac{1}{2\mu} \|\zeta - \eta\|_2, \quad (16)$$

which is sometimes referred to as Moreau proximal smoothing (Beck and Teboulle, 2012). Due to the above definition it is clear that  $F_\mu(\eta) \leq F(\eta) \leq F_\mu(\eta) + \mu \max_{x,y \in \mathcal{P}} \|x - y\|_2 \leq F_\mu(\eta) + 2\mu$ . Hence by choosing sufficiently small  $\mu$ , we can ensure that minimizing  $F_\mu$  will closely minimize  $F$ . In addition,  $F_\mu$  is smooth, differentiable, and smooth as summarized in the following lemma.

**Lemma 2** (Bounded smoothness). *Let  $F_\mu$  be smoothing of  $F$  as in Eq. (16). The function  $F_\mu$  is  $\frac{L}{1+\mu L}$ -smooth.*

*Proof.* In order to show this we will use the definition in Eq. (15) and the fact that convex conjugate has the property that a conjugate of  $s$ -strongly convex function is  $1/s$ -smooth and vice versa (Borwein and Zhu, 2005).

Notice that  $F^*$  is  $1/L$  strongly-convex, then  $F^* + \mu/2 \|\cdot\|$  is  $1/L + \mu$  strongly-convex. Conjugating these cause the function to be  $\frac{1}{1/L+\mu} = \frac{L}{1+\mu L}$  smooth. As  $L \rightarrow \infty$ , the smoothness constant is  $1/\mu$ .  $\square$

As a corollary of the above definition, we also have a bound on the gradient that we will utilize later.

**Lemma 3** (Bounded gradient<sup>3</sup>). *Let  $F_\mu$  be smoothing of  $F$  as in Eq. (16), then*

$$\|\nabla F_\mu(\eta)\|_p \leq \|\nabla F(\eta)\|_p \quad (17)$$

for any  $\eta \in \mathcal{P}$ , and  $p \in [1, \infty]$ .

*Proof.* To prove this relation, we use representation in Eq. (16). Note that  $\nabla F_\mu(\eta) = \frac{1}{\mu}(\eta - \zeta^*)$ , where  $\zeta^*$  is where the infimum is realized. In particular, it holds that  $\nabla F(\zeta^*) + \frac{1}{\mu}(\zeta^* - \eta) = 0$ , i.e.  $\zeta^* = (\mathbf{I} + \mu \nabla F)^{-1}(\eta)$ . The  $\mathbf{I}$  designates the identity operator.

Now,

$$\begin{aligned} \|\nabla F_\mu(\eta)\|_p &= \frac{1}{\mu} \|\eta - \zeta^*\|_p \\ &= \frac{1}{\mu} \|\eta - (\mathbf{I} + \mu \nabla F)^{-1} \eta\|_p \\ &= \frac{1}{\mu} \|(\mathbf{I} + \mu \nabla F)^{-1} ((\mathbf{I} + \mu \nabla F)\eta - \eta)\|_p \\ &\leq \frac{1}{\mu} \|((\mathbf{I} + \mu \nabla F)\eta - \eta)\|_p \\ &= \|\nabla F(\eta)\|_p, \end{aligned}$$

where in the fourth line we use that resolvent is non-expansive.  $\square$

### A.3 A-design and D-design regularity

The Assumption 1 is satisfied for the objectives in Table 1 with the exception of E-design which is non-smooth. Using the smoothing technique above one can make sure the conditions are satisfied for the optimization algorithm used in this work.

The proofs of the smoothness and convexity of A and V-design can be found in Appendix, Lemma 8 of Borsos et al. (2020). The convexity of D-design is clear from Pukelsheim (2006), the objective is also smooth in the regularized form, however, the constant has poor scaling. The following following lemma demonstrates it.

**Lemma 4.** *The objective  $-\log \det (\sum_{x \in \mathcal{X}} xx^\top \eta(x) + \lambda \mathbf{I})$  is  $L$ -smooth in terms of  $\eta \in \Delta_{|\mathcal{X}|}$ , with  $L = \frac{1}{\lambda^2} \lambda_{\max}((\mathbf{X}\mathbf{X}^\top) \circ (\mathbf{X}\mathbf{X}^\top))$ , and additionally the  $\lambda_{\max}(\mathbf{H}(\eta)) \leq \|\nabla F(\eta)\|_2^2$ , where  $\mathbf{H}$  is the Hessian.*

<sup>3</sup>Special thanks to Ya-Ping Hsieh for the proof of this lemma.

*Proof.* The Hessian of the above objective is  $\mathbf{H} = (\mathbf{X}\mathbf{V}^{-1}\mathbf{X}^\top) \circ (\mathbf{X}\mathbf{V}^{-1}\mathbf{X}^\top)$ , where  $\mathbf{V} = \mathbf{X}^\top \mathbf{D}(\eta)\mathbf{X} + \lambda\mathbf{I}$  and  $\circ$  refers to the Hadamard product. This can be shown by taking the derivative, linearity of trace, and  $\partial \mathbf{A}^{-1} = \mathbf{A}^{-1}(\partial \mathbf{A})\mathbf{A}^{-1}$  for symmetric matrix.

The last part is by noting that all elements of  $\mathbf{H}_{ij}$  are positive, and strictly smaller than  $\nabla_i F \nabla_j F$ . The result follows by using Theorem 5.22 from (Zhang, 2011).  $\square$

## B Formal Results and Proofs

### B.1 Concentration Results

First, we show that empirical measures constructed via sampling from a probability distribution can closely track its probability distributions using Azuma inequality. We will use this result in the convergence proofs later in Appendix B.4.

**Lemma 5** (Empirical measure concentration (linear)). *Let  $\{\eta_t\}_{t=1}$  be an adapted sequence of probability distributions on  $\mathcal{X}$ ,  $\mathcal{P}(\mathcal{X})$  with respect to filtration  $\mathcal{F}_{t-1}$ . Likewise let  $\{f_t\}_{t=1}$  be an adapted sequence of linear functionals  $f_t : \mathcal{P}(\mathcal{X}) \rightarrow \mathbb{R}$  s.t.  $\|f_t\|_\infty \leq B_t$ . Also, let  $x_t \sim \eta_t$ , and  $\delta_t(x) = \mathbf{1}_{x_t=x}$ , then*

$$\mathbb{P} \left( \left| \sum_{t=1}^T f_t(\delta_t - \eta_t) \right| \geq \sqrt{2 \sum_{t=1}^T B_t^2 \log \left( \frac{2}{\delta} \right)} \right) \leq \delta \quad (18)$$

*Proof.* Let  $Q_t = f_t(\delta_t - \eta_t)$ . In other words  $Q_t = \int_{x \in \mathcal{X}} f_t(x)(\delta_t(x) - \eta_t(x))dx$ . Since,

$$\begin{aligned} \mathbb{E}[Q_t | \mathcal{F}_{t-1}] &= \mathbb{E}[f_t(\delta_t(x) - \eta_t(x)) | \mathcal{F}_{t-1}] \stackrel{\text{lin.}}{=} f_t(\mathbb{E}[\delta_t(x) - \eta_t(x) | \mathcal{F}_{t-1}]) \\ &= f_t(\eta_t(x) - \eta_t(x)) = f_t(0) = 0. \end{aligned} \quad (19)$$

and each  $Q_t$  is bounded as  $|Q_t| \leq \|f_t\|_\infty \|\delta_t - \eta_t\|_1 \leq \|f_t\| \leq B_t$ ,  $Q_t$  it is martingale difference sequence. Consequently, we can use the generalized Azuma-Hoeffding (Wainwright, 2019, Corr. 2.20) inequality to control the sum,

$$\mathbb{P} \left( \left| \sum_{t=1}^T Q_t \right| \geq \epsilon \right) \leq 2 \exp \left( -\frac{2\epsilon^2}{4 \sum_{t=1}^T B_t^2} \right),$$

Setting  $\delta/2 = \exp \left( -\frac{2\epsilon^2}{4 \sum_{t=1}^T B_t^2} \right)$ , gives  $\epsilon^2 = 2 \log(2/\delta) \sum_{t=1}^T B_t^2$ .  $\square$

**Lemma 6** (Hoeffding on Hilbert space (Lugosi, 2009)). *Let  $X_i \in L_2$  be zero mean s.t.  $\|X_i\|_2 \leq s$ , then for any  $t \geq \sqrt{Ns^2}$*

$$P(\|S_N\| \geq t) \leq \exp(-(t - \sqrt{Ns^2})^2 / (2(Ns^2)))$$

Pick  $t = 2\sqrt{Ns^2 \log(1/\delta)}$ , then  $P(\|S_N\| \geq \sqrt{2Ns^2 \log(1/\delta)}) \leq \exp(-\log(1/\delta)) = \delta$

### B.2 Resampling convergence: Proofs

First, we prove the saturation of the resampling, NON-ADAPTIVE, algorithm. Namely, given a policy  $\pi^*$  which induces the optimal  $d^*$  and hence optimal  $\eta^*$ , how quickly does the empirical  $\eta_T = \frac{1}{T} \sum_{t=1}^T \delta_{\tau_t}$  converge to  $\eta^*$ ? The proof trivially uses Lemma 5 and 6, to show concentration.

**Proposition 1.** *Under Assumption 1, for resampling oracle with  $T$  resamplings, and  $\eta_T = \frac{1}{T} \sum_{t=1}^T \delta_{\tau_t}$ ,*

$$F(\eta_T) - F(\eta^*) \leq \|\nabla F(\eta_T)\|_\infty \sqrt{\frac{2 \log(2/\delta)}{T}}$$

with probability  $1 - \delta$ . For  $D$ -design, we can show that

$$\mathbb{E}[F(\eta_T)] - F(\eta^*) \leq \frac{4\|\nabla F(\eta_T)\|_\infty^2}{T} \log(T) + 2$$

where the expectation is over the sampling process.

*Proof.*

$$F(\eta_T) - F(\eta^*) \stackrel{\text{convexity}}{\leq} \nabla F(\eta_T)^\top (\eta^* - \eta_T) = \frac{1}{T} \sum_{i=1}^T \nabla F(\eta_T)^\top (\eta^* - \delta_{\tau_i})$$

Using  $\|\nabla F(\eta_T)\|_\infty = B$ , and Lemma 5, with  $B_t = B/T$ , we get

$$F\left(\frac{1}{T} \sum_{t=1}^T \delta_t\right) - F(\eta^*) \leq \sqrt{4 \sum_{t=1}^T \frac{B^2}{T^2} \log(2/\delta)}$$

To show the bound in expectation,

$$\begin{aligned} F(\eta_T) - F(\eta^*) &\leq \nabla F(\eta^*)^\top (\eta_T - \eta^*) + \frac{L_t}{2} \left\| \left( \frac{1}{T} \sum_{t=1}^T \delta_t \right) - \eta^* \right\|^2 \\ &\stackrel{\text{Lemma 4}}{\leq} |\nabla F(\eta^*)^\top (\eta_T - \eta^*)| + \frac{\|\nabla F(z)\|_2^2}{2} \left\| \left( \frac{1}{T} \sum_{t=1}^T \delta_t \right) - \eta^* \right\|^2 \\ \mathbb{E}[F(\eta_T)] - F(\eta^*) &\leq \frac{B^2}{T^2} \mathbb{E} \left[ \left\| \sum_{t=1}^T (\delta_t - \eta^*) \right\|^2 \right] \\ &\stackrel{\text{Lemma 6}}{\leq} B^2 T^2 (2\sqrt{T \log(1/\delta)})^2 (1 - \delta) + \delta 2T \\ &\leq \frac{4B^2}{T} \log(T) + 2 \end{aligned}$$

where in the last step we used  $\delta = 1/T$ , and in the second to last we used that  $\mathbb{E}[\eta_T] = \eta^*$ .  $\square$

Note that the value depends on the constant  $\|\nabla F(\eta_T)\|_\infty$ . We cannot globally control the value of the gradient, at least with a satisfactory constant, but for sufficiently large  $T$ , this value will be well-behaved. The following section B.3 provides intuition when this number becomes well-behaved.

### B.3 Discussion on the regularity of the objective

The convergence rates in Theorems 1 and 2 depend on the norm of the gradient  $\|\nabla F(\eta_t)\|_\infty$  and local smoothness parameter  $L_t$ . Likewise, after  $t$  resamplings the suboptimality of  $\eta_t$  depends on it as in Proposition 1. In this section, we would like to provide some intuition about these quantities.

To explore these quantities, we will use a completely degenerate example, where the Markov chain has  $d$  states and  $d$  action, where playing  $i$ th action leads to  $i$ th state, and the agent has to stay there as  $H = 1$ . This corresponds to the classical experimental design problem where the Markov chain is trivial. On top of this, we assume that each  $\Phi(x_i, a_i) = e_i$  where  $e_i$  is unit vector in  $\mathbb{R}^d$ , and we focus on D-design objective.

In this case, the optimal policy  $\eta^*$  puts equal mass on all trajectories, i.e.  $P(\tau = \tau_i) = 1/d$  for all  $i \in [d]$ .

**Resampling** The gradient in this case is equal to:

$$\|\nabla F(\eta_t)\|_\infty = \max_{i \in [d]} e_i^\top \left( \sum_{j=1}^t e_{\tau_j} e_{\tau_j}^\top + \frac{\lambda}{T} \mathbf{I} \right)^{-1} e_i.$$

In this case, since the feature of each state-action is orthogonal to each other, the above is proportional linearly to  $T$  only if  $\{\tau_j\}_{j=1}^t$  does not span the whole basis. In order to do so, we need to sample each trajectory at least once. However, since the sampling is with replacement, we know due to classical *coupon collector problem* we need to perform in expectation  $d \log(d)$  resamplings in order to have each of them single. One can also express this using a high-probability event. Namely, let  $T_{\text{all}}$  be the time after all are sampled at least once. The probability  $P(T_{\text{all}} \leq d \log(d) + d \log(1/\delta)) \geq 1 - \delta$ . Notice that  $d \log(d)$  appears regardless of confidence score  $\delta \in (0, 1)$ .

**Adaptive method** Notice in contrast that e.g. ONE-STEP or EXACT algorithm would always sample a different  $e_{\tau_j}$  than previously collected in their next episode since they would always pick  $\tau_j$  which maximizes the gradient (or any of them). Hence, already with  $d$  steps, *deterministically*, in this example, would lead to a bound on the gradient as  $\frac{d}{1+\lambda/T} \leq d$ . In subsequent steps, the value of the gradient can grow, however, it does not grow linearly with  $T$ . Notice that since the smoothness constant of the D-objective is related to the gradient squared due to Lemma 4, after  $d$  steps even the local smoothness constant is bounded by  $d^2$  in this example.

This reveals that while the resampling and adaptive optimization have the same convergence in terms of  $T$ , in expectation or with high probability, their actual performance depends significantly on the auxiliary constants such as gradient norm or local Lipschitz constant.

Generalizing this procedure to more complicated feature spaces and Markov chain structure is a challenging problem. The above simplification suggests that adaptive methods perform some form of efficient initialization scheme. The closest in literature is the work of Todd (2016) which provides an initialization scheme for D-design such that the gradient is bounded. They consider only objectives without regularization. Extending this technique to a kernelized setting is non-trivial and perhaps not as practically appealing as the methods suggested here, especially in the regularized scenario that we consider. Detailed analysis and bounding of these constants in more general settings is an open problem for future work.

#### B.4 Adaptive Method: Proofs

In this section, we give proof of our adaptive methods. The proofs are related and essentially the same as those of Berthet and Perchet (2017) for Frank-Wolfe UCB albeit the source of error in gradients is different and handled differently. Due to the connection to Frank-Wolfe, we also know that the rate in expectation cannot be improved with this step size (Lacoste-Julien and Jaggi, 2015). We present ONE-STEP and EXACT variant in the following two theorems.

**Theorem 3** (Convergence high probability). *Under Assumption 1, using Nesterov (2005) smoothing technique as in Appendix A.2 with  $\mu = \sqrt{\log T/T}$ , the suboptimality of ONE-STEP and EXACT variant can be bounded as,*

$$F(\eta_T) - F(\eta^*) \leq \frac{1}{T}(F(\eta_1) - F(\eta^*)) + 2\sqrt{\frac{\log T}{T^3}} + \frac{1}{T}\sqrt{2\sum_{t=1}^T \|\nabla F(\eta_t)\|_\infty^2 \log\left(\frac{2}{\delta}\right)} + 3\sqrt{\frac{\log T}{T}} + \frac{1}{T}\sum_{t=1}^T \Delta_t$$

with probability  $(1 - \delta)$  over the stochasticity of policies and the Markov chain. For ONE-STEP variant  $\Delta_t = 0$  for all  $t$ , and for EXACT variant each subproblem is solved to near-optimality s.t.  $\tilde{v}_t$  approximates

$$v_t = \arg \min_{q \in \mathcal{P}} F_\mu \left( \eta_t + \frac{1}{t+1}(q - \eta_t) \right), \text{ and } F_\mu(\tilde{v}_t) - F_\mu(v_t) \leq \Delta_t.$$

*Proof.* Let us refresh the definition of the following terms,

$$q_t = \arg \min_{q \in \mathcal{P}} \nabla F_\mu(\eta_t)^\top q \tag{21}$$

$$\eta_\mu^* = \arg \min_{q \in \mathcal{P}} F_\mu(q), \quad \text{and} \quad \eta^* = \arg \min_{q \in \mathcal{P}} F(q). \tag{22}$$

We will first show the proof of ONE-STEP variant and later show how EXACT can be reduced such that the same analysis applies.

$$\begin{aligned} F_\mu(\eta_{t+1}) &= F_\mu \left( \eta_t + \frac{1}{t+1}(\delta_t - \eta_t) \right) \\ &\stackrel{L_\mu\text{-smooth}}{\leq} F_\mu(\eta_t) + \frac{1}{t+1} \nabla F_\mu(\eta_t)^\top (\delta_t - \eta_t) + \frac{L_\mu}{2(1+t)^2} \|\delta_t - \eta_t\|_2^2 \\ F_\mu(\eta_{t+1}) &\stackrel{\text{bounded}}{\leq} F_\mu(\eta_t) + \frac{1}{t+1} \nabla F_\mu(\eta_t)^\top (\delta_t - \eta_t) + \frac{L_\mu}{(1+t)^2} \\ &= F_\mu(\eta_t) + \frac{1}{t+1} \nabla F_\mu(\eta_t)^\top (q_t - \eta_t) + \frac{1}{t+1} \nabla F_\mu(\eta_t)^\top (-q_t + \delta_t) + \frac{L_\mu}{(1+t)^2} \\ &\stackrel{(21)}{\leq} F_\mu(\eta_t) + \frac{1}{t+1} \nabla F_\mu(\eta_t)^\top (\eta_\mu^* - \eta_t) + \frac{1}{t+1} \nabla F_\mu(\eta_t)^\top (-q_t + \delta_t) + \frac{L_\mu}{(1+t)^2} \end{aligned}$$



$$\begin{aligned}
 & \stackrel{\text{convexity}}{\leq} F_\mu(\eta_t) - \frac{1}{t+1}(F_\mu(\eta_t) - F_\mu(\eta_\mu^*)) + \frac{1}{t+1}\nabla F_\mu(\eta_t)^\top(-q_t + \delta_t) + \frac{L_\mu}{(1+t)^2} \\
 F_\mu(\eta_{t+1}) - F_\mu(\eta_\mu^*) & \leq F_\mu(\eta_t) - F_\mu(\eta_\mu^*) - \frac{1}{t+1}(F_\mu(\eta_t) - F_\mu(\eta_\mu^*)) + \frac{1}{t+1}\nabla F_\mu(\eta_t)^\top(-q_t + \delta_t) + \frac{L_\mu}{(1+t)^2} \\
 & \leq \frac{t}{1+t}(F_\mu(\eta_t) - F_\mu(\eta_\mu^*)) + \frac{1}{t+1}\nabla F_\mu(\eta_t)^\top(-q_t + \delta_t) + \frac{L_\mu}{(1+t)^2}
 \end{aligned}$$

where we used the shorthand  $\epsilon_t = \nabla F_\mu(\eta_t)^\top(-q_t + \delta_t)$ . Using shorthand  $\rho_{t+1} = F_\mu(\eta_{t+1}) - F_\mu(\eta_\mu^*)$ :

$$\begin{aligned}
 (F_\mu(\eta_{t+1}) - F_\mu(\eta_\mu^*))(t+1) & \leq t(F_\mu(\eta_t) - F_\mu(\eta_\mu^*)) + \nabla F_\mu(\eta_t)^\top(-q_t + \delta_t) + \frac{L_\mu}{(1+t)} \\
 \rho_{t+1}(t+1) & \leq \left( t\rho_t + \epsilon_t + \frac{L_\mu}{1+t} \right) \\
 \rho_{t+1}(t+1) - \rho_t t & \leq \epsilon_t + \frac{L_\mu}{1+t} \\
 \sum_{t=1}^{T-1} \rho_{t+1}(t+1) - \rho_t t & \leq \sum_{t=1}^{T-1} \epsilon_t + L_\mu \log T \\
 T\rho_T - \rho_1 & \leq \sum_{t=1}^{T-1} \epsilon_t + L \log T \\
 \rho_T & \leq \frac{1}{T}\rho_1 + \frac{1}{T} \sum_{t=1}^{T-1} \epsilon_t + \frac{L_\mu \log T}{T}
 \end{aligned}$$

Lastly as, we are interested in suboptimality of the actual function  $\varrho_{t+1} = F(\eta_{t+1}) - F(\eta^*)$ , which can be bound as

$$\begin{aligned}
 \varrho_T & \leq \rho_T + 2\mu \leq 2\mu + \frac{1}{T}\rho_1 + \frac{1}{T} \sum_{t=1}^{T-1} \epsilon_t + \frac{L_\mu \log T}{T} \\
 & \stackrel{\text{Lemma 2}}{\leq} \frac{1}{T}\rho_1 + \frac{1}{T} \sum_{t=1}^{T-1} \epsilon_t + \frac{\log T}{T} \frac{1}{\mu} + 2\mu \\
 & \stackrel{\text{Lemma 5}}{\leq} \frac{1}{T}\rho_1 + \frac{1}{T} \sqrt{2 \sum_{t=1}^T \|\nabla F_\mu(\eta_t)\|_\infty^2 \log\left(\frac{2}{\delta}\right)} + \frac{\log T}{T} \frac{1}{\mu} + 2\mu \\
 & \stackrel{\mu = \sqrt{\log T/T}}{\leq} \frac{1}{T}\rho_1 + \frac{1}{T} \sqrt{2 \sum_{t=1}^T \|\nabla F_\mu(\eta_t)\|_\infty^2 \log\left(\frac{2}{\delta}\right)} + 3\sqrt{\frac{\log T}{T}} \\
 & \stackrel{\text{Lemma 3}}{\leq} \frac{1}{T}\rho_1 + \frac{1}{T} \sqrt{2 \sum_{t=1}^T \|\nabla F(\eta_t)\|_\infty^2 \log\left(\frac{2}{\delta}\right)} + 3\sqrt{\frac{\log T}{T}} \\
 & \leq \frac{1}{T}\varrho_1 + 2\sqrt{\frac{\log T}{T^3}} + \frac{1}{T} \sqrt{2 \sum_{t=1}^T \|\nabla F(\eta_t)\|_\infty^2 \log\left(\frac{2}{\delta}\right)} + 3\sqrt{\frac{\log T}{T}}
 \end{aligned}$$

This proves the theorem for ONE-STEP variant. Now we will focus on the EXACT variant. The exact variant uses,

$$v_t = \arg \min_{q \in \mathcal{P}} G_t(q) := \arg \min_{q \in \mathcal{P}} F_\mu \left( \eta_t + \frac{1}{t+1}(q - \eta_t) \right). \quad (23)$$

First, notice that due to the convexity of  $F_\mu$ ,  $G_t$  is convex as well. Also, note that  $\frac{1}{1+t}\nabla F_\mu(\eta_t) = \nabla G_t(\eta_t)$ . Using convexity,

$$\frac{1}{1+t}\nabla F_\mu(\eta_t)^\top(v_t - \eta_t) = \nabla G_t(\eta_t)^\top(v_t - \eta_t) \leq G_t(v_t) - G_t(\eta_t) = G_t(v_t) - F_\mu(\eta_t) \quad (24)$$

Following the similar analysis notice,

$$F_\mu(\eta_{t+1}) = F_\mu(\eta_t + \frac{1}{t+1}(\delta_t - \eta_t)) \quad (25)$$

$$\leq F_\mu(\eta_t) + \frac{1}{1+t} \nabla F_\mu(\eta_t)^\top (\delta_t - \eta_t) + \frac{L_\mu}{(1+t)^2} \quad (26)$$

$$\leq F_\mu(\eta_t) + \frac{1}{1+t} \nabla F_\mu(\eta_t)^\top (v_t - \eta_t) + \frac{1}{1+t} \nabla F_\mu(\eta_t)^\top (\delta_t - v_t) + \frac{L_\mu}{(1+t)^2} \quad (27)$$

$$\stackrel{(24)}{\leq} F_\mu(\eta_t + \frac{1}{t+1}(v_t - \eta_t)) + \frac{1}{1+t} \nabla F_\mu(\eta_t)^\top (\delta_t - v_t) + \frac{L_\mu}{(1+t)^2} \quad (28)$$

$$\stackrel{(23)}{\leq} F_\mu(\eta_t + \frac{1}{t+1}(q_t - \eta_t)) + \frac{1}{1+t} \nabla F_\mu(\eta_t)^\top (\delta_t - v_t) + \frac{L_\mu}{(1+t)^2} \quad (29)$$

where the last line follows due to  $v_t$  being a minimizer. The rest of the analysis is identical to the ONE-STEP variant.  $\square$

The consequence of this theorem is that if the gradient  $\|\nabla F(\eta_t)\|_\infty \approx \mathcal{O}(1)$  and  $\Delta_t \propto \mathcal{O}(t^{-1/2})$ , or zero, the overall complexity is of order  $\mathcal{O}\left(\sqrt{\frac{\log T}{T}}\right)$  limited by the concentration event. Now, in order not to be limited by the concentration event, we will look at the convergence in expectation.

**Theorem 4** (Convergence expectation). *Under Assumption 1, the convergence of ONE-STEP and EXACT variant satisfies*

$$\mathbb{E}[F(\eta_t) | \mathcal{F}_{t-1}] - F(\eta^*) \leq \frac{1}{t} (F(\eta_1) - F(\eta^*)) + \frac{1}{t} \sum_{k=1}^{t-1} \frac{L_{\eta_k, 1/k}}{(1+k)}$$

for  $t \leq T$ , where the expectation over stochasticity of policies  $\{\pi_k\}_{k=1}^t$ , and the Markov chain, and, where  $\mathcal{F}_{t-1}$  designates filtration up to time  $t-1$ . For ONE-STEP variant  $\Delta_k = 0$  for all  $k$ , and for EXACT variant each subproblem is solved to near-optimality s.t.  $\tilde{v}_k$  approximates

$$v_k = \arg \min_{q \in \mathcal{P}} F\left(\eta_k + \frac{1}{k+1}(q - \eta_k)\right), \text{ and } F(\tilde{v}_k) - F(v_k) \leq \Delta_k.$$

*Proof.* Following the proof of Theorem 3, we note that by taking expectation over  $\mathbb{E}[\epsilon_k] = \nabla F(\eta_k)^\top (-q_k + \mathbb{E}[\delta_k]) = 0$ . Notice that we apply the smoothness under Assumption 2 not due to Nesterov (2005) smoothing technique.

$$\begin{aligned} F(\eta_{k+1}) &= F\left(\eta_k + \frac{1}{k+1}(\delta_k - \eta_k)\right) \\ &\stackrel{(9)}{\leq} F(\eta_k) + \frac{1}{k+1} \nabla F(\eta_k)^\top (\delta_k - \eta_k) + \frac{L_{\eta_k, 1/k}}{2(1+k)^2} \|\delta_k - \eta_k\|_2^2 \\ &\leq F(\eta_k) + \frac{1}{k+1} \nabla F(\eta_k)^\top (\delta_k - \eta_k) + \frac{1}{k+1} \nabla F(\eta_k)^\top (-q_k + \delta_k) + \frac{L_{\eta_k, 1/k}}{(1+k)^2} \\ \mathbb{E}[F(\eta_{k+1})] &\leq F(\eta_t) + \frac{1}{k+1} \nabla F(\eta_k)^\top (q_k - \eta_k) + \frac{L_{\eta_k, 1/k}}{(1+k)^2} \\ &\leq F(\eta_t) + \frac{1}{k+1} \nabla F(\eta_k)^\top (\eta^* - \eta_k) + \frac{L_{\eta_k, 1/k}}{(1+k)^2} \\ &\stackrel{\text{convexity}}{\leq} F(\eta_t) - \frac{1}{k+1} (F(\eta_k) - F(\eta^*)) + \frac{L_{\eta_k, 1/k}}{(1+k)^2} \\ \mathbb{E}[F(\eta_{k+1})] - F(\eta^*) &\leq F(\eta_t) - F(\eta^*) - \frac{1}{k+1} (F(\eta_k) - F(\eta^*)) + \frac{L_{\eta_k, 1/k}}{(1+k)^2} \end{aligned}$$

Using shorthand  $\rho_{k+1} = \mathbb{E}[F(\eta_{k+1})] - F(\eta^*)$ ,

$$\rho_{k+1}(k+1) \leq k\rho_k + \frac{L_{\eta_k, 1/k}}{(1+k)}$$

$$\sum_{k=1}^{t-1} \rho_{k+1}(k+1) - k\rho_k \leq \sum_{k=1}^{t-1} \frac{L_{\eta_{k,1}/k}}{(1+k)}$$

$$\rho_t \leq \frac{1}{t}\rho_1 + \frac{1}{t} \sum_{k=1}^{t-1} \frac{L_{\eta_{k,1}/k}}{(1+k)}$$

The proof for the exact variant follows analogically as in the proof of Theorem 3.  $\square$

Notice that if  $L_{\eta_{k,1}/k}$  could be globally bounded the suboptimality decreases as  $\mathcal{O}(\log T/T)$ .

## C Robust and Uncertain objectives

### C.1 Robust and Uncertain Objectives: Motivation

The functional  $\mathbf{C}$  studied in the paper can itself depend on an unknown quantity, which we designate as  $\gamma$  as  $\mathbf{C}_\gamma$ , where  $\gamma \in \Gamma_t$ ,  $t \in [T]$ . For example, the value of  $\sigma_{a,x}$  as in Eq. (4) might not be known in advance. To deal with this complication, there are two approaches one can take:

- *Robust design*: take a supremum over the set  $\Gamma$ , and have a design that takes into account any possible values of  $\gamma$ . If  $F$  is convex then so is the supremum over the compact set.
- *Sequential design*: amend the objective  $F$  with the new value of the supremum if by executing a trajectory we can reduce the size of the set  $\Gamma_t$ . In the context of the example (4), we can learn the variance from repeated samples and update over confidence set over them  $\Gamma_t$ .

We give examples for both of these design approaches in the experimental Section 7.

### C.2 Sequential Design and Uncertain Objectives: Proofs

It turns out that we can be competitive to the *true* value of  $\gamma^* \in \Gamma_t$  in the set  $\Gamma_t$  if the set decreases with time  $t$ . Its decrease influences the rate and the convergence can be guaranteed if it decreases at least as  $1/\sqrt{t}$  if the size of  $\Gamma_t$  is measured as Euclidean diameter and the set  $\Gamma_t \subset \mathbb{R}^m$  as we show in the following assumption. We do not directly show that  $\Gamma_t$  decreases as this are different for each application and it might not be true in general.

**Assumption 2** (Regularity with respect to the unknown). *Under Assumption 1 suppose further that either*

1.  $F_\gamma(\eta)$  is Lipschitz in  $\gamma$  for EXACT variant:

$$|F_\gamma(\eta) - F_{\gamma'}(\eta)| \leq U \|\gamma - \gamma'\|_2 \quad (30)$$

where  $U$  is independent of  $\eta$ .

2.  $\nabla F_\gamma(\eta)$  is Lipschitz in  $\gamma$  for ONE-STEP variant:

$$\|\nabla F_\gamma(\eta) - \nabla F_{\gamma'}(\eta)\|_2 \leq U \|\gamma - \gamma'\|_2 \quad (31)$$

where  $U$  is independent of  $\eta$ .

The proof of the following theorem is very much inspired by Berthet and Perchet (2017), where this is essentially a Frank-Wolfe UCB algorithm. Note also that for the example of Poisson sensing from Sec. 7 we know that the set  $\|\gamma^* - \gamma_t\|_2$  decreases at least as  $1/\sqrt{t}$  since the objective is exactly designed to reduce the uncertainty of the  $f$  and hence  $\gamma$  and  $f$  are the same this must be true.

**Theorem 5** (Convergence for unknown). *Under Assumptions 1 and 2 suppose there exists  $F_{\gamma^*}$ , where  $\gamma^* \in \Gamma_t$  for all  $t \in [T]$  is unknown, then using smoothed target of  $F_t(\eta) = \sup_{\gamma \in \Gamma_t} F_\gamma(\eta)$ , denoted as  $F_{t,\mu}$ , with iteration varying smoothing parameter  $\mu = \sqrt{\log T/T}$  in each round  $t \in [T]$  of EXACT algorithm we can show*

$$F_{\gamma^*}(\eta_T) - F_{\gamma^*}(\eta^*) \leq \mathcal{O} \left( (U + B) \sqrt{\frac{\log(\frac{2}{\delta}) + \log T}{T}} \right). \quad (32)$$

Using oracle  $\min_{q \in \mathcal{P}} \inf_{\gamma \in \Gamma_t} \nabla F_{\gamma, \mu}(\eta_t)^\top q$ , where  $F_{\gamma, \mu}$  is the smoothed objective with  $\mu = \sqrt{\log T / T}$  in ONE-STEP algorithm we can show,

$$F_{\gamma^*}(\eta_T) - F_{\gamma^*}(\eta^*) \leq \mathcal{O} \left( (U + B) \sqrt{\frac{\log \left( \frac{2}{\delta} \right) + \log T}{T}} \right) \quad (33)$$

as long as  $\|\gamma^* - \bar{\gamma}_t\| \leq \mathcal{O}(1/\sqrt{t})$ , where  $\bar{\gamma}_t$  achieves the supremum value in  $\Gamma_t$ . The value  $B = \sqrt{\frac{1}{T} \sum_{t=1}^T \|\nabla F_{\gamma^*}(\eta_t)\|_\infty^2}$  denotes the average gradient of the objective. The statements hold with  $1 - \delta$  probability.

*Proof.*

**EXACT** In what follows we utilizing nearly identical proof technique as in Theorem 3. Namely, we continue where Eq. (29) left. Due to notation clutter, we drop the  $\mu$  dependence on in  $F$ ,

$$\begin{aligned} F_{\gamma^*}(\eta_{t+1}) &\leq F_{\gamma^*} \left( \eta_t + \frac{1}{1+t} (v_t - \eta_t) \right) + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t) + \frac{1/\mu}{(1+t)^2} \\ &\stackrel{\text{due to } \sup_\gamma}{\leq} F_{\bar{\gamma}_t} \left( \eta_t + \frac{1}{1+t} (v_t - \eta_t) \right) + U \|\bar{\gamma}_t - \gamma^*\|_2 + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t) + \frac{1/\mu}{(1+t)^2} \\ &\stackrel{(23)}{\leq} F_{\bar{\gamma}_t} \left( \eta_t + \frac{1}{1+t} (q_t - \eta_t) \right) + U \|\bar{\gamma}_t - \gamma^*\|_2 + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t) + \frac{1/\mu}{(1+t)^2} \\ &\stackrel{(9)}{\leq} F_{\bar{\gamma}_t}(\eta_t) + \frac{1}{1+t} \nabla F_{\bar{\gamma}_t}(\eta_t)^\top (q_t - \eta_t) + U \|\bar{\gamma}_t - \gamma^*\|_2 + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t) \\ &\quad + \frac{2/\mu}{(1+t)^2} \\ &\stackrel{(21)}{\leq} F_{\bar{\gamma}_t}(\eta_t) + \frac{1}{1+t} \nabla F_{\bar{\gamma}_t}(\eta_t)^\top (\eta^* - \eta_t) + U \|\bar{\gamma}_t - \gamma^*\|_2 + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t) \\ &\quad + \frac{2/\mu}{(1+t)^2} \\ &\stackrel{(9),(30)}{\leq} F_{\gamma^*}(\eta_t) - \frac{1}{1+t} (F_{\gamma^*}(\eta_t) - F_{\gamma^*}(\eta^*)) + 2U(1 + (1+t)^{-1}) \|\bar{\gamma}_t - \gamma^*\|_2 \\ &\quad + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t) + \frac{2/\mu}{(1+t)^2} \\ &\leq F_{\gamma^*}(\eta_t) - \frac{1}{1+t} (F_{\gamma^*}(\eta_t) - F_{\gamma^*}(\eta^*)) + 4U \|\bar{\gamma}_t - \gamma^*\|_2 \\ &\quad + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t) + \frac{2/\mu}{(1+t)^2} \end{aligned}$$

The rest of the proof is similar to Thm. 3, where  $\rho_t = F_{\gamma^*}(\eta_t) - F_{\gamma^*}(\eta^*)$ ,  $\epsilon_t = \nabla F_{\gamma^*}(\eta_t)^\top (-v_t + \delta_t)$ ,

$$\rho_{t+1}(t+1) \leq t\rho_t + 4U \|\bar{\gamma}_t - \gamma^*\|_2 + \epsilon_t + \frac{2/\mu}{(1+t)}$$

Summing both sides on  $t = 1$  to  $T - 1$ ,

$$\begin{aligned} T\rho_T - \rho_1 &\leq \sum_{t=1}^{T-1} 4U \|\bar{\gamma}_t - \gamma^*\|_2 + \epsilon_t + \sqrt{\frac{\log T}{T}} \\ \rho_T &\leq \frac{1}{T} \rho_1 + \frac{4U}{\sqrt{T}} + B \sqrt{\frac{2 \log \left( \frac{2}{\delta} \right)}{T}} + \sqrt{\frac{\log T}{T}} \end{aligned}$$

where in the last step we used the fact that the distance  $\|\bar{\gamma}_t - \gamma^*\| \leq 1/\sqrt{t}$ , and the Lemma 5 to bound the deviation of  $\sum_{t=1}^T \epsilon_t$ . Again, notice that due to the smoothing the value is off by  $\sqrt{\frac{\log T}{T}}$  which does correspond to the leading term.

**ONE-STEP** Dropping  $\mu$  from  $F$  as above and utilizing nearly identical proof technique as in Thm.3 we can show the following,

$$\begin{aligned}
 F_{\gamma^*}(\eta_{t+1}) &\stackrel{(21)}{\leq} F_{\gamma^*}(\eta_t) + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (q_t - \eta_t) + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-q_t + \delta_t) \frac{1/\mu}{(1+t)^2} \\
 &= F_{\gamma^*}(\eta_t) + \frac{1}{t+1} (\nabla F_{\gamma^*}(\eta_t) - \nabla F_{\bar{\gamma}_t}(\eta_t) + \nabla F_{\bar{\gamma}_t}(\eta_t))^\top (q_t - \eta_t) \\
 &\quad + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-q_t + \delta_t) + \frac{1/\mu}{(1+t)^2} \\
 &\stackrel{(31)}{\leq} F_{\gamma^*}(\eta_t) + \frac{1}{t+1} (\nabla F_{\bar{\gamma}_t}(\eta_t))^\top (q_t - \eta_t) + U \|\gamma^* - \bar{\gamma}_t\|_2 \|q_t - \eta_t\|_2 \\
 &\quad + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-q_t + \delta_t) + \frac{1/\mu}{(1+t)^2} \\
 &\stackrel{\text{update rule}}{\leq} F_{\gamma^*}(\eta_t) + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (\eta^* - \eta_t) + U \|\gamma^* - \bar{\gamma}_t\|_2 \|q_t - \eta_t\|_2 \\
 &\quad + \frac{1}{t+1} \nabla F_{\gamma^*}(\eta_t)^\top (-q_t + \delta_t) + \frac{1/\mu}{(1+t)^2}
 \end{aligned}$$

where in the last step, we used the update rule  $q_t = \arg \min_{q \in \mathcal{P}} \inf_{\gamma \in \Gamma_t} \nabla F_\gamma(\eta_t)^\top q$ . The problem as appears now is reduced to the proof of Theorem 3, as all terms feature only  $\gamma^*$ , and the only additional factor is  $\sum_{t=1}^{t-1} U \|\gamma^* - \bar{\gamma}_t\|_2 \|q_t - \eta_t\|_2 \leq 2U/\sqrt{t}$  upon summing as in Theorem 3, which is  $\mathcal{O}(U/\sqrt{T})$  due to the assumption in the theorem, which finished the proof.  $\square$

## D Further Considerations

### D.1 Linear MDPs: Density Oracle

If the system is not tabular, the formula given in Sec.4.1 to calculate  $d$  cannot be used. However, one can always estimate  $d$  via sampling which converges at a rate  $N^{-1/2}$  with a number of trajectory samples  $N$  given that we know the *simulator* - i.e. the transition matrix  $P$ .

We provide a specific way to calculate the density for recently introduced class of Markov chains called *linear MDPs* (Jin et al., 2020), which stipulate that  $P(x'|x, a) = \mu(x')^\top \psi(x, a) \pi(a|x)$ , where  $\cdot^\top$  designates an inner product in an Euclidean space, and  $\psi(x, a) \in \mathbb{R}^m$ , where  $m$  is the dimension of the feature space.

Linear MDPs provide a way to improve the scalability when  $\mathcal{X}$  is larger or even infinite. In this case, the transition matrix  $P_\pi$  with policy  $\pi$  is equal to  $P_\pi(x', x) = \sum_a \mu(x')^\top \psi(x, a) \pi(a|x) = \mu(x')^\top z_\pi(x)$ , where  $z_\pi(x) \in \mathbb{R}^m$  is the mean embedding of  $\psi(x, a)$  with probability distribution  $\pi(a|x)$ . Lets us define an operator  $\mathbf{U} : \mathcal{X} \rightarrow \mathbb{R}^m$  and  $\mathbf{V}_h : \mathcal{X} \rightarrow \mathbb{R}^m$ , then the equation in previous line can be written as  $P_{(\pi_i)_h} = \mathbf{U} \mathbf{V}_{h,i}^\top$ . Using that  $d_0 = \delta_{x_0}$  and the definition from above  $d_{\pi_i}(x) = \frac{1}{H} \sum_{h=1}^H \prod_{i=1}^h P_{(\pi_i)} d_0(x) = \frac{1}{H} \sum_{h=1}^H \mu(x) \left( \prod_{i=2}^h \mathbf{V}_{h,i}^\top \mathbf{U} \right) z(x_0) = \frac{1}{H} \sum_{h=1}^H \mu(x) \mathbf{M}_{h,i} z(x_0)$ , where  $\mathbf{M} \in \mathbb{R}^{m \times m}$ . Hence in order to estimate  $d_\pi(x)$  we only need to multiply  $m \times m$  matrices despite  $|\mathcal{X}|$  infinite. This allows us to operate in lazy-regime of evaluating  $d(x)$  at  $x$  we only need in efficiently. Alternatively one could use the algorithm of Jin et al. (2020) directly to solve the sub-problem which can be implemented as an efficient algorithm.

### D.2 Relationship to Submodular optimization

The objective  $F$  can be reformulated as set function  $H(S) : 2^T \rightarrow \mathbb{R}$  defined on subsets  $S = \{\tau_1, \dots, \tau_t\}$  of the set of all possible trajectories. In addition, if  $H$  is submodular, as is the case with  $\log \det$  objective (Krause and Guestrin, 2005), we can benefit from the  $(1 - e^{-1})$  approximation guarantee for the greedy algorithm discovered by Nemhauser et al. (1978) when solving cardinality constrained maximization  $\max_{|S| \leq T} H(S)$ . The objective  $H$  upon reformulation can be related to the objective studied in this work as  $H(S) = \log \det(\sum_{\tau \in S} I(\tau) + \lambda \mathbf{I}) = d \log(|S|) - F(\frac{1}{|S|} \sum_{i=1}^{|S|} \delta_{\tau_i})$ , where  $F(\eta) = -\log \det(\sum_{\tau \in T} \eta(\tau) I(\tau) + \lambda \mathbf{I}/|S|)$ . The objective values are only directly relatable only when  $|S| = T$  due to different way of handling regularization.

Nevertheless, we can show that greedy selection from the ground formed by the power set of  $\mathcal{T}$ ,  $2^{\mathcal{T}}$  defines as:

$$\tau_t = \arg \max_{\tau \in \mathcal{T}} H(S \cup \{\tau\})$$

leads to the same update as if we optimized the convex-relaxation used within our framework,

$$\arg \min_{\tau \in \mathcal{T}} F \left( \frac{1}{|\mathcal{S}| + 1} \left( \sum_{i=1}^{|\mathcal{S}|} \delta_{\tau_i} + \delta_{\tau} \right) \right),$$

with the specially chosen  $\frac{1}{1+t}$  step-size (as used in this work). The objectives cannot be easily compared, and hence greedy or convex optimization guarantees are different with the two viewpoints, but both frameworks lead to the *same solutions*, i.e. same selection of trajectories, in the end when convex-RL is run with the varying regularization constant  $\lambda_t = \lambda/t$ ; a minor technicality.

**Stochastic set cover** However, in our case, we cannot pick trajectories exactly (unless the system is deterministic). We pick distributions over  $\mathcal{T}$ ,  $q \in \mathcal{P}$ . In particular a trajectory  $\tau \sim q$  is generated from  $q$  (or likewise associated  $\pi$ ), and hence the objective we study is  $\tilde{H}(\{\tau_1, \dots\}) = \mathbb{E}_{\tau_i \sim \pi_i} [H(\{\tau_i, \dots, \tau_T\})]$  where the ground set changes to the *set of all policies*. Submodularity is preserved upon nonnegative linear combinations (see [Krause and Golovin \(2014\)](#)), hence the objective  $\tilde{H}$  as a function of policies is still submodular. In fact, this formulation exactly coincides with the stochastic set cover problem ([Golovin and Krause, 2011](#)). However the parallel to convex allocation used in this work is more complicated now. Notice that the greedy step of choosing a policy  $\pi_{t+1}$  (equivalently  $q_t$ ) to sample as an addition to already chosen trajectories  $\{\tau_1, \dots, \tau_t\}$  due to  $\{q_1 \dots, q_t\}$  is solving the following objective

$$q_{t+1} = \arg \max_{q \in \mathcal{P}} = \mathbb{E}_{\tau_{t+1} \sim q} [H(\{\tau_1, \dots\} \cup \{\tau_{t+1}\})].$$

If we were to look at the corresponding greedy update rule, where the convex measures are augmented by a single step (with the step size  $1/(1+t)$ ) leading to the same update in terms of elements we would arrive at:

$$q_{t+1} = \arg \min_{q \in \mathcal{P}} \mathbb{E}_{\tau_{t+1} \sim q} \left[ F \left( \frac{1}{t+1} \left( \sum_{i=1}^t \delta_{\tau_i} + \delta_{\tau_{t+1}} \right) \right) \right]. \quad (34)$$

Again the objective values are different but the sequence of elements chosen is the same. This objective is, however, different to the oracle we are using in Alg. 1 step 2. Notice that our oracle looks for the best  $q$  that minimizes

$$F \left( \frac{t}{t+1} \eta_t + \frac{1}{1+t} q \right) = F \left( \frac{1}{1+t} \left( \sum_{i=1}^t \delta_{\tau_i} + q \right) \right) = F \left( \frac{1}{1+t} \left( \sum_{i=1}^t \delta_{\tau_i} + \mathbb{E}_{\tau_{t+1} \sim q} [\delta_{\tau_{t+1}}] \right) \right),$$

which is related to the above greedy marginal gain via Jensen inequality. Unfortunately, the gap between these two can be large, and hence our method can be seen as a heuristic approximation to the greedy algorithm without an explicit guarantee on the greedy oracle. Only in the case when  $q \in \mathcal{P}$  where  $\mathcal{P}$  is restricted to trajectory probabilities due to the deterministic policies, do the two methods coincide (as in the previous paragraph). This is because the expectation contains only one term and the Jensen gap is zero. This suggests that greedy formulation as in Eq. (34) is more powerful.

### D.3 Relationship to multi-agent systems

If we had  $T$  agents that we were to release jointly at the same time, or equivalently we would like to plan jointly for  $T$  episodes in one optimization step, we could increase the state-action space by assuming the new action space be  $(\mathcal{X} \times \mathcal{A}) \times (\mathcal{X} \times \mathcal{A}) \dots$ , where we do the product  $T$  times. This increases the action-space exponentially in  $T$ , and we denote it  $\tilde{\mathcal{X}} \times \tilde{\mathcal{A}}$ . Visiting the state  $x_i^{(t)}, a_i^{(t)}$  in episode  $t$  does not provide any different information and hence the observations for  $\Phi(x_i^{(t)}, a_i^{(t)}) = \Phi(x_i^{(t')}, a_i^{(t')})$  for all  $t, t' \in [T]$ . Hence, we are allowed to drop the time superscript.

Thus, if we were to consider the information matrix due to observing a states  $\{(x_i^{(t)}, a_i^{(t)})\}_{t=1}^T$  as a function of state-action visitation over  $\tilde{d}$  over  $\tilde{\mathcal{X}} \times \tilde{\mathcal{A}}$ , i.e.,  $\tilde{d} \in \tilde{\mathcal{D}}$

$$U(\tilde{d}) = s \left( \sum_{x_i, a_i \in \mathcal{X} \times \mathcal{A}} \Phi(x_i, a_i) \Phi(x_i, a_i)^\top \left( \sum_{t=1}^T \tilde{d}(x_i^{(t)}, a_i^{(t)}) \right) + \mathbf{I} \lambda \right)$$

$$= \frac{1}{T} s \left( \sum_{x_i, a_i \in \mathcal{X} \times \mathcal{A}} \Phi(x_i, a_i) \Phi(x_i, a_i)^\top \left( \frac{1}{T} \sum_{t=1}^T \tilde{d}(x_i^{(t)}, a_i^{(t)}) \right) + \mathbf{I} \frac{\lambda}{T} \right)$$

where  $s$  is the scalarization and  $\tilde{\mathcal{D}}$  is the average state-action polytope on  $\tilde{\mathcal{X}} \times \tilde{\mathcal{A}}$ . Upon this formulation, we can see the relation to the objective in Sec. 4.1. In fact, an solution that minimizes is such that  $d(x_i^{(t)}, a_i^{(t)}) = d(x_i, a_i)$  fixed for all  $t$ . Hence optimizing for multiple reruns jointly has two disadvantages a) blows up the state-action space combinatorially and b) does not adapt to the prior executed trajectories, and leads to a fixed policy that we know empirically converges slower.

## E Experiments: Further Information

In this section, we provide details of the experiments that we introduced in Sec. 7. Before we do so a couple of general comments. When optimizing over the polytope  $\mathcal{D}$ , we either use the exact polytope for small  $H$  as with the synthetic grid, but average-case polytope for large  $H$  due to faster implementation. This, from our experience, does not reduce the performance but provides a significant simplification in the code, and is a commonly made simplification found in other works such as Hazan et al. (2019). Secondly, we never use Nesterov (2005) smoothing technique, which is strictly not necessary since all our objectives are smooth only with globally large smoothness constants. Despite this, the algorithm performs very well suggesting the analysis is pessimistic in nature.

To run the experiments we used a smaller server-class machine with 28 CPU cores that we utilized for no more than 20 hours of active time. In general, this is a methodological paper and does not rely on any heavy calculation.

**Synthetic grid** In this experiment we use  $H = 20$ . The exact location of different unit vectors as described in Sec. 7 is visualized in Fig. 1 with different pictograms. The optimization is run with EXACT-method such that when the duality gap is below  $\epsilon = 0.05$  the optimization terminates. We used the exact line search with bisections to solve the line search problem. We always marginalize the policy before execution; except for TRACKING variant. The initial state is in the lower-left corner while the final is in the upper-right corner as in Fig. 1.

### E.1 Beilschmeida

First of all this dataset comes from the seminal work of Baddeley et al. (2015) and their R package. We closely follow the setup from Mutný and Krause (2021) to design the sensing problem. We assume a collection of Borel sets  $\mathcal{A}$ , s.t. each set  $A \in \mathcal{A}$ . They are all subsets of  $[-1, 1]^2$  in this case and are generated via hierarchical splitting of the domain. You can see all the sets in Fig. 4a, There are  $16^2$  of these sensing actions. The Poisson process we model has an associated *intensity function*  $\lambda(x)$  (plotted in Fig. 4b), where the number of events sensed during the unit time in  $A$  is distributed as,

$$N(A) \sim \text{Poisson} \left( \int_A f(x) dx \right). \quad (35)$$

We assume that  $f \in \mathcal{H}_k$ , and that  $f \geq 0$ . Given *number* of events in  $A$ ,  $n(A)$  for the duration period  $\Delta$  in  $t$  sessions we have  $\{(n(A_i), A_i)\}$  we use the heteroscedasdict least-squares estimator given the data which is defined via:

$$\hat{f}_t = \arg \min_{f \in \mathcal{H}_k, f \geq 0} \sum_{i=1}^t \frac{(\int_{A_i} f(x) dx - n(A_i))^2}{\sigma_i^2} + \frac{\gamma}{2} \|f\|_k^2. \quad (36)$$

where  $\sigma_i^2 = \int_{A_i} \bar{f}_{i-1}(x) dx$  is the upper confidence bound in the  $i$ th iteration. The bar denotes the element in the confidence set which maximizes the value of  $\sigma_i^2$  over all plausible  $f$ . For the construction of the confidence set please refer to the Mutný and Krause (2021).

Alternatively, we could use an absolute bound on  $\sigma_i^2$  as,  $\sigma_i \leq \int_{A_i} \bar{f}_{i-1}(x) dx \leq \text{vol}(A_i) \max_x |f(x)| \leq \text{vol}(A_i) \max_x k(x, x) \|f\|_{\mathcal{H}_k} \leq \text{vol}(A_i) B \kappa$ . The value of  $B = 840$  for this experiment. See  $\kappa$  below. Notice that this is a modeling parameter that depends on the volumes of the sets  $A_i$  and these are scaled to be  $[-1, 1]$  which makes this value seem large, but if integrated over such small sets the rate will be of order  $\mathcal{O}(1)$ . We use both approaches with known upper bound and with estimated  $\sigma_i^2$  in Fig. 3.

To model the  $\mathcal{H}_k$ , we use the squared exponential kernel that takes the slope  $s_{x,y}$  and height  $h_{x,y}$  of a point  $(x, y)$  as inputs, as these are predictive of the habitat of Beilschmiedia, as  $k((x, y), (x', y')) = \kappa \exp\left(-\frac{(s_{x,y} - s_{x',y'})^2 + (h_{x,y} - h_{x',y'})^2}{2\gamma}\right)$  where  $\gamma = 0.1$ , where  $\kappa = 1000$  to match the scaling of the domain to  $[-1, 1]$ . To relate this experiment to the grid-world example as in Figure 1 we can cluster the sectors in the  $(x, y)$  map due to the similarity, and close to ‘orthogonal’ regions in space

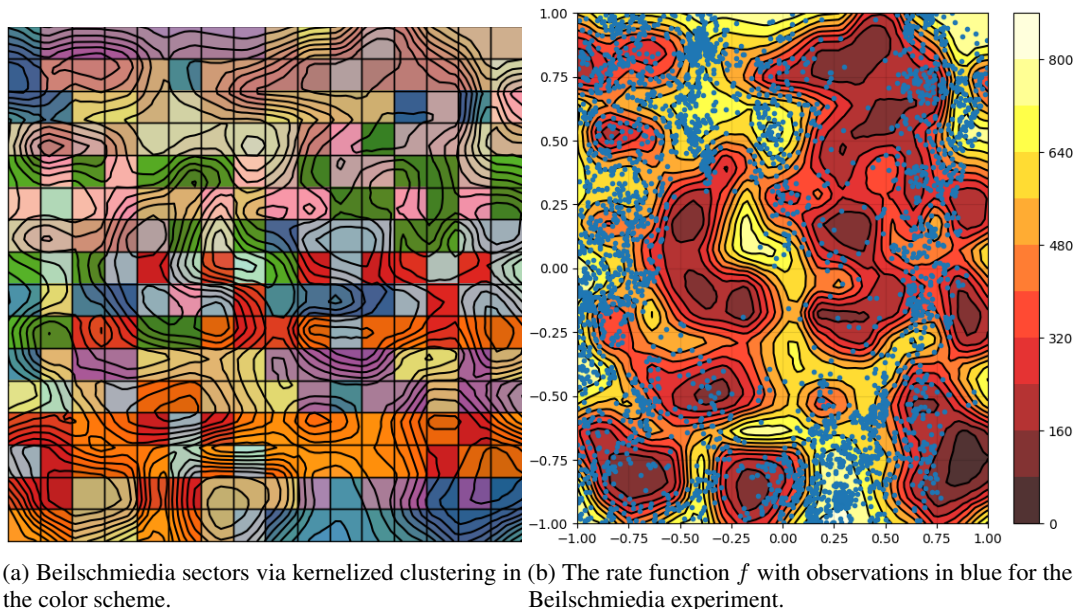


Figure 4: Beilschmidia details: In a) we report kernelized clustering due to the kernel  $k$  of map sensed sectors. The same color corresponds to the same group overlaid with the rate function  $f$ . This map connects this example to the motivating example of Fig. 1. b) Estimated rate function with the ground truth points.

will belong to a different cluster. We report this clustering in Fig. 4a with 12 clusters. These clusters are not used in the algorithm; this is just for visualization purposes and to relate it to the toy experiment. The action space is the same as for the grid-world example with the difference that the initial and final states are in the lower-left corner.

Notice also that due to RKHS assumption,  $\int_{A_i} f(x)dx = \int_{A_i} f^\top \Phi(x)dx = f^\top \varphi(A)$ , where the new kernel due to evaluation function  $\varphi(\cdot)$  is defined on Borel sets  $A \in \mathcal{A}$ . We approximate the kernel using the triangle basis (with 400 basis functions) as in (Mutný and Krause, 2021) to make sure that the positivity constraint can be implemented. We use their publicly available code for the fitting.

**Parameters of MARKOV-DESIGN** The episode lengths is set to  $H = 64$ ,  $T = 128$ ,  $|\mathcal{X}| = 16^2$ . We use D-design objective on the approximated  $f(x) = \Phi(x)^\top f$  on the 400-dim vector  $f$ , with  $\lambda = 1$ . The algorithm is run with line search and accuracy 0.05 for the convex-RL part. We used the sequential variant of the algorithm as in Appendix C.2 for the case when variances are estimated after each episode.

## E.2 Pharmacokinetics

This dataset is synthetic, however, requires only 3 numbers which are choice reasonably with accordance to the prior works on this subject as in Mutný and Krause (2022a). A similar problem occurs in Foster et al. (2021). One can study the details in the exhaustive book of Gabrielsson and Weiner (1995). The explicit treatment of patient specific-contamination is novel to the best of our knowledge. The grand goal of this experiment is to estimate the parameters of the linear differential equation which we do by first estimating the concentration levels of individual patients. We obtain data by sampling (drawing blood) at specific time intervals and estimating the concentration of medication in the blood.

For each patient  $i \in [T]$  we have the following oracle,

$$f_i(t) = c_b(t) + g_i(t) + \epsilon \quad \text{where} \quad \epsilon \sim \mathcal{N}(0, \sigma^2) \quad \text{and } t \text{ is time.}$$

where  $g_i \sim \text{GP}(0, k)$ , patient-specific random contamination, where  $k$ , is non-stationary variant of squared exponential kernel as  $k(t, t') = \kappa \exp(-3t) \exp(-3t') k_{SE, \gamma}(t, t')$  with  $\gamma = 0.1$ , where  $k_{SE}$  is the classical squared exponential kernel, and  $\kappa = 0.05$ . The small  $\kappa$  models the fact that the variation is small and the non-stationary ensures that the variation decreases with increasing time. The value of  $\sigma = 0.05$  is used as noise standard deviation.

The  $c_b$  denotes the concentration of a medication in the blood, which is shared among all patients. The  $c_s$  denotes the concentration of the medication in the stomach. You can see in Figure 5a that the green curve decreases over time while  $c_b$



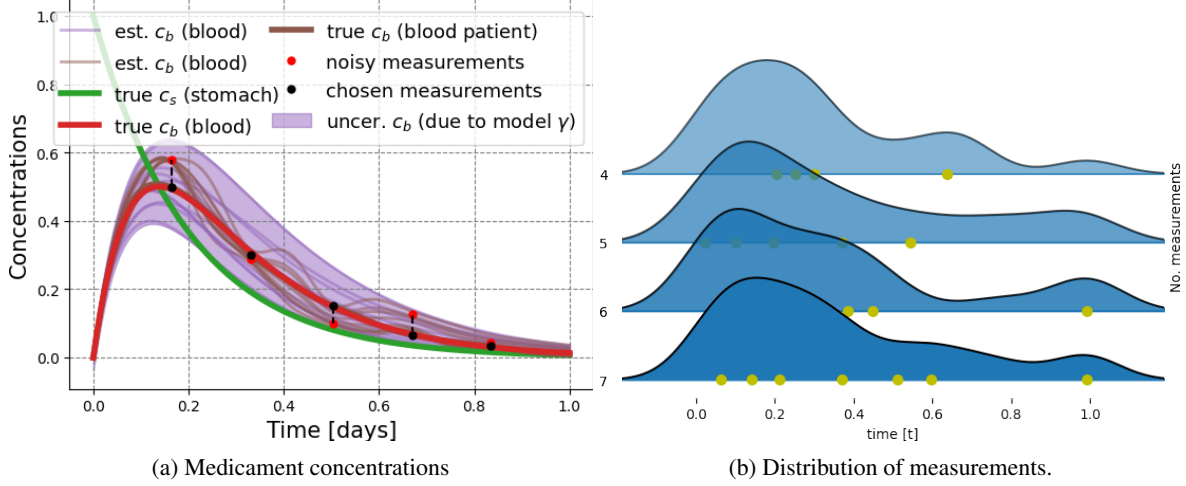


Figure 5: Pharmacokinetics: a) We plot the typical experimental measurements over time  $t$ . We see the two concentration values:  $c_s(t)$  in stomach and  $c_b(t)$  in blood. On top of that in brown, we see subject-specific values of  $f_i$  which deviate from purple  $c_b^\gamma$ , where dependence on  $\gamma$  is explicit, which correspond to possible  $c_b$  values due to different  $\gamma$  consistent with the data. The true blood concentration is in red. The uncertainty due to the unknown  $\gamma$  is in shaded regions, which contains all plausible explanations to the data that we observed in red points. The chosen measurements are not due to our algorithm here. b): This plots a smoothed distribution of measurement locations over  $T$  patients with different constraints on a total number of blood measurements. Notice that the measurements are denser in the initial stages of the experiment as a signal-to-noise ratio is much better. In yellow we depict one measurement plan for a patient (the first one). As one goes higher we vary the number of blood draws allowed in our Markov chain.

increases. The dynamics is governed by a *linear differential equation*:

$$\frac{d}{dt} \begin{pmatrix} c_s(t) \\ c_b(t) \end{pmatrix} - \begin{pmatrix} -a & 0 \\ b & -c \end{pmatrix} \begin{pmatrix} c_s(t) \\ c_b(t) \end{pmatrix} = 0 \quad (37)$$

We stack the coefficients  $\gamma = (a, b, c)$ , and denote the above operator as  $L_\gamma$  acting on  $(c_s(t), c_b(t))$ . We do not know the values of  $\gamma$  as we want to estimate them, however we know that they vary in a box constraint  $\Gamma$  which for this experiment is  $\Gamma = [4, 6] \times [9, 11] \times [9, 11]$ .

Now, note that as  $L_\gamma(c_s(t), c_b(t)) = 0$ , we see that the real solutions lie in the kernel of  $L_\gamma$ . If we discretize the operator over a time-horizon  $t \in [t_0, t_1]$  and enforce this constraint and apply it to  $c_b(t) = \Phi(t)^\top u_b$  (likewise  $c_s(t) = \Phi(t)^\top u_s$ ), where  $\Phi(t)$  is the evaluation functional of a kernel  $k$ , in this example corresponding to sufficiently rich kernel to reproduce any trajectory (in this case squared exponential as it is an universal kernel). We know that  $c_b$  is such that it has to be in the kernel of the operator of discretized operator  $\mathbf{L}_\gamma$ . We can stack the orthogonal rows of the kernel of discretized operator  $\mathbf{L}_\gamma$  to get  $\mathbf{C}_\gamma$ . Thus  $(u_b, u_s) = \mathbf{C}_\gamma^\top v$ , where  $v \in \mathcal{H}_k \setminus \text{span}(\mathbf{L}_\gamma)$ . Thus the only unknown of the trajectory is the  $\mathbf{C}_\gamma$  which corresponds to unknown initial conditions given a fixed  $\gamma$ .

We model the initial conditions of blood to be known  $c_b(0) = 0$ , and to be slightly unknown for stomach in order to have a well-defined problem  $(c_s(0) - c_{\text{dose}})^2 \leq \epsilon$ . For a fixed  $\gamma$  we can design an experiment that reduces the uncertainty in the initial condition the most. Despite the initial condition not being the core *unknown* problem here we use them as a tool to get an informative data collection scheme. So given  $\gamma$ , the objective would be A-design as minimization of  $\text{Tr} \left[ (\mathbf{C}_\gamma (\sum_{\tau \in \mathcal{T}} \eta(\tau) I(\tau) + (1/T) \mathbf{V}_0)^{-1} \mathbf{C}_\gamma^\top) \right]$ , where  $\mathbf{V}_0 = \mathbf{I} + [\Phi(0), 0][\Phi(0), 0]^\top$ , where the second term corresponds to prior on initial conditions of  $c_s$ . As  $\gamma$  is unknown we take  $\inf_\gamma$  over the above objective to get the overall objective which should yield a data collection scheme which is good for any  $\gamma \in \Gamma$ . The scheme we just explained were complete if there was not any contamination due to  $g_i$ . As this one is present, we can absorb it into noise as correlated noise,

$$f_i(t) = (\Phi(t))^\top (u_b) + \epsilon + \Phi_N(t)^\top g_i$$

where  $\Phi_N(t)$  is the evaluating functional of random element  $g_i$ . Stacking all the problem vectors together we have  $f_i = (u_s, u_b)$  and features  $\tilde{\Phi}(t) = (\Phi(t), \Phi(t))$ . In order to implement  $\tilde{\Phi}(t)$  for convenience of implementation we use high fidelity approximation squared-exponential kernel from Mutný and Krause (2018) known as *Quadrature Fourier Features*

(QFF) with  $\gamma = 0.05$  with  $m = 150$ . The  $\Phi(x)_N$  correspond to the non-stationary kernel we explained for patient-specific contamination.

We absorb the noise  $g_i$  into the information matrix by scaling the features properly. First, consider all times  $t$  (i.e. all states), corresponding to matrix  $\Sigma(t, t') = \Phi_N(t)^\top \Phi_N(t')$ , then we define, new features  $\Psi(t) = \sum_{t'} \Sigma^{-1/2}(t, t') \tilde{\Phi}_{t'}$ , which are in turn define the information matrix  $I(\tau) = \sum_{t \in \tau} \frac{1}{\sigma^2} \Psi(t) \Psi(t)^\top$  used in for this experiment. This makes sure that features are properly scaled to reflect the heteroscedastic noise due to  $g_i$ . A more detailed treatment of heteroscedastic noise is provided in [Kirschner and Krause \(2018\)](#), which inspired the definition of the information matrix in this work.

We show the smoothed distribution of measurement location in [Fig. 5b](#) and we see that the locations are distributed in the initial phases of the experiment as we would expect from the pharmacological perspective ([Gabrielsson and Weiner, 1995](#)). In this time frame the signal-to-noise ratio is higher than later when the concentration drops below a noise level. Having the trajectories estimated we can then run maximum likelihood estimation to identify the parameter gamma from the inferred trajectories given  $\gamma$  similarly as done in [Mutný and Krause \(2022a\)](#) or by [Foster et al. \(2021\)](#).

**Parameters of MARKOV-DESIGN** The episode lengths are set to  $H = 128$ ,  $T = 128$ ,  $|\mathcal{X}| = 128 \times 5 \times 3$ , where we allow 5 draws of blood in the experiment over the whole time duration, and the spacing between blood draws needs to be at least 3; in other words, we have to wait for 3 time-steps before we draw blood again. We use the A-design objective on the values approximated with linear functional  $C_\gamma$  as  $\gamma$  is not known we take the robust version as explained in [App.C](#), we use  $\lambda = 0.5$ . The algorithm is run with line search and accuracy  $10^{-8}$  due to different scaling of the objective ( $C_\gamma$  is small as it contains normalized rows). The domain  $t \in [0, 1]$  for defining  $C_\gamma$  is discretized with 128 points.