Safe Exploration for Optimization with Gaussian Processes

(extended version with supplementary material)

Yanan Sui YSUI@CALTECH.EDU
California Institute of Technology, Pasadena, CA, USA

Alkis Gotovos ALKISG@INF.ETHZ.CH
ETH Zurich, Zurich, Switzerland

Joel W. Burdick JWB@ROBOTICS.CALTECH.EDU
California Institute of Technology, Pasadena, CA, USA

Andreas Krause KRAUSEA@ETHZ.CH
ETH Zurich, Zurich, Switzerland

Abstract

We consider sequential decision problems under uncertainty, where we seek to optimize an unknown function from noisy samples. This requires balancing exploration (learning about the objective) and exploitation (localizing the maximum), a problem well-studied in the multi-armed bandit literature. In many applications, however, we require that the sampled function values exceed some prespecified "safety" threshold, a requirement that existing algorithms fail to meet. Examples include medical applications where patient comfort must be guaranteed, recommender systems aiming to avoid user dissatisfaction, and robotic control, where one seeks to avoid controls causing physical harm to the platform. We tackle this novel, yet rich, set of problems under the assumption that the unknown function satisfies regularity conditions expressed via a Gaussian process prior. We develop an efficient algorithm called SAFEOPT, and theoretically guarantee its convergence to a natural notion of optimum reachable under safety constraints. We evaluate SAFEOPT on synthetic data, as well as two real applications: movie recommendation, and therapeutic spinal cord stimulation.

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

1. Introduction

Many machine learning applications in areas such as recommender systems or experimental design need to make sequential decisions to optimize an unknown function. In particular, each decision leads to a stochastic reward with initially unknown distribution, while new decisions are made based on the observations of previous rewards. To maximize the total reward, one needs to solve the tradeoff between exploring different decisions and exploiting decisions currently estimated as optimal within a given set. In some applications, however, it is unacceptable to ever incur low rewards; rather, the reward of any sampled strategy must lie above some specified "safety" threshold.

Consider, for example, medical applications (e.g., rehabilitation), where physicians may choose among a large set of therapies, some of which may be novel. The effects of different therapies are initially unknown, and can only be determined through experimentation. Free exploration, however, is not possible, since some therapies might cause severe discomfort, or even physical harm to the patient. Oftentimes, the effects of similar therapies are correlated, hence, a feasible way to explore the space of therapies is to start from some therapies similar to those known to be safe. Proceeding this way, more and more choices can be established to be safe, facilitating further exploration. In our experiments, we address an instance of such a problem, where the goal is to choose stimulation patterns for epidurally implanted electrode arrays to aid rehabilitation of patients who have suffered spinal cord injuries. Challenges of this kind arise in learning robotic controllers by experimenting with the robot, where some parameters might lead to physical harm of the platform. Similarly, in recommender systems, we might wish to

avoid recommendations that are severely disliked by the user, an application we also consider in our experiments.

Related work. The tradeoff between exploration and exploitation has been extensively studied in the context of (stochastic) multi-armed bandit problems. These problems model sequential decision tasks, in which one chooses among a number of different decisions (arms), each associated with a stochastic reward with initially unknown distribution. Classic bandit algorithms usually aim to maximize the cumulative reward, while in the "best-arm identification" variant (Audibert et al., 2010), they seek to identify the decision of highest reward with the minimum number of trials. Since their introduction by Robbins (1952), bandit problems have been widely studied in many settings (see Bubeck & Cesa-Bianchi (2012) for an overview). A number of efficient algorithms build on the work of Auer (2002), and their key idea is to use *upper confidence bounds* to implicitly negotiate the explore-exploit tradeoff by optimistic sampling. This idea naturally extends to bandit problems with complex (or even infinite) decision sets under certain regularity conditions of the reward function (Dani et al., 2008; Kleinberg et al., 2008; Bubeck et al., 2008). Srinivas et al. (2010) show how confidence bounds can be used to address bandit problems with a reward function that is modeled by a Gaussian process (GP), a regularity assumption also commonly made in the Bayesian optimization literature (Brochu et al., 2010), which is closely related to best-arm identification. These approaches effectively optimize long-term performance by accepting low immediate rewards for the sake of exploration. While this compromise is acceptable in certain settings, it makes these techniques unsuitable for safety-critical applications. Another related problem, is that of active sampling for localizing level sets, that is, decisions where the objective crosses a specified threshold (Bryan et al., 2005; Gotovos et al., 2013). In general, these approaches sample both above and below the threshold, and, consequently, do not meet our safety requirements.

The problem of safe exploration has been considered in control and reinforcement learning (Hans et al., 2008; Gillula & Tomlin, 2011; Garcia & Fernandez, 2012). For example, Moldovan & Abbeel (2012) consider the problem of safe exploration in MDPs. They ensure safety by restricting policies to be ergodic with high probability, i.e., able to "recover" from any state visited. This is a more general problem, which comes at a cost—feasible safe policies do not always exist, algorithms are far more complex, and there are no convergence guarantees. In contrast, we restrict ourselves to the bandit setting, where decisions do not cause state transitions, which leads to simpler algorithms with stronger guarantees, even in the agnostic (non-Bayesian) setting.

Our contributions. We model a novel class of safe optimization problems as maximizing an unknown expected-reward function over the decision set from noisy samples. By exploiting regularity assumptions on the function, which capture the intuition that similar decisions are associated with similar rewards, we aim to balance exploration (learning about the function) and exploitation (identifying near-optimal decisions), while additionally ensuring safety throughout the process. This additional requirement leads to novel considerations, different from those addressed in the classic bandit setting, since exploration is now not only a means to reducing uncertainty about the function, but also crucial in expanding the set of decisions established as safe.

More concretely, we propose a novel algorithm, SAFEOPT, which models the unknown function as a sample from a Gaussian process (GP), and uses the predictive uncertainty to guide exploration. In particular, it uses confidence bounds to assess the safety of as yet unexplored decisions. We theoretically analyze SAFEOPT under the assumptions that (1) the objective has bounded norm in the Reproducing Kernel Hilbert Space associated with the GP covariance function, and (2) the objective is Lipschitz-continuous, which is guaranteed by many common kernels. We establish convergence of SAFEOPT to a natural notion of "safely reachable" near-optimal decision. We further evaluate SAFEOPT on two real-world applications: movie recommendation, and therapeutic stimulation of patients with spinal cord injuries.

2. Problem Statement

We consider sequential decision problems, where we seek to optimize an unknown reward function $f : D \to \mathbb{R}$ defined on a finite set of decisions D. Concretely, we pick a sequence of decisions (e.g., items to recommend, experimental stimuli) $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots \in D$, and, after each selection $\boldsymbol{x}_t$, acquire a noise-perturbed value of f, that is, we observe $y_t = f(\boldsymbol{x}_t) + n_t$ (e.g., user rating, stimulus response). Our goal is to identify a decision $\boldsymbol{x}^*$ of maximum reward f, akin to the problem of best-arm identification in multi-armed bandits (Audibert et al., 2010). Crucially, though, we additionally wish to ensure that, for all rounds t, it holds that $f(\boldsymbol{x}_t) \geq h$, where $h \in \mathbb{R}$ is a problem-specific *safety threshold*. We call decisions that satisfy the above condition *safe*. In our recommender systems example, we seek to identify items that are particularly liked by the user, while guaranteeing that we never propose items the user strongly dislikes. In our medical setting, we seek to find stimuli that are particularly beneficial to the rehabilitation process, while guaranteeing that no painful stimuli are applied. It is important to note that, since f is unknown, the set of safe decisions is initially unknown as well.

Regularity assumptions. Without any assumptions about f, this is clearly a hopeless task. In particular, without any knowledge of the function, we do not even know where to start our exploration. Hence, we first assume that, before starting the optimization, we are given a "seed" set $S_0 \subset D$ that contains at least one safe decision. This establishes starting points for our exploration. To be able to identify new safe decisions to consider for exploration, we need to make some further assumption about f. In what follows, we assume that D is endowed with a positive definite kernel function, and that f has bounded norm in the associated Reproducing Kernel Hilbert Space (RKHS, see Schölkopf & Smola (2002)).[1] This assumption allows us to model our reward function f as a sample from a Gaussian process (GP) (Rasmussen & Williams, 2006). A $GP(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}'))$ is a probability distribution across a class of "smooth" functions, which is parameterized by a kernel function $k(\boldsymbol{x}, \boldsymbol{x}')$ that characterizes the smoothness of f. We assume w.l.o.g. that $\mu(\boldsymbol{x}) = 0$, and that our observations are perturbed by i.i.d. Gaussian noise, i.e., for samples at points $A_T = [\boldsymbol{x}_1 \dots \boldsymbol{x}_T]^T \subseteq D$, we have $\boldsymbol{y}_t = f(\boldsymbol{x}_t) + n_t$, where $n_t \sim N(0, \sigma^2)$. (We will relax this assumption later.) The posterior over f is then also Gaussian with mean $\mu_T(\boldsymbol{x})$, covariance $k_T(\boldsymbol{x}, \boldsymbol{x}')$, and variance $\sigma_T^2(\boldsymbol{x}, \boldsymbol{x}')$, that satisfy,

$$\begin{aligned}
\mu_T(\boldsymbol{x}) &= \boldsymbol{k}_T(\boldsymbol{x})^T (\boldsymbol{K}_T + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{y}_T \\
k_T(\boldsymbol{x}, \boldsymbol{x}') &= k(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}_T(\boldsymbol{x})^T (\boldsymbol{K}_T + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{k}_T(\boldsymbol{x}') \\
\sigma_T^2(\boldsymbol{x}, \boldsymbol{x}') &= k_T(\boldsymbol{x}, \boldsymbol{x}),
\end{aligned}$$

where $\boldsymbol{k}_T(\boldsymbol{x}) = [k(\boldsymbol{x}_1, \boldsymbol{x}) \dots k(\boldsymbol{x}_T, \boldsymbol{x})]^T$ and $\boldsymbol{K}_T$ is the positive definite kernel matrix $[k(\boldsymbol{x}, \boldsymbol{x}')]_{\boldsymbol{x}, \boldsymbol{x}' \in A_T}$.

Furthermore, we assume that f is L-Lipschitz continuous with respect to some metric d on D. This is automatically satisfied, for example, when considering commonly used isotropic kernels, such as the Gaussian kernel, on D.

Optimization goal. Under the Lipschitz-continuity assumption, what is the best solution that *any* algorithm might be able to find? Suppose our observations were noise free. In this case, after exploring the decisions in our seed set S_0, we could establish any decision $\boldsymbol{x}$ as safe, if there existed a decision $\boldsymbol{x}' \in S_0$, such that $f(\boldsymbol{x}') - L \cdot d(\boldsymbol{x}', \boldsymbol{x}) \geq h$. Exploring these newly identified safe decisions would establish further decisions as safe, and so on. Unfortunately our knowledge of f comes from noisy observations, so even after experimenting with the same decision $\boldsymbol{x}$ repeatedly, we are not able to infer $f(\boldsymbol{x})$ exactly, but only up to some statistical confidence $f(\boldsymbol{x}) \pm \epsilon$. Based on this insight, we define the *one-step reachability* operator

$$\begin{aligned}
&R_\epsilon(S) := \\
&\quad S \cup \left\{ \boldsymbol{x} \in D \;\middle|\; \exists \boldsymbol{x}' \in S, f(\boldsymbol{x}') - \epsilon - L d(\boldsymbol{x}', \boldsymbol{x}) \geq h \right\},
\end{aligned}$$

which represents the subset of D that can be established as safe upon learning f up to absolute error at most ϵ within S. Clearly, it holds that $S \subseteq R_\epsilon(S) \subseteq D$. Similarly, we can define the n-step reachability operator by

$$R_\epsilon^n(S) := \underbrace{R_\epsilon(R_\epsilon \dots (R_\epsilon}_{n \text{ times}}(S)) \dots),$$

and its closure by $\bar{R}_\epsilon(S) := \lim_{n \to \infty} R_\epsilon^n(S)$. It is easy to see that no algorithm that is able to learn f only up to ϵ will ever be able to establish a decision $\boldsymbol{x} \in D \setminus \bar{R}_\epsilon(S_0)$ as safe. Hence, we cannot hope that any safe algorithm will be able to identify the global optimum $f^* = \max_{\boldsymbol{x} \in D} f(\boldsymbol{x})$. We consider, instead, our benchmark to be the ϵ-reachable maximum, defined as

$$f_\epsilon^* = \max_{\boldsymbol{x} \in \bar{R}_\epsilon(S_0)} f(\boldsymbol{x}). \tag{1}$$

Failure of naive approaches. There are a number of approaches for trading exploration and exploitation under the smoothness assumptions expressed via a GP. One such approach is the GP-UCB algorithm (Srinivas et al., 2010), which greedily chooses

$$\boldsymbol{x}_t = \operatorname*{argmax}_{\boldsymbol{x} \in D} \left(\mu_{t-1}(\boldsymbol{x}) + \beta_t^{1/2} \sigma_{t-1}(\boldsymbol{x}) \right) \tag{2}$$

for a suitable schedule of β_t. While this algorithm is guaranteed to achieve sublinear cumulative regret, it places no restrictions on the decisions sampled, and, as a result, neither theoretically guarantees safety, nor exhibits it in our experiments. This is symptomatic of typical multi-armed bandit approaches when applied to our problem. In the following, we present an efficient algorithm, SAFEOPT, which, under the aforementioned assumptions, for any $\epsilon > 0$, is guaranteed with high probability to identify a solution $\hat{\boldsymbol{x}}$, such that $f(\hat{\boldsymbol{x}}) \geq f_\epsilon^* - \epsilon$, while sampling only safe decisions. Furthermore, we provide a sample complexity bound on the number of iterations required to achieve this condition.

3. The SAFEOPT Algorithm

We start with a high-level description of SAFEOPT. The algorithm uses Gaussian processes to make predictions about f based on noisy evaluations, and uses their predictive uncertainty to guide exploration. To guarantee safety, it maintains an increasing sequence of subsets $S_t \subseteq D$ established as safe using the GP posterior. It never chooses a sample

[1] Note that for finite decision sets and any universal kernel, this assumption is automatically satisfied.

Algorithm 1 SAFEOPT

Input: sample set D,
GP prior (μ_0, k, σ_0),
Lipschitz constant L,
seed set S_0,
safety threshold h
$C_0(\boldsymbol{x}) \leftarrow [h, \infty)$, for all $\boldsymbol{x} \in S_0$
$C_0(\boldsymbol{x}) \leftarrow \mathbb{R}$, for all $\boldsymbol{x} \in D \setminus S_0$
$Q_0(\boldsymbol{x}) \leftarrow \mathbb{R}$, for all $\boldsymbol{x} \in D$
for $t = 1, \ldots$ **do**
  $C_t(\boldsymbol{x}) \leftarrow C_{t-1}(\boldsymbol{x}) \cap Q_{t-1}(\boldsymbol{x})$
  $S_t \leftarrow \bigcup_{\boldsymbol{x} \in S_{t-1}} \left\{\boldsymbol{x}' \in D \;\middle|\; \ell_t(\boldsymbol{x}) - Ld(\boldsymbol{x}, \boldsymbol{x}') \geq h\right\}$
  $G_t \leftarrow \left\{\boldsymbol{x} \in S_t \;\middle|\; g_t(\boldsymbol{x}) > 0\right\}$
  $M_t \leftarrow \left\{\boldsymbol{x} \in S_t \;\middle|\; u_t(\boldsymbol{x}) \geq \max_{\boldsymbol{x}' \in S_t} \ell_t(\boldsymbol{x}')\right\}$
  $\boldsymbol{x}_t \leftarrow \operatorname{argmax}_{\boldsymbol{x} \in G_t \cup M_t} (w_t(\boldsymbol{x}))$
  $y_t \leftarrow f(\boldsymbol{x}_t) + n_t$
  Compute $Q_t(\boldsymbol{x})$, for all $\boldsymbol{x} \in S_t$
end for

outside of S_t, while it balances two objectives within that set: the desire to expand the safe region, and the need to localize high-reward regions within S_t. For the former, it maintains a set $G_t \subseteq S_t$ of candidate decisions that, upon potentially repeated selection, have a chance to expand S_t. For the latter, it maintains a set $M_t \subseteq S_t$ of decisions that are potential maximizers of f. To make progress, in each round it greedily picks the most uncertain decision $\boldsymbol{x}$, that is, the one with largest predictive variance among $G_t \cup M_t$. We present pseudocode of SAFEOPT in Algorithm 1, and next explain its workings in more detail.

Confidence-based classification. The classification of the domain into sets M_t, G_t, and S_t is done according to the GP posterior. In particular, at each iteration t, SAFEOPT uses the predictive confidence intervals

$$Q_t(\boldsymbol{x}) := \left[\mu_{t-1}(\boldsymbol{x}) \pm \beta_t^{1/2} \sigma_{t-1}(\boldsymbol{x})\right]. \tag{3}$$

We discuss the choice of β_t in the next section. Based on the assumptions about f, the sampled reward value at $\boldsymbol{x}$ lies within $Q_t(\boldsymbol{x})$ with high probability for all t. For technical reasons, instead of using Q_t directly, we use their intersection $C_t(\boldsymbol{x}) := C_{t-1}(\boldsymbol{x}) \cap Q_t(\boldsymbol{x})$, which ensures that confidence intervals are monotonically contained in each other. Based on this, we define $u_t(\boldsymbol{x}) := \max C_t(\boldsymbol{x})$ as an upper confidence bound on $f(\boldsymbol{x})$, monotonically decreasing in t, and similarly, $\ell_t(\boldsymbol{x}) := \min C_t(\boldsymbol{x})$ as a lower confidence bound, monotonically increasing in t. We also define the width $w_t(\boldsymbol{x}) := u_t(\boldsymbol{x}) - \ell_t(\boldsymbol{x})$ of the confidence interval, which is monotonically decreasing in t, and captures the uncertainty of the GP model about decision $\boldsymbol{x}$.

Having introduced the above notation, we define the essential sets considered by our algorithm, S_t, M_t, and G_t. The decisions that are certified to be safe are given by the set

$$S_t = \bigcup_{\boldsymbol{x} \in S_{t-1}} \left\{\boldsymbol{x}' \in D \;\middle|\; \ell_t(\boldsymbol{x}) - Ld(\boldsymbol{x}, \boldsymbol{x}') \geq h\right\}.$$

The potential maximizers are those decisions, for which the upper confidence bound is higher than the largest lower confidence bound, that is,

$$M_t = \left\{\boldsymbol{x} \in S_t \;\middle|\; u_t(\boldsymbol{x}) \geq \max_{\boldsymbol{x}' \in S_t} \ell_t(\boldsymbol{x}')\right\}.$$

In order to identify the set G_t, we first define the function

$$g_t(\boldsymbol{x}) := \left|\left\{\boldsymbol{x}' \in D \setminus S_t \;\middle|\; u_t(\boldsymbol{x}) - Ld(\boldsymbol{x}, \boldsymbol{x}') \geq h\right\}\right|,$$

which (optimistically) quantifies the potential enlargement of the current safe set after sampling a new decision $\boldsymbol{x}$. Then, G_t contains all decisions that could potentially expand the safe set, that is,

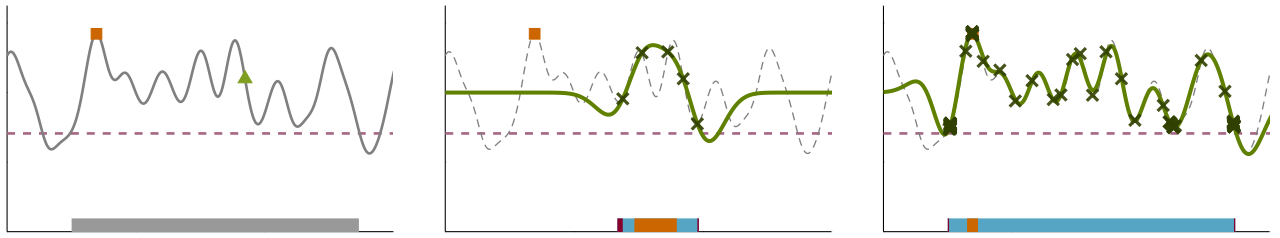
$$G_t = \left\{\boldsymbol{x} \in S_t \;\middle|\; g_t(\boldsymbol{x}) > 0\right\}.$$

Sampling criterion. Given the classification of points presented above, the selection rule of SAFEOPT is straightforward: it greedily selects the most uncertain decision among the potential maximizers (M_t), or expanders (G_t), that is, it selects decision $\boldsymbol{x}_t$ defined as

$$\boldsymbol{x}_t \in \operatorname*{argmax}_{\boldsymbol{x} \in M_t \cup G_t} w_t(\boldsymbol{x}).$$

Reducing the uncertainty within G_t eventually leads to expansion, that is, the discovery of new safe decisions. In turn, sampling within M_t reduces the uncertainty about the location of f's maximizers within S_t. The above greedy selection rule balances these two goals. An illustration of the sampling process is shown in Figure 1. SAFEOPT starts with a singleton seed set S_0. After 10 iterations, the safe set S_t has grown, while SAFEOPT picks new points from $G_t \cup M_t$. After 100 iterations, M_t has localized the near-optimal decisions, while S_t is close to the maximal reachable set $\bar{R}_0(S_0)$.

Discussion. The sets S_t, G_t, and M_t exhibit some interesting dynamics. As mentioned above, the safe set S_t is monotonically increasing, $S_0 \subseteq S_1 \subseteq S_2 \ldots$, and the algorithm consists of stages, within each of which, the set S_t does not change (possibly for several iterations). S_t expands only when enough evidence has been accrued to establish new decisions as safe. Within each such stage, G_t and M_t keep shrinking, due to the monotonicity of the confidence bounds used. As soon as new decisions are identified as safe, G_t and M_t may increase again. Furthermore, note that, even though we defined our optimization goal (1) with respect to an accuracy parameter ϵ, this parameter is



(a) True function and seed set (b) $t = 5$ (c) $t = 100$

Figure 1. Illustration of SAFEOPT. (a) The solid curve is the (unknown) function to optimize, the straight dashed line represents the threshold, and the triangle is the (singleton) seed set S_0. The gray bar shows the reachable set $\bar{R}_0(S_0)$, and the orange square is f_0^*. (b, c) The solid line is the estimated GP mean function after a number of observations shown as crosses. G_t is shown in purple, M_t in orange, and the rest of S_t in cyan.

not used by the algorithm, though it can be employed as a stopping condition. Namely, if the algorithm stops under the following condition

$$\max_{\boldsymbol{x} \in M_t \cup G_t} w_t(\boldsymbol{x}) \leq \epsilon,$$

then we will show below that w.h.p. the decision defined as $\hat{\boldsymbol{x}} := \operatorname{argmax}_{\boldsymbol{x} \in S_t} \ell_t(\boldsymbol{x})$ satisfies $f(\hat{\boldsymbol{x}}) \geq f_\epsilon^* - \varepsilon$.

4. Theoretical Results

The correctness of SAFEOPT crucially relies on the fact that the classification into sets S_t, M_t, and G_t is accurate. While this requires that the confidence bounds C_t be conservative, using bounds that are too conservative tends to slow down the algorithm considerably. Tightness of the confidence bounds is controlled by parameter β_t in equation (3), the choice of which has been studied by Srinivas et al. (2010). While their problem setting is different, the choice of β_t is still valid in our setting. In particular, for our theoretical results to hold it suffices to choose

$$\beta_t = 2B + 300\gamma_t \log^3(t/\delta), \tag{4}$$

where B is a bound on the RKHS norm of f, δ is the allowed failure probability, and γ_t quantifies the effective degrees of freedom associated with the kernel function. Concretely,

$$\gamma_t = \max_{|A| \leq t} I(f; \boldsymbol{y}_A)$$

is the maximal mutual information that can be obtained about the GP prior from t samples. For finite $|D|$, this quantity is always bounded by

$$\gamma_t \leq |D| \log\Big(1 + \sigma^{-2} t |D| \max_{\boldsymbol{x} \in D} k(\boldsymbol{x}, \boldsymbol{x})\Big),$$

i.e., $O(|D| \log t|D|)$, but for commonly used kernels (such as the Gaussian kernel), γ_t has sublinear dependence on $|D|$ (Srinivas et al., 2010). The following lemma, which immediately follows from Theorem 6 of Srinivas et al. (2010), justifies the above choice of β_t.

Lemma 1. *Suppose that $\|f\|_k^2 \leq B$, and that the noise n_t is zero-mean conditioned on the history, as well as uniformly bounded by σ_0 for all t. If β_t is chosen as in (4), then, for all $t \geq 1$, and all $\boldsymbol{x} \in D$, it holds with probability at least $1 - \delta$ that $f(\boldsymbol{x}) \in C_t(\boldsymbol{x})$.*

Based on this choice of β_t, we now present our main theorem, which establishes that SAFEOPT indeed manages to identify an ϵ-optimal decision, while staying safe throughout.

Theorem 1. *Assume that f is L-Lipschitz continuous, and satisfies $\|f\|_k^2 \leq B$, and the noise n_t is as in Lemma 1. Also, assume that $S_0 \neq \varnothing$, and $f(\boldsymbol{x}) \geq h$, for all $\boldsymbol{x} \in S_0$. Choose β_t as in Lemma 1, define $\hat{\boldsymbol{x}}_t := \operatorname{argmax}_{\boldsymbol{x} \in S_t} \ell_t(\boldsymbol{x})$, and let t^* be the smallest positive integer satisfying*

$$\frac{t^*}{\beta_{t^*} \gamma_{t^*}} \geq \frac{C_1 \left(|\bar{R}_0(S_0)| + 1\right)}{\epsilon^2},$$

where $C_1 = 8/\log(1 + \sigma^{-2})$. For any $\epsilon > 0$, and $\delta \in (0, 1)$, when running SAFEOPT the following jointly hold with probability at least $1 - \delta$:

- $\forall t \geq 1$, $f(\boldsymbol{x}_t) \geq h$,
- $\forall t \geq t^*$, $f(\hat{\boldsymbol{x}}_t) \geq f_\epsilon^* - \epsilon$.

The theorem states that, with high probability, SAFEOPT guarantees safety, and identifies at least one ϵ-optimal decision within the ϵ-reachable set after at most t^* iterations. The value of t^* depends on the size $|\bar{R}_0(S_0)|$, the accuracy parameter ϵ, the confidence parameter δ, the complexity of the function B, and the smoothness assumptions of the GP via γ_t. The proof is based on the following idea. Within a stage wherein S_t does not expand, the uncertainty $w_t(\boldsymbol{x}_t)$ monotonically decreases due to the construction of M_t and G_t. We prove that, the condition $\max_{\boldsymbol{x} \in G_t} w(\boldsymbol{x}) < \epsilon$ implies either of two possibilities: S_t will expand after the next evaluation, i.e., the reachable region will increase,

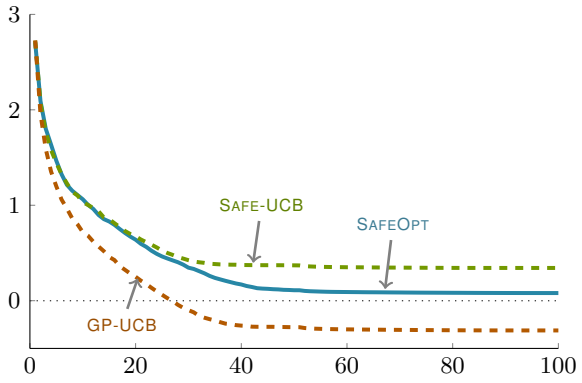


Figure 2. Synthetic data. Regret of the three algorithms.

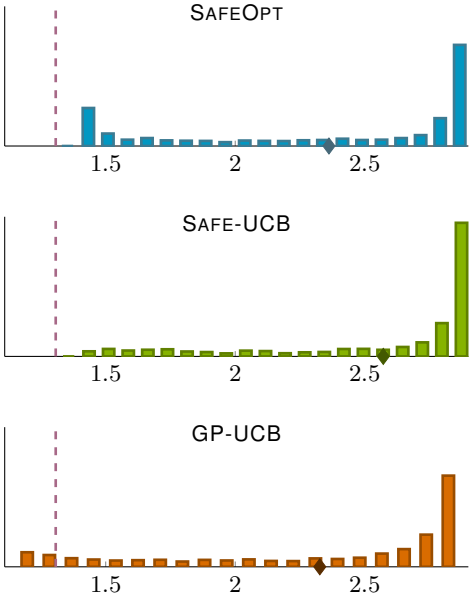


Figure 3. Synthetic data. Histograms of sampled function values after 100 iterations. The dashed lines represent the threshold and diamonds indicate the mean of the sampled function values.

and, hence, the next stage shall commence; or, we have already established all decisions within $\bar{R}_\epsilon(S_0)$ as safe, i.e., $S_t \supseteq \bar{R}_\epsilon(S_0)$. Similarly, we prove that the condition $\max_{\boldsymbol{x}\in M_t} w(\boldsymbol{x}) < \epsilon$ implies that we have identified an ϵ-optimal decision within the current safe set S_t. Finally, to establish the sample complexity we use a bound on how quickly $w_t(\boldsymbol{x}_t)$ decreases. We present the detailed proof of Theorem 1 in the longer version of this paper.

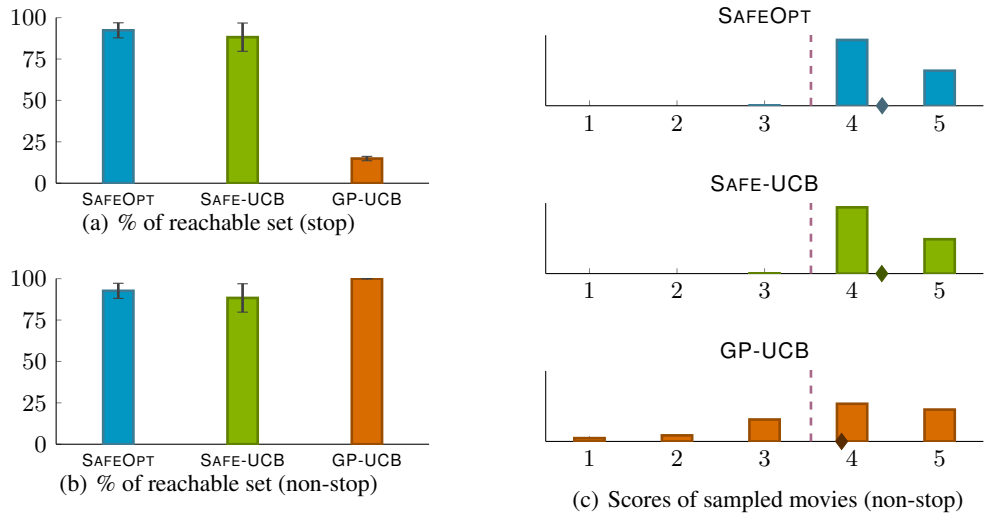
Guaranteeing safety via GP confidence bounds. In line 7 of Algorithm 1 we update S_t in a manner that allows us to guarantee safety in Theorem 1 based on the Lipschitz continuity assumption about f. It is natural to ask whether it is possible to also use the GP confidence intervals themselves to guarantee safety. As it turns out, we can easily modify the algorithm to additionally certify a decision as safe when its lower confidence bound lies above h. This merely requires changing the condition in the update rule of line 7 to be $\max\{\ell_t(\boldsymbol{x}) - Ld(\boldsymbol{x}, \boldsymbol{x}'), \ell_t(\boldsymbol{x}')\} \geq h$. We can prove a variant of Theorem 1, with the constant $|\bar{R}_0(S_0)| + 1$ replaced by $|D|$ (i.e., a worse bound). However, this modified version of the algorithm, which is more aggressive towards expanding, may sometimes perform better for some applications. Furthermore, it makes SAFEOPT less sensitive to the choice of L. In fact, it is possible in practice to solely use the confidence intervals to certify safety (by setting $L = \infty$).

5. Experiments

We evaluate our algorithm on synthetic data, as well as two real applications. In our experiments, we seek to address the following questions: Does SAFEOPT reliably respect the safety requirement? How effective is it at localizing good solutions quickly? How does it compare against standard (non-safe) bandit algorithms? In particular, we compare SAFEOPT against GP-UCB (Srinivas et al., 2010), a multi-armed bandit algorithm designed for Gaussian processes, which does not respect the safety constraint (see (2)). We also compare against SAFE-UCB, a heuristic variant of GP-UCB, which selects at every step the decision that maximizes the upper confidence bound (similar to GP-UCB), but only among decisions that are certified as safe in the same way as in SAFEOPT:

$$\boldsymbol{x}_t = \operatorname*{argmax}_{\boldsymbol{x}\in S_t} \left(\mu_{t-1}(\boldsymbol{x}) + \beta_t^{1/2}\sigma_{t-1}(\boldsymbol{x})\right).$$

Synthetic data. We first evaluate the algorithm on synthetic data. The purpose of this experiment is to validate our theory, and demonstrate the convergence of SAFEOPT in situations that perfectly match our prior assumptions. In particular, we sampled a set of 100 random functions from a zero-mean GP with squared exponential kernel over the space $D = [0, 1] \times [0, 1]$, uniformly discretized into 50×50 points. For each random function, we repeated the experiment 100 different random safe seeds (random points with value above the threshold). We estimated the Lipschitz constant from the gradient of several random functions sampled from the GP. Regarding each safe point as a separate seed set, we ran both SAFE-UCB and SAFEOPT for $T = 100$ iterations. Here, we report a notion of *regret*, defined as $r_t = f_0^* - \max_{1\leq i\leq t} f(\boldsymbol{x}_i)$. The regret values achieved by each algorithm are averaged over the 100 seeds for each of the 100 random functions. As we can see from Figure 2, SAFEOPT achieves smaller regret than SAFE-UCB on average. This is because for some cases SAFE-UCB fails to expand some low-rewarding boundary points,



(a) % of reachable set (stop)

(b) % of reachable set (non-stop)

(c) Scores of sampled movies (non-stop)

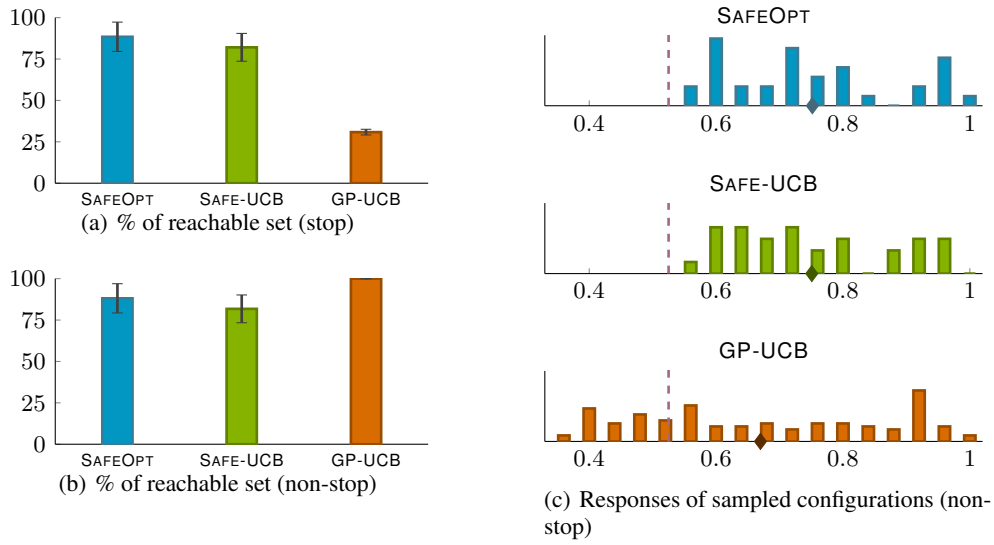
Figure 4. Movie recommendation. (a) Fraction of the safely reachable set $\bar{R}_0(S_0)$ that the algorithms explore within 300 iterations before violating the safety threshold. (b) Similar to (a) except the algorithms do not stop after violating the threshold. (c) The distribution of sampled movie ratings. The dashed lines represent the threshold and diamonds are the mean values of the sampled ratings.

which lie slightly above the safe threshold and, therefore, gets stuck at a local optimum. In contrast, SAFEOPT balances the localization of the optimal value within the current safe set with the expansion of the reachable region. Since GP-UCB is searching for the global optimum without any safety constraints, it achieves negative regret under the above definition. Figure 3 presents the histograms of the sampled function values obtained by the three algorithms after $T = 100$ iterations. As can be seen, SAFEOPT and SAFE-UCB are very unlikely to sample values below the threshold, while, as expected, a number of the GP-UCB samples are unsafe.

Safe movie recommendations. Next we consider an application in recommender systems, namely how to recommend movies, while ensuring that our suggestions are to the likes of the user under question (or at least are not particularly disliked). We test the algorithms on the MovieLens-100k dataset, which contains (sparse) ratings of 1682 movies from 943 customers. The main difference between our objective and commonly used objectives such as cumulative reward is that we are not only looking for high scoring movies, but also focus on avoiding low scoring ones. To put the problem into our framework, we proceed as follows. We first partition the data by selecting a subset of users for training. On the training data, we apply a matrix factorization with $k = 20$ latent factors, which provides a feature vector $\mathbf{v}_i \in \mathbb{R}^k$ for each movie i, and a feature vector $\mathbf{u}_j \in \mathbb{R}^k$ for each user in the training set. We then fit a Gaussian distribution $P(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mu, \Sigma)$ to the training user features. For a new user in the test set, we consider $P(\mathbf{u})$ as a prior, and use the Gaussian likelihood for their ratings of movie $\mathbf{v}_i$ as $P(y_i \mid \mathbf{u}, \mathbf{v}_i) = \mathcal{N}(\mathbf{v}_i^T \mathbf{u}, \sigma^2)$, where σ^2 is the residual variance on the training data. Thus, the ratings $(y_i)_i$ form a Gaussian process with linear kernel and Gaussian likelihood.

The safety threshold is set equal to the mean of all ratings. Given that the dataset only contains partial ratings, we restrict the algorithms to only be able to sample from the movies that the user has actually rated. After each selection, the actual rating from the data set is provided as feedback to the algorithms and we run each of them for $T = 300$ iterations. Since the ratings are discrete, taking values between 1 and 5, we observed that, for all algorithms, the regret reaches zero quickly within a fairly small number of iterations. Figure 4(a) shows the percentage of movies explored with respect to the maximal reachable set $\bar{R}_0(S_0)$, under the constraint that the algorithms stop after the first unsafe selection. GP-UCB violates the safety constraint considerably faster than the other two algorithms.

As a particular example, we present here the recommendations for a particular user in our data set, starting with a singleton seed set that consists of the movie "Return of the Jedi", which the user rated with a 5. During the first four iterations, SAFEOPT recommends {→ *The Empire Strikes Back* → *Stargate* → *Star Wars* → *Heavy Metal* }, while SAFE-UCB recommends {→ *The Empire Strikes Back* → *Star Wars* → *Star Trek* → *Raiders of the Lost Ark*}; all these movies score above the threshold. On the other hand, GP-UCB recommends {→ *Star Wars* → *Men in Black* → *A Close Shave* → *So I Married an Axe Murderer*}. The last movie recommended by GP-UCB returns a score below the threshold.



(a) % of reachable set (stop)

(b) % of reachable set (non-stop)

(c) Responses of sampled configurations (non-stop)

Figure 5. Spinal cord therapy application. (a) Fraction of the safely reachable set $\bar{R}_0(S_0)$ that the algorithms explore within 300 iterations before violating the safety threshold. (b) Similar to (a) except the algorithms do not stop after violating the threshold. (c) The distribution of sampled muscle activities. The dashed lines represent the threshold and diamonds indicate the mean values of the sampled muscle activities.

The movies recommended by SAFEOPT share some similarity with those of SAFE-UCB due to the locality encouraged by safe exploration. GP-UCB, on the other hand, recommends more diversely due to the lack of any safety restrictions while exploring. We can also see that SAFEOPT reaches a slightly larger set of movies than SAFE-UCB because it expands the current safe region more aggressively. Figure 4(b) shows the same plots as (a) with no stopping criteria applied, and Figure 4(c) shows histograms of the sampled ratings. GP-UCB practically always samples the whole reachable set, but its mean sampled rating is lower, and its variance is larger, a direct result of sampling a number of low-rated movies while exploring. In contrast, SAFEOPT and SAFE-UCB only very rarely happen to make an unsafe recommendation.

Safe exploration for spinal cord therapy. Our second application lies in the domain of spinal cord therapy (Harkema et al., 2011). Our dataset measures muscle activity triggered by therapeutic spinal electro-stimulation in rats that have suffered spinal cord injuries. The clinical goal is to choose stimulating configurations that maximize the resulting activity in lower limb muscles, as measured by electromyography (EMG), in order to improve spinal reflex and locomotor function. Bad configurations have negative effects on the rehabilitation and are often painful, hence the configurations we choose must result in responses that lie above some threshold. Electrode configurations are points in $\mathbb{R}^4$ representing cathode and anode locations. We fitted a squared exponential ARD kernel using experimental data from 351 stimulations (126 distinct configurations).

Figures 5(a) and (b) show the percentages of reachable configurations by the algorithms depending on whether or not we stop after violating the safety constraint respectively. Figure 5(c) shows the distribution of sampled values by each algorithm. We can make similar observations to our previous application. GP-UCB violates safety rather quickly, while SAFEOPT and SAFE-UCB safely explore a large part of $\bar{R}_0(S_0)$. Again, the resulting mean sampled values of SAFEOPT and SAFE-UCB are very close to each other, and clearly superior to those of GP-UCB, which samples a number of unsafe configurations.

6. Conclusions

We introduced the novel problem of sequentially optimizing an unknown function under safety constraints, and posed it formally using the concept of reachability. For this problem, we proposed SAFEOPT, an efficient algorithm that balances the tradeoff between exploring, expanding, and optimizing. Theoretically, we proved a bound on its sample complexity to achieve an ϵ-optimal solution, while guaranteeing safety with high probability. Experimentally, we demonstrated that SAFEOPT indeed exhibits its safety and convergence properties. We believe our results provide an important step towards employing machine learning algorithms "live" in safety-critical applications.

Acknowledgments

This work was partially supported by the Christopher and Dana Reeve Foundation, the National Institutes of Health (NIH), Swiss National Science Foundation Grant 200020_159557, and ERC Starting Grant 307036.

References

Audibert, Jean-Yves, Bubeck, Sebastien, and Munos, Remi. Best arm identification in multi-armed bandits. In *Conference on Learning Theory (COLT)*, 2010.

Auer, Peter. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research (JMLR)*, 2002.

Brochu, Eric, Cora, Vlad M., and de Freitas, Nando. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *CoRR*, abs/1012.2599, 2010.

Bryan, Brent, Schneider, Jeff, Nichol, Robert, Miller, Christopher, Genovese, Christopher, and Wasserman, Larry. Active learning for identifying function threshold boundaries. In *Neural Information Processing Systems (NIPS)*, 2005.

Bubeck, Sébastien and Cesa-Bianchi, Nicolo. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 2012.

Bubeck, Sébastien, Munos, Rémi, Stoltz, Gilles, and Szepesvári, Csaba. Online optimization in X-armed bandits. In *Neural Information Processing Systems (NIPS)*, 2008.

Dani, Varsha, Hayes, Thomas P., and Kakade, Sham M. Stochastic linear optimization under bandit feedback. In *Conference on Learning Theory (COLT)*, 2008.

Garcia, Javier and Fernandez, Fernando. Safe exploration of state and action spaces in reinforcement learning. *Journal of Machine Learning Research (JMLR)*, 2012.

Gillula, Jeremy and Tomlin, Claire. Guaranteed safe online learning of a bounded system. In *International Conference on Intelligent Robots and Systems (IROS)*, 2011.

Gotovos, Alkis, Casati, Nathalie, Hitz, Gregory, and Krause, Andreas. Active learning for level set estimation. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2013.

Hans, Alexander, Schneegaß, Daniel, Schäfer, Anton, and Udluft, Steffen. Safe exploration for reinforcement learning. In *European Symposium on Artificial Neural Networks (ESANN)*, 2008.

Harkema, Susan, Gerasimenko, Yury, Hodes, Jonathan, Burdick, Joel, Angeli, Claudia, Chen, Yangsheng, Ferreira, Christie, Willhite, Andrea, Rejc, Enrico, Grossman, Robert G, et al. Effect of epidural stimulation of the lumbosacral spinal cord on voluntary movement, standing, and assisted stepping after motor complete paraplegia: a case study. *The Lancet*, 2011.

Kleinberg, Robert, Slivkins, Aleksandrs, and Upfal, Eli. Multi-armed bandits in metric spaces. In *Symposium on Theory of Computing (STOC)*, 2008.

Moldovan, Teodor and Abbeel, Pieter. Safe exploration in markov decision processes. In *International Conference on Machine Learning (ICML)*, 2012.

Rasmussen, Carl Edward and Williams, Christopher K. I. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Robbins, Herbert. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 1952.

Schölkopf, Bernhard and Smola, Alex J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, 2002.

Srinivas, Niranjan, Krause, Andreas, Kakade, Sham, and Seeger, Matthias. Gaussian process optimization in the bandit setting: No regret and experimental design. In *International Conference on Machine Learning (ICML)*, 2010.

A. Proofs

Note All following lemmas hold for any $\varnothing \subsetneq S_0 \subseteq D$, $h \in \mathbb{R}$, $\delta \in (0, 1)$, and $\epsilon > 0$.

Define

$$\hat{\boldsymbol{x}}_t := \operatorname*{argmax}_{\boldsymbol{x} \in S_t} \ell_t(\boldsymbol{x}) \left(= \operatorname*{argmax}_{\boldsymbol{x} \in M_t} \ell_t(\boldsymbol{x}) \right)$$

Lemma 2. *The following hold for any $t \geq 1$:*

(i) $\forall \boldsymbol{x} \in D, u_{t+1}(\boldsymbol{x}) \leq u_t(\boldsymbol{x})$,

(ii) $\forall \boldsymbol{x} \in D, \ell_{t+1}(\boldsymbol{x}) \geq \ell_t(\boldsymbol{x})$,

(iii) $\forall \boldsymbol{x} \in D, w_{t+1}(\boldsymbol{x}) \leq w_t(\boldsymbol{x})$,

(iv) $S_{t+1} \supseteq S_t \supseteq S_0$,

(v) $S \subseteq R \Rightarrow R_\epsilon(S) \subseteq R_\epsilon(R)$,

(vi) $S \subseteq R \Rightarrow \bar{R}_\epsilon(S) \subseteq \bar{R}_\epsilon(R)$.

Proof. (i), (ii), and (iii) follow directly from their definitions and the definition of $C_t(\boldsymbol{x})$.

(iv) Proof by induction. For the base case, let $\boldsymbol{x} \in S_0$. Then,

$$\ell_1(\boldsymbol{x}) - Ld(\boldsymbol{x}, \boldsymbol{x}) = \ell_1(\boldsymbol{x}) \geq \ell_0(\boldsymbol{x}) \geq h,$$

where the last inequality follows from the initialization in line 2 of Algorithm 1. But then, from the above equation and line 7 of Algorithm 1, it follows that $\boldsymbol{x} \in S_1$.

For the induction step, assume that for some $t \geq 2$, $S_{t-1} \subseteq S_t$ and let $\boldsymbol{x} \in S_t$. By line 7 of Algorithm 1, this means that $\exists \boldsymbol{z} \in S_{t-1}, \ell_t(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) \geq h$. But, since $S_{t-1} \subseteq S_t$, it means that $\boldsymbol{z} \in S_t$. Furthermore, by part (ii), $\ell_{t+1}(\boldsymbol{z}) \geq \ell_t(\boldsymbol{z})$. Therefore, we conclude that $\ell_{t+1}(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) \geq h$, which implies that $\boldsymbol{x} \in S_{t+1}$.

(v) Let $\boldsymbol{x} \in R_\epsilon(S)$. Then, by definition, $\exists \boldsymbol{z} \in S, f(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) \geq h$. But, since $S \subseteq R$, it means that $\boldsymbol{z} \in R$, and, therefore, $f(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) \geq h$ also implies that $\boldsymbol{x} \in R_\epsilon(R)$.

(vi) This follows directly by repeatedly applying the result of part (v).

□

Lemma 3. *Assume that $\|f\|_k^2 \leq B$ and $n_t \leq \sigma$, $\forall t \geq 1$. If $\beta_t = 2B + 300\gamma_t \log^3(t/\delta)$, then the following holds with probability at least $1 - \delta$:*

$$\forall t \geq 1 \, \forall \boldsymbol{x} \in D, \ |f(\boldsymbol{x}) - \mu_{t-1}(\boldsymbol{x})| \leq \beta_t^{1/2} \sigma_{t-1}(\boldsymbol{x}).$$

Proof. See Theorem 6 by Srinivas et al. (2010). □

Corollary 1. *For β_t as above, the following holds with probability at least $1 - \delta$:*

$$\forall t \geq 1 \, \forall \boldsymbol{x} \in D, \ f(\boldsymbol{x}) \in C_t(\boldsymbol{x}).$$

Note Where needed in the following lemmas, we implicitly assume that the assumptions of Lemma 3 hold, and that β_t is defined as above.

Lemma 4. *For any $t_1 \geq t_0 \geq 1$, if $S_{t_1} = S_{t_0}$, then, for any t, such that $t_0 \leq t < t_1$, it holds that*

$$G_{t+1} \cup M_{t+1} \subseteq G_t \cup M_t.$$

Proof. Given the assumption that S_t does not change, both $G_{t+1} \subseteq G_t$ and $M_{t+1} \subseteq M_t$ follow directly from the definitions of G_t and M_t. In particular, for G_t, note that for any $\boldsymbol{x} \in S_t$, $g_t(\boldsymbol{x})$ is decreasing in t, since $u_t(\boldsymbol{x})$ is decreasing in t. For M_t, note that $\max_{\boldsymbol{x}' \in S_t} \ell_t(\boldsymbol{x}')$ is increasing in t, while $u_t(\boldsymbol{x})$ is decreasing in t (see Lemma 2 (i), (ii)). □

Lemma 5. *For any $t_1 \geq t_0 \geq 1$, if $S_{t_1} = S_{t_0}$ and $C_1 := 8/\log(1 + \sigma^{-2})$, then, for any t, such that $t_0 \leq t \leq t_1$, it holds that*

$$w_t(\boldsymbol{x}_t) \leq \sqrt{\frac{C_1 \beta_t \gamma_t}{t - t_0}}.$$

Proof. Given Lemma 4, the definition of $\boldsymbol{x}_t := \operatorname{argmax}_{\boldsymbol{x} \in G_t \cup M_t}(w_t(\boldsymbol{x}))$, and the fact that, by definition, $w_t(\boldsymbol{x}_t) \leq 2\beta_t^{1/2} \sigma_{t-1}(\boldsymbol{x}_t)$, the proof is completely analogous to that of Lemma 5.3 by Srinivas et al. (2010). □

Corollary 2. *For any $t \geq 1$, if C_1 is defined as above, T_t is the smallest positive integer satisfying $\frac{T_t}{\beta_{t+T_t} \gamma_{t+T_t}} \geq \frac{C_1}{\epsilon^2}$, and $S_{t+T_t} = S_t$, then, for any $\boldsymbol{x} \in G_{t+T_t} \cup M_{t+T_t}$, it holds that*

$$w_{t+T_t}(\boldsymbol{x}) \leq \epsilon.$$

Note Where needed in the following lemmas, we assume that C_1 and T_t are defined as above.

Lemma 6. *For any $t \geq 1$, if $\bar{R}_\epsilon(S_0) \setminus S_t \neq \varnothing$, then $R_\epsilon(S_t) \setminus S_t \neq \varnothing$.*

Proof. Assume, to the contrary, that $R_\epsilon(S_t) \setminus S_t = \varnothing$. By definition, $R_\epsilon(S_t) \supseteq S_t$, therefore $R_\epsilon(S_t) = S_t$. Iteratively applying R_ϵ to both sides, we get in the limit $\bar{R}_\epsilon(S_t) = S_t$. But then, by Lemma 2 (iv) and (vi), we get

$$\bar{R}_\epsilon(S_0) \subseteq \bar{R}_\epsilon(S_t) = S_t, \tag{5}$$

which contradicts the lemma's assumption that $\bar{R}_\epsilon(S_0) \setminus S_t \neq \varnothing$. □

Lemma 7. *For any $t \geq 1$, if $\bar{R}_\epsilon(S_0) \setminus S_t \neq \varnothing$, then the following holds with probability at least $1-\delta$:*

$$S_{t+T_t} \supsetneq S_t.$$

Proof. By Lemma 6, we get that, $R_\epsilon(S_t) \setminus S_t \neq \varnothing$, For equivalently, by definition,

$$\exists \boldsymbol{x} \in R_\epsilon(S_t) \setminus S_t \, \exists \boldsymbol{z} \in S_t, f(\boldsymbol{z}) - \epsilon - Ld(\boldsymbol{z}, \boldsymbol{x}) \geq h. \tag{6}$$

Now, assume, to the contrary, that $S_{t+T_t} = S_t$ (see Lemma 2 (iv)), which implies that $\boldsymbol{x} \in D \setminus S_{t+T_t}$ and $\boldsymbol{z} \in S_{t+T_t}$. Then, we have

$$\begin{aligned} u_{t+T_t}(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) &\geq f(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) && \text{by Lemma 3} \\ &\geq f(\boldsymbol{z}) - \epsilon - Ld(\boldsymbol{z}, \boldsymbol{x}) \\ &\geq h. && \text{by (6)} \end{aligned}$$

Therefore, by definition, $g_{t+T_t}(\boldsymbol{z}) > 0$, which implies $\boldsymbol{z} \in G_{t+T_t}$.

Finally, since $S_{t+T_t} = S_t$ and $\boldsymbol{z} \in G_{t+T_t}$, we can use Corollary 2 as follows:

$$\begin{aligned} \ell_{t+T_t}(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) &\geq \ell_{t+T_t} - f(\boldsymbol{z}) + \epsilon + h && \text{by (6)} \\ &\geq -w_{t+T_t}(\boldsymbol{z}) + \epsilon + h \\ & && \text{by Lemma 3} \\ &\geq h. && \text{by Corollary 2} \end{aligned}$$

This means that by line 7 of Algorithm 1 we get $\boldsymbol{x} \in S_{t+T_t}$, which is a contradiction. □

Lemma 8. *For any $t \geq 1$, if $S_{t+T_t} = S_t$, then the following holds with probability at least $1-\delta$:*

$$f(\hat{\boldsymbol{x}}_{t+T_t}) \geq \max_{x \in \bar{R}_\epsilon(S_0)} f(x) - \epsilon.$$

Proof. Let $\boldsymbol{x}^* := \operatorname{argmax}_{\boldsymbol{x} \in S_{t+T_t}} f(\boldsymbol{x})$. Note that $\boldsymbol{x}^* \in M_{t+T_t}$, since

$$\begin{aligned} u_{t+T_t}(\boldsymbol{x}^*) &\geq f(\boldsymbol{x}^*) && \text{by Lemma 3} \\ &\geq f(\hat{\boldsymbol{x}}) && \text{by definition of } \boldsymbol{x}^* \\ &\geq \ell_{t+T_t}(\hat{\boldsymbol{x}}) && \text{by Lemma 3} \\ &\geq \max_{\boldsymbol{x} \in S_{t+T_t}} \ell_{t+T_t}(\boldsymbol{x}). && \text{by definition of } \hat{\boldsymbol{x}} \end{aligned}$$

We will first show that $f(\hat{\boldsymbol{x}}_{t+T_t}) \geq f(\boldsymbol{x}^*) - \epsilon$. Assume, to the contrary, that

$$f(\hat{\boldsymbol{x}}_{t+T_t}) < f(\boldsymbol{x}^*) - \epsilon. \tag{7}$$

Then, we have

$$\begin{aligned} \ell_{t+T_t}(\boldsymbol{x}^*) &\leq \ell_{t+T_t}(\hat{\boldsymbol{x}}) && \text{by definition of } \hat{\boldsymbol{x}} \\ &\leq f(\hat{\boldsymbol{x}}) && \text{by Lemma 3} \\ &< f(\boldsymbol{x}^*) - \epsilon && \text{by (7)} \\ &\leq u_{t+T_t}(\boldsymbol{x}^*) - \epsilon && \text{by Lemma 3} \\ &\leq \ell_{t+T_t}(\boldsymbol{x}^*), \\ & && \text{by Corollary 2 and } \boldsymbol{x}^* \in M_{t+T_t} \end{aligned}$$

which is a contradiction.

Finally, since $S_{t+T_t} = S_t$, Lemma 7 implies that $\bar{R}_\epsilon(S_0) \subseteq S_t = S_{t+T_t}$. Therefore,

$$\begin{aligned} \max_{x \in \bar{R}_\epsilon(S_0)} f(x) - \epsilon &\leq \max_{x \in S_{t+T_t}} f(x) - \epsilon \quad \bar{R}_\epsilon(S_0) \subseteq S_{t+T_t} \\ &= f(\boldsymbol{x}^*) - \epsilon && \text{by definition of } \boldsymbol{x}^* \\ &\leq f(\hat{\boldsymbol{x}}_{t+T_t}). && \text{proven above} \end{aligned}$$

□

Corollary 3. *For any $t \geq 1$, if $S_{t+T_t} = S_t$, then the following holds with probability at least $1-\delta$:*

$$\forall t' \geq 0, f(\hat{\boldsymbol{x}}_{t+T_t+t'}) \geq \max_{x \in \bar{R}_\epsilon(S_0)} f(x) - \epsilon.$$

Proof. This is a direct consequence of the proof of the preceding lemma, combined with the facts that both $S_{t+T_t+t'}$ and $\ell_{t+T_t+t'}(\hat{\boldsymbol{x}}_{t+T_t+t'})$ are increasing in t' (by Lemma 2 (iv) and (ii) respectively), which imply that $\max_{\boldsymbol{x} \in S_{t+T_t+t'}} \ell_{t+T_t+t'}(\boldsymbol{x})$ can only increase in t'. □

Lemma 9. *For any $t \geq 0$, the following holds with probability at least $1-\delta$:*

$$S_t \subseteq \bar{R}_0(S_0).$$

Proof. Proof by induction. For the base case, $t = 0$, we have by definition that $S_0 \subseteq \bar{R}_0(S_0)$.

For the induction step, assume that for some $t \geq 1$, $S_{t-1} \subseteq \bar{R}_0(S_0)$. Let $\boldsymbol{x} \in S_t$, which, by definition, means $\exists \boldsymbol{z} \in S_{t-1}$, such that

$$\begin{aligned} & \ell_t(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) \geq h \\ \Rightarrow \; & f(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) \geq h. && \text{by Lemma 3} \end{aligned}$$

Then, by definition of $\bar{R}_0$ and the fact that $\boldsymbol{z} \in \bar{R}_0(S_0)$, it follows that $\boldsymbol{x} \in \bar{R}_0(S_0)$. □

Lemma 10. *Let t^* be the smallest integer, such that $t^* \geq |\bar{R}_0(S_0)| T_{t^*}$. Then, there exists $t_0 \leq t^*$, such that $S_{t_0+T_{t_0}} = S_{t_0}$.*

Proof. Assume, to the contrary, that for any $t \leq t^*$, $S_t \subsetneq S_{t+T_t}$. (By Lemma 2 (iv), we know that $S_t \subseteq S_{t+T_t}$.) Since T_t is increasing in t, we have

$$S_0 \subsetneq S_{T_0} \subseteq S_{T_{t^*}} \subsetneq S_{T_{t^*}+T_{T_{t^*}}} \subseteq S_{2T_{t^*}} \subsetneq \cdots,$$

which implies that, for any $0 \leq k \leq |\bar{R}_0(S_0)|$, it holds that $|S_{kT_{t^*}}| > k$. In particular, for $k^* := |\bar{R}_0(S_0)|$, we get

$$|S_{k^*T}| > |\bar{R}_0(S_0)|$$

which contradicts $S_{k^*T} \subseteq \bar{R}_0(S_0)$ by Lemma 9. □

Corollary 4. *Let t^* be the smallest integer, such that $\frac{t^*}{\beta_{t^*}\gamma_{t^*}} \geq \frac{C_1|\bar{R}_0(S_0)|}{\epsilon^2}$. Then, there exists $t_0 \leq t^*$, such that $S_{t_0+T_{t_0}} = S_{t_0}$.*

Proof. This is a direct consequence of combining Lemma 10 and Corollary 2. □

Lemma 11. *If f is L-Lipschitz continuous, then, for any $t \geq 0$, the following holds with probability at least $1-\delta$:*

$$\forall \boldsymbol{x} \in S_t, f(\boldsymbol{x}) \geq h.$$

Proof. We will prove this by induction. For the base case $t = 0$, by definition, for any $\boldsymbol{x} \in S_0$, $f(\boldsymbol{x}) \geq h$.

For the induction step, assume that for some $t \geq 1$, for any $\boldsymbol{x} \in S_{t-1}$, $f(\boldsymbol{x}) \geq h$. Then, for any $\boldsymbol{x} \in S_t$, by definition, $\exists \boldsymbol{z} \in S_{t-1}$,

$$\begin{aligned} h &\leq \ell_t(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) \\ &\leq f(\boldsymbol{z}) - Ld(\boldsymbol{z}, \boldsymbol{x}) && \text{by Lemma 3} \\ &\leq f(\boldsymbol{x}). && \text{by } L\text{-Lipschitz-continuity} \end{aligned}$$

□

Proof of Theorem 1. The first part of the theorem is a direct consequence of Lemma 11. The second part follows from combining Corollary 3 and Corollary 4. □