A/B Testing and Best-arm Identification for Linear Bandits with Robustness to Non-stationarity

Zhihan Xiong* University of Washington

Romain Camilleri* University of Washington

Maryam Fazel University of Washington

Lalit Jain University of Washington

Kevin Jamieson University of Washington

{zhihanx,camilr,mfazel,lalitj,jamieson}@uw.edu

*denotes equal contribution

Abstract

We investigate the fixed-budget best-arm identification (BAI) problem for linear bandits in a potentially non-stationary environment. Given a finite arm set $\mathcal{X} \subset \mathbb{R}^d$, a fixed budget T, and an unpredictable sequence of parameters $\{\theta_t\}_{t=1}^T$, an algorithm will aim to correctly identify the best arm $x^* := \arg\max_{x \in \mathcal{X}} x^\top \sum_{t=1}^T \theta_t$ with probability as high as possible. Prior work has addressed the stationary setting where $\theta_t = \theta_1$ for all t and demonstrated that the error probability decreases as $\exp(-T/\rho^*)$ for a problem-dependent constant ρ^*. But in many real-world $A/B/n$ multivariate testing scenarios that motivate our work, the environment is non-stationary and an algorithm expecting a stationary setting can easily fail. For robust identification, it is well-known that if arms are chosen randomly and non-adaptively from a G-optimal design over $\mathcal{X}$ at each time then the error probability decreases as $\exp(-T\Delta_{(1)}^2/d)$, where $\Delta_{(1)} = \min_{x \neq x^*}(x^* - x)^\top \frac{1}{T}\sum_{t=1}^T \theta_t$. As there exist environments where $\Delta_{(1)}^2/d \ll 1/\rho^*$, we are motivated to propose a novel algorithm P1-RAGE that aims to obtain the best of both worlds: robustness to non-stationarity and fast rates of identification in benign settings. We characterize the error probability of P1-RAGE and demonstrate empirically that the algorithm indeed never performs worse than G-optimal design but compares favorably to the best algorithms in the stationary setting.

Keywords: fixed-budget best-arm identification, non-stationary linear bandits, A/B testing, robust algorithms.

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

1 INTRODUCTION

Data-driven decision-making and A/B testing enable businesses to evaluate strategies using real-time customer data, offering insights into customer tendencies. As the use of these methods has increased, these technologies are being utilized to determine problems with smaller effect sizes, while also targeting smaller audiences. These two competing trends of smaller effect sizes and smaller sample sizes make it increasingly challenging to obtain statistical significance and correct inference since the absolute number of observations is limited. Consequently, there is a rising trend in using *adaptive* sampling like multi-armed bandits to obtain the same statistical insights using fewer total observations.

However, using adaptive experimentation schemes can come with many pitfalls. Most algorithms that are effective in practice (e.g., Thompson Sampling) are developed with the assumption that the *environment is stationary* and that rewards from treatments are stochastic. However in practice this is far from the case. Non-stationarity can be introduced from a variety of sources such as user populations that change from

hour to hour, customer preferences which vary over the course of a year, changes in one part of a platform that lead to latency and higher bounceback, site-wide promotions and sales, interference from competitors, macroeconomic shifts, and many other disruptions. Many of these issues are often totally unobservable, and therefore cannot be controlled, modeled, or accounted for by an experimenter. Under such an environment, it is also possible for the underlying performance of treatments to wildly change, and as a result, the treatment that is best performing on any given day may change. This makes the concept of "the best-performing arm" poorly defined.

Instead, in time-varying settings, the goal of an experimenter is to identify the "counterfactual best treatment" at the end of the experimentation period. That is, the treatment that would have received the *highest total reward had received all the samples*. However, in the absence of being able to predict or model time-variation, predicting precisely how a treatment would behave at every time point, at which time at most one treatment can be evaluated, is impossible. Fortunately, randomization is a powerful tool to provide the next best thing: unbiased *estimates* of how a treatment would behave as if it had been used at every time in the past. These methods are well-understood in the causal-inference and online learning literature and are commonly known as inverse-propensity score (IPS) estimators. The idea is simple: consider a sequence of evaluations from n treatments at each time $\{x_t\}_{t=1}^T \subset \mathbb{R}^n$. Note that a procedure can only observe at most one treatment per time denoted as $I_t \in [n]$, which is drawn from a distribution p_t over the n treatments. Then $\widehat{X}_i = \frac{1}{T}\sum_{t=1}^T \frac{\mathbb{1}\{I_t=i\}}{p_{t,i}} x_{t,i}$ is an unbiased estimator of the cumulative gain $\frac{1}{T}\sum_{t=1}^T x_{t,i}$ by

$$\begin{aligned}\mathbb{E}\left[\frac{\mathbb{1}\{I_t=i\}}{p_{t,i}}x_{t,i}\right] &= \sum_{j=1}^n \mathbb{P}(I_t=j)\frac{\mathbb{1}\{j=i\}}{p_{t,i}}x_{t,i} \\ &= \sum_{j=1}^n p_{t,j}\frac{\mathbb{1}\{j=i\}}{p_{t,i}}x_{t,i} = x_{t,i},\end{aligned} \tag{1}$$

as long as $\min_{t,i} p_{t,i} > 0$. Of course, there is no free lunch, and the variance of $\widehat{X}_i$ behaves like $\frac{1}{T^2}\sum_{t=1}^T 1/p_{t,i}$. Intuitively, to maximize efficiency of the samples we do take for inference, we should try to minimize the probabilities on poor performing treatments and prioritize mass for the high performing treatments. However, if the treatment performances vary over time, it can be challenging to determine how one might do this optimally. Fortunately, Abbasi-Yadkori et al. [2018] proposes a novel solution to defining these probabilities in a dynamic way that achieves a "Best of Both Worlds" (BOBW) guarantee, which is an algorithm called P1 that manages to achieve near-optimal rates regardless of whether the environment is stochastic or arbitrarily non-stationary (adversarial). This seminal work is the gold standard for A/B testing in unpredictable non-stationary settings.

If the number of treatments is small (<10 in practice), BOBW provides a robust solution for practitioners. However, there are many situations that practitioners are interested in for which the number of treatments is very large and intractable for traditional A/B testing. For example, multivariate testing Hill et al. [2017] aims to identify not just a single best item, but a set or bundle of items, such as the best 6 pieces of content to highlight on a home screen. Given n possibilities, this results in $\binom{n}{6}$ total distinct treatments for the A / B test! Given this combinatorial explosion, practitioners have made structural parametric assumptions, such as the expected value of a set of items behaves like

$$\theta^{(0)} + \sum_{i=1}^n \theta_i^{(1)}\alpha_i + \sum_{i=2}^n\sum_{j<i}\theta_{i,j}^{(2)}\alpha_i\alpha_j,$$

where $\alpha \in \{0,1\}^n$ with $\sum_i \alpha_i = 6$ indicates whether an item was included in the set or not. Note that these sums can be succinctly written as $\langle x, \theta\rangle$ for $\theta = (\theta^{(0)}, \theta^{(1)}, \theta^{(2)})^\top \in \mathbb{R}^{1+n+\binom{n}{2}}$ and an appropriate $x \in \{0,1\}^{1+n+\binom{n}{2}}$. This can reduce the overall number of unknowns, and dimension, to just $O(n^2)$ versus $O(n^6)$. But now the vectors $x \in \mathcal{X}$, each associated with a particular bundle, are overlapping and can share information. A similar situation arises if we have features or covariates that describe each possible treatment. For example, a particular song comes with lots of metadata including artist, genre, beats per minute, etc. which can encode the useful properties about the song.

In these kinds of scenario—whether it be multivariate testing or items with feature descriptions—we would like to perform adaptive experimentation in the presence of time-variation. Recall that without covariates, we have solutions like P1 that are near-optimal for time-variation. And without time-variation, there are many methods that take covariates into account and are known to be near-optimal. This work aims to develop an algorithmic framework for handling covariates with time variation.

The remainder of the paper is organized as follows. We discuss the related work in Section 2 and presents detailed problem formulations in Section 3. In Section 4, we propose a simple algorithm for general non-stationary environments and then in Section 5, we propose a robust algorithm that can simultaneously tackle stationary and non-stationary environments. Ex-

periment results are presented in Section 6 and our conclusions in Section 7.

2 RELATED WORK

The problem of identifying the best arm in linear bandits is a well-established and extensively researched problem. [Soare et al., 2014, Karnin, 2016, Xu et al., 2018, Fiez et al., 2019, Katz-Samuels et al., 2020, Degenne et al., 2020, Jedra and Proutiere, 2020, Wagenmaker and Foster, 2023]. Notably, Katz-Samuels et al. [2020], Azizi et al. [2021], Yang and Tan [2021] focus on the fixed-budget setting and are closely related to our paper. One notable limitation of these algorithms is their reliance on (unrealistic) stationary settings, which leads to their critical failure when applied in non-stationary scenarios. This motivated increasing interest in studying models for non-stationarity in bandits problems and algorithms agnostic to non-stationary settings, which we review next.

Models for non-stationarity in bandits. A reasonable approach in bandit problems with distribution shifts is to provide tight models for unknown variations in the reward distribution. Most literature in this setting focuses on minimizing the dynamic regret, which compares the reward obtained against the reward of the best arm in each round t. Garivier and Moulines [2011] demonstrates that existing methods such as Auer et al. [2002] could achieve a dynamic regret of $\widetilde{O}(\sqrt{LT})$ when L, the number of distribution shifts, is known. Then, Auer et al. [2019] makes a significant advancement by introducing an adaptive approach with the same dynamic regret but without the knowledge of L. More recently, Chen et al. [2019], Wei and Luo [2021] establish analogous results in the contextual bandits settings. Measures of non-stationarity other than L are also considered. In particular, Chen et al. [2019] measures the non-stationarity by total variation and Suk and Kpotufe [2022] proposes the novel notion of severe shifts. Note importantly that while this extensive body of work focuses on building tight models of non-stationarity and developing regret minimization algorithms tuned to them, our work is agnostic to such models.

Agnostic non-stationary bandits (Best of both worlds). Bubeck and Slivkins [2012], Seldin and Slivkins [2014], Seldin and Lugosi [2017], Auer and Chiang [2016], Abbasi-Yadkori et al. [2018], Lee et al. [2021] focus on the "best of both worlds" (BOBW) problem: design a bandit algorithm that agnostically achieves optimal performance in both stationary and non-stationary scenarios, even without prior knowledge of the environment. While most BOBW work focus on regret minimization goals, Abbasi-Yadkori et al. [2018] focuses on BOBW for best-arm identification. In this work, as in Abbasi-Yadkori et al. [2018], we focus on the agnostic setting.

A/B testing. As discussed in the introduction, our work is closely related to non-stationary A/B testing. In settings with non-stationarity and adaptive sample allocations, non-stationarity can lead to Simpson's paradox if the sample means are used to estimate arm means Kohavi and Longbotham [2011]. It is common in large-scale industrial platforms to assume that means vary smoothly Wu et al. [2022], or that the differences between them are constant; i.e., all arms are subject to the same random exogeneous shock Optimizely [2023]. The recent work Qin and Russo [2022] models time-variation as arising from confounding due to a context distribution and aims to find the arm with the best reward on average under this context distribution. Their goal is similar to ours, but, unlike them, we do not assume a context distribution.

3 PRELIMINARIES

Notation. Let $[a:b] = \{a, a+1, \dots, b\}$ for $a, b \in \mathbb{N}$ with $b > a$ and $[a] = \{1, \dots, a\}$. For a vector $x \in \mathbb{R}^d$ and symmetric positive semi-definite (PSD) matrix $A \in \mathbb{S}_+^d$, we use $\|x\|_A = \sqrt{x^\top A x}$ to denote the Mahalanobis norm. For a finite set $\mathcal{X} \subset \mathbb{R}^d$ and distribution $\lambda \in \triangle_{\mathcal{X}}$ over $\mathcal{X}$, we use $A(\lambda) = \mathbb{E}_{x\sim\lambda}\left[xx^\top\right]$ to denote the covariance matrix under λ.

3.1 Linear Bandits Problem Formulation

General stationary/non-stationary environments. In this paper, we assume a standard stationary/non-stationary linear bandits model with fixed horizon T. In particular, let $\mathcal{X} \subset \mathbb{R}^d$ be a finite arm set with $|\mathcal{X}| = K$ such that $\text{span}(\mathcal{X}) = \mathbb{R}^d$. At each time $t = 1, \dots, T$, the learner will pick some arm $x_t \in \mathcal{X}$ and receive some noisy reward $r_t = x_t^\top \theta_t + \epsilon_t$, where $\epsilon_t \in [-1, 1]$ is some independent zero-mean noise. All parameters $\{\theta_t\}_{t=1}^T$ are chosen and fixed by the environment before the game starts.[1] The ultimate goal of the learner is to find the optimal arm $\text{argmax}_{x\in\mathcal{X}} x^\top \overline{\theta}_T$, where $\overline{\theta}_T = \frac{1}{T}\sum_{t=1}^T \theta_t$ is the average parameter. This protocol is summarized in Figure 1.

For simplicity, we further assume that $\forall t \in [T]$, $\forall x \in \mathcal{X}$, $x^\top \theta_t \in [-1, 1]$ and the optimal arm $\text{argmax}_{x\in\mathcal{X}} x^\top \overline{\theta}_T$ is unique. Meanwhile, similar to Abbasi-Yadkori et al. [2018], we use the subscript (k) to denote the index of k-th best arm in $\mathcal{X}$, which means to have $x_{(1)}^\top \overline{\theta}_T >$

[1] Theoretically, this non-stationary setting has no essential difference with the adversarial setting. We choose this non-stationary setting mainly to keep our presentation concise.

Input: time horizon, T; arm set, $\mathcal{X} \subset \mathbb{R}^d$
For $t = 1, \ldots, T$
The learner plays arm $x_t \in \mathcal{X}$
The learner receives reward $r_t = x_t^\top \theta_t + \epsilon_t$,
where ϵ_t is independent zero-mean noise
The learner recommends arm x_{J_T}

Figure 1: General protocol of fixed-budget best-arm identification (BAI) for linear bandits.

$x_{(2)}^\top \overline{\theta}_T \geq \cdots \geq x_{(K)}^\top \overline{\theta}_T$. For each arm $k \in [K]$, we define its gap Δ_k as

$$\Delta_k = \begin{cases} (x_{(1)} - x_k)^\top \overline{\theta}_T & \text{if } k \neq (1), \\ (x_{(1)} - x_{(2)})^\top \overline{\theta}_T & \text{if } k = (1). \end{cases}$$

That is,we have $\Delta_{(1)} = \Delta_{(2)} \leq \Delta_{(3)} \leq \cdots \leq \Delta_{(K)}$. As a slight abuse of notation, for unindexed arm $x \in \mathcal{X}$, we will use Δ_x to denote the gap of x. The performance of the learner is measured by its error probability $\mathbb{P}_{\overline{\theta}_T}(J_T \neq (1))$, where J_T is the index of the learner's recommendation and the probability measure is taken over the randomness inside the learner and the reward noise. Finally, we note that when the setting is stationary, we simply have $\theta_1 = \cdots = \theta_T = \theta^*$ and everything else is then defined accordingly.

Remark 1 (Comparison to the adversarial setting). The traditional oblivious adversarial setting can be viewed as a special case of our non-stationary setting, in which we simply pick $\epsilon_t = 0$ for all t [Abbasi-Yadkori et al., 2018].

3.2 BAI for Linear Bandits in Stationary Environments

In this section, we briefly review the well-studied best-arm identification problem for linear bandits in stationary settings. This problem's complexity, first proposed in Soare et al. [2014], is defined as

$$\rho^*(\theta) = H_{\mathsf{LB}}(\theta) = \inf_{\lambda \in \triangle_\mathcal{X}} \max_{x \neq x_{(1)}} \frac{\left\|x - x_{(1)}\right\|^2_{A(\lambda)^{-1}}}{\Delta_x^2}, \quad (2)$$

where the optimal arm index (1) and gaps Δ_k are defined based on the input parameter θ. As discussed in Soare et al. [2014], this complexity is approximately equal to the number of samples required (up to logarithmic terms) to find the best arm by running an oracle algorithm. Later in Fiez et al. [2019], this complexity is proved to be the optimal sample complexity that a BAI algorithm can possibly achieve in a fixed-confidence setting. Recently, Katz-Samuels et al. [2020] proposes algorithm Peace in fixed-budget setting that achieves error probability $\mathbb{P}_\theta(J_T \neq (1)) \leq \widetilde{O}\left(\exp\left(-\frac{T}{\rho^*(\theta)\log(d)}\right)\right)$.[2]

4 BAI FOR LINEAR BANDITS IN GENERAL NON-STATIONARY ENVIRONMENTS

In this section, we present a simple algorithm G-BAI for the general non-stationary environment and analyze its theoretical guarantee. The algorithm is based on the G-optimal design, which refers to the distribution $\lambda^* \in \triangle_\mathcal{X}$ such that

$$\lambda^* = \operatorname*{arginf}_{\lambda \in \triangle_\mathcal{X}} \max_{x \in \mathcal{X}} \|x\|^2_{A(\lambda)^{-1}}. \quad (3)$$

Intuitively, G-optimal design allows us to estimate unknown parameter θ_t uniformly well over all directions of the arms in $\mathcal{X}$ [Soare et al., 2014]. which is suitable for addressing non-stationarity since θ_t may change arbitrarily and each $x \in \mathcal{X}$ may become the optimal at time t. Meanwhile, to make sure the estimation of θ_t is unbiased in a non-stationary environment, we use an IPS estimator.

Therefore, briefly speaking, at each time t, G-BAI simply samples x_t based on G-optimal design and estimate θ_t through an IPS estimator, whose details are summarized in Algorithm 1.[3]

Algorithm 1 G-optimal Best-arm Identification (G-BAI)

Require: budget, $T \in \mathbb{N}$; arm set $\mathcal{X} \subset \mathbb{R}^d$
1: Compute G-optimal design λ^* based on Eq. (3)
2: **for** $t = 1, 2, \ldots, T$ **do**
3: Sample $x_t \sim \lambda^*$ and receive reward r_t
4: **end for**
5: Estimate $\widehat{\theta}_T \leftarrow \frac{1}{T}\sum_{t=1}^T \mathbb{E}_{x \sim \lambda^*}\left[xx^\top\right]^{-1} x_t r_t$
6: **return** $\operatorname{argmax}_{x \in \mathcal{X}} x^\top \widehat{\theta}_T$

By the famous Kiefer-Wolfowitz theorem, an important property of the G-optimal design is that $\max_{x \in \mathcal{X}} \|x\|^2_{A(\lambda^*)^{-1}} = d$ [Lattimore and Szepesvári, 2020]. With this property, the variance of estimator $\widehat{\theta}_t$ can be easily controlled. We can then bound the error probability of G-BAI by this fact and the result is summarized in the following theorem.

[2] Rigorously speaking, the error probability of Peace contains another complexity term called $\gamma^*(\theta)$, which is defined as the minimum of a Gaussian width term. However, as argued in Katz-Samuels et al. [2020], $\gamma^*(\theta)$ is roughly in a same order of $\rho^*(\theta)$.

[3] We can see $\widehat{\theta}_T$ exactly becomes the more commonly-seen IPS estimator examined in Eq. (1) if we apply it to the multi-armed bandits setting, in which we have $K = d$ arms and $\mathcal{X} = \{\mathbf{1}_1, \ldots, \mathbf{e}_d\}$.

Theorem 1 (Error probability of G-BAI). *Fix time horizon T, arm set $\mathcal{X} \subset \mathbb{R}^d$ with $|\mathcal{X}| = K$ and arbitrary unknown parameters $\{\theta_t\}_{t=1}^T$. If we run Algorithm 1 in this non-stationary environment and obtain x_{J_T}, then it holds that*

$$\mathbb{P}_{\bar{\theta}_T}\left(J_T \neq (1)\right) \leq K \exp\left(-\frac{T}{12 H_{\text{G-BAI}}\left(\bar{\theta}_T\right)}\right),$$

where $H_{\text{G-BAI}}\left(\bar{\theta}_T\right) = \dfrac{d}{\Delta_{(1)}^2}$.

The proof of Theorem 1 is deferred to Appendix B. Here, we briefly compare this result with the one in multi-armed bandits, which can be treated as a special case of linear bandits by taking $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_K\}$ to be the canonical vectors (standard basis) with $K = d$.

In particular, Abbasi-Yadkori et al. [2018] shows that in multi-armed bandits setting, a simple uniform sampling algorithm reaches complexity $H_{\text{UNIF}}\left(\bar{\theta}_T\right) = \frac{K}{\Delta_{(1)}^2}$ and it is optimal in non-stationary environments. Meanwhile, based on Theorem 1, we can see the complexity of G-BAI is $H_{\text{G-BAI}}\left(\bar{\theta}_T\right) = \frac{d}{\Delta_{(1)}^2}$, which is exactly $H_{\text{UNIF}}(\bar{\theta}_T)$ if we treat multi-armed bandits as a special case of linear bandits since $d = K$. Furthermore, if we directly apply G-BAI to multi-armed bandits, meaning to use $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_K\}$, then λ^* is exactly the uniform distribution over $\mathcal{X}$. That is, in multi-armed bandits, G-BAI exactly recovers the optimal complexity in non-stationary environments, which shows that G-BAI is minimax optimal for linear bandits.

5 A ROBUST ALGORITHM FOR STATIONARY/NON-STATIONARY ENVIRONMENTS

In this section, we present and analyze a new robust linear bandits BAI algorithm called P1-RAGE, which performs comparable to G-BAI in non-stationary environments but much better than it in stationary environments. We will show that it attains good error probability in both stationary and non-stationary environments simultaneously, without knowing a priori which environment it will encounter. We first discuss some intuitions behind the algorithm design.

Stationary environments. The development of our algorithm P1-RAGE is largely inspired by the high-level idea of the robust algorithm P1, proposed in Abbasi-Yadkori et al. [2018], and the allocation strategy of RAGE, proposed in Fiez et al. [2019]. In particular, as discussed in Abbasi-Yadkori et al. [2018], in multi-armed bandits, to minimize the error probability in stationary environment, we need to control the estimation variance of the optimal arm well enough. Therefore, at each time step, algorithm P1 pulls the current estimated best arm with the highest probability (unnormalized "probability one"), then subsequently the second best arm with second highest probability (unnormalized "probability half") and so on. We can notice that it actually matches the allocation strategy of the successive halving algorithm in Karnin et al. [2013], which is proved to be near-optimal for BAI in stationary multi-armed bandits. Therefore, we design our probability allocation based on the allocation strategy of RAGE, which is proven to be near-optimal for fixed-confidence BAI in stationary linear bandits [Fiez et al., 2019]. In particular, with the estimated parameter $\widehat{\theta}_t$, we first find the estimated best arms $\hat{x}_t^* = \operatorname{argmax}_{x \in \mathcal{X}} x^\top \widehat{\theta}_t$. Then, we use a subroutine to repeatedly and virtually eliminate arms with estimated gaps larger than certain threshold and compute $\mathcal{XY}$-allocation of the (virtually) remaining arms.[4] Then, we average over the allocation probabilities computed during each iteration.

Non-stationary environments. Finally, to address the potential non-stationarity in environments, we uniformly mix the allocation probability computed above with a G-optimal design. With such a mixture, the variance over all arms can be controlled well and thus the algorithm will be robust for both stationary and non-stationary environments. The details of P1-RAGE are summarized in Algorithm 2 and the subroutine to compute the allocation probability, called RAGE-Elimination, is summarized in Algorithm 3.

Algorithm 2 P1-RAGE

Input: budget, $T \in \mathbb{N}$; arm set $\mathcal{X} \subset \mathbb{R}^d$; maximum number of virtual phases, m
Compute G-optimal design λ^* based on Eq. (3) and initialize $\lambda_1 = \lambda^*$
for $t = 1, 2, \ldots, T$ **do**
  Sample $x_t \sim \lambda_t$ and receive reward r_t
  Estimate $\widehat{\theta}_t \leftarrow \frac{1}{t}\sum_{s=1}^t \mathbb{E}_{x \sim \lambda_s}\left[xx^\top\right]^{-1} x_s r_s$
  Update $\lambda_{t+1} \leftarrow$RAGE-Elimination$(\widehat{\theta}_t, m)$ // {Call Algorithm 3}
end for
return $\operatorname{argmax}_{x \in \mathcal{X}} x^\top \widehat{\theta}_T$

We bound the error probability of P1-RAGE under both stationary and non-stationary settings in the following theorem and its proof is deferred to Appendix C.

Theorem 2 (Error Probability of P1-RAGE). *Fix arm set $\mathcal{X} \subset \mathbb{R}^d$ with $|\mathcal{X}| = K$ and budget T. For a stationary environment with unknown parameter θ, if $m \geq i_0 = \lceil \log_2\left(1/\Delta_{(1)}\right) \rceil + 1$, then there exists absolute constant $c > 0$ such that the error probability of*

[4] The elimination is virtual because no samples are collected during the elimination subroutine.

Algorithm 3 RAGE-Elimination

Input: arm set $\mathcal{X} \subset \mathbb{R}^d$; current estimate $\widehat{\theta}_t$; maximum number of virtual phases, m
Find $\hat{x}_t^* \leftarrow \operatorname{argmax}_{x \in \mathcal{X}} x^\top \widehat{\theta}_t$
Initialize $\mathcal{X}_t^{(0)} \leftarrow \mathcal{X}$ and $i \leftarrow 0$
while $|\mathcal{X}_t^{(i)}| > 1$ and $i \le m$ **do**
  $\lambda_t^{(i)} \leftarrow \operatorname{arginf}_{\lambda \in \triangle_{\mathcal{X}}} \max_{x,x' \in \mathcal{X}_t^{(i)}} \|x - x'\|^2_{A(\lambda)^{-1}}$
  $\mathcal{X}_t^{(i+1)} \leftarrow \left\{ x \in \mathcal{X}_t^{(i)} \,\middle|\, \widehat{\theta}_t^\top (\hat{x}_t^* - x) \le 2^{-i} \right\}$
  $i \leftarrow i + 1$
end while
return $(\bar{\lambda}_t + \lambda^*)/2$, where $\bar{\lambda}_t = \frac{1}{i} \sum_{i'=0}^{i-1} \lambda_t^{(i')}$

P1-RAGE satisfies

$$\mathbb{P}_\theta \left(J_T \neq (1) \right) \le 2 i_0 K T \exp\left(-\frac{cT}{H_{\textit{P1-RAGE}}(\theta)} \right),$$

$$H_{\textit{P1-RAGE}}(\theta) = \frac{m i_0}{\Delta_{(1)}} \inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{x \neq x_{(1)}} \frac{\left\| x - x_{(1)} \right\|^2_{A(\lambda)^{-1}}}{\Delta_x} + \frac{m\sqrt{d}}{\Delta_{(1)}} \inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{x \neq x_{(1)}} \left\| x - x_{(1)} \right\|_{A(\lambda)^{-1}}. \tag{4}$$

For a non-stationary environment with unknown parameter $\{\theta_t\}_{t=1}^T$, there exists absolute constant $c' > 0$ such that the error probability of P1-RAGE satisfies

$$\mathbb{P}_{\bar{\theta}_T} \left(J_T \neq (1) \right) \le K \exp\left(-\frac{c' T \Delta_{(1)}^2}{d} \right).$$

We can immediately see that in non-stationary environments, the error probability of P1-RAGE matches (up to a constant) with G-BAI, showing that P1-RAGE is minimax optimal for linear bandits under non-stationarity. On the other hand, because of the $\frac{1}{\Delta_{(1)}}$ factor, we can see that in stationary environments, $H_{\textsf{P1-RAGE}}(\theta) \gtrsim H_{\textsf{LB}}(\theta)$ (defined in Eq. (2)), which implies that P1-RAGE is suboptimal in stationary settings. However, this should be expected since even for multi-armed bandits, as proved in Abbasi-Yadkori et al. [2018], it is impossible for an algorithm to achieve $H_{\textsf{LB}}(\theta)$ while being robust to non-stationarity, let alone linear bandits.

Nevertheless, when applying Theorem 2 to multi-armed bandits ($\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_K\}$), as long as we choose $m \approx i_0$, we can show that (Corollary 1 in Appendix C)

$$H_{\textsf{P1-RAGE}}(\theta) = \widetilde{O}\left(\frac{1}{\Delta_{(1)}} \max_{k \in [K]} \frac{k}{\Delta_{(k)}} \right) = \widetilde{O}\left(H_{\text{BOB}}(\theta) \right),$$

where $H_{\text{BOB}}(\theta)$ is the best-of-both-worlds complexity proposed in Abbasi-Yadkori et al. [2018]. In particular, Abbasi-Yadkori et al. [2018] proves that $H_{\text{BOB}}(\theta)$ is the best complexity that any algorithm can possibly achieve if it is constrained to be robust to non-stationarity. That is, again, our algorithm P1-RAGE retains the near-optimal complexity for stationary multi-armed bandits if it is constrained to be robust in non-stationary environments.

Remark 2. Here, we do not elaborate the proof details of Theorem 2 mainly because we do not recognize them as widely applicable techniques. However, we do want to emphasize that this proof is by no means a simple extension of the analysis of the algorithm P1 in Abbasi-Yadkori et al. [2018]. In particular, our proof uses a different set of virtual events based on the estimated gaps. Meanwhile, the analysis of subroutine RAGE-Elimination is intricately tailored to the unique characteristics of being a virtual elimination strategy, which is not presented in neither RAGE nor P1 [Abbasi-Yadkori et al., 2018, Fiez et al., 2019].

Theoretical limitations of P1-RAGE. Despite being near-optimal in multi-armed bandits, $H_{\textsf{P1-RAGE}}(\theta)$ includes an extra low-order term $\frac{m\sqrt{d}}{\Delta_{(1)}} \inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{x \neq x_{(1)}} \left\| x - x_{(1)} \right\|_{A(\lambda)^{-1}}$. This term appears because the Bernstein's inequality requires a bound of the estimator's magnitude, which can be removed if the concentration bound only scales with the estimator's variance. Although this can often be accomplished by using Catoni's robust mean estimator [Wei et al., 2020], it requires a concrete confidence level to be specified before estimation, which is not feasible in our fixed budget setting. Finding an approach to circumvent this difficulty and remove this extra term, or alternatively, demonstrate that it is necessary, is an open question.

Remark 3. The question of whether the extra term is removable naturally relates the instance-dependent lower bound of this problem. However, proving an instance-dependent lower bound for our setting requires constructing both stationary and non-stationary counterexamples. This task is thereby more challenging compared to proving an instance-dependent lower bound for the fixed-budget best-arm identification problem in linear bandits within a purely stationary setting, an open question that persists (see Yang and Tan [2022] for a minimax lower bound). We thus leave establishing such instance-dependent lower bounds for future work.

Parameter choice of P1-RAGE. Although P1-RAGE requires a user-specified parameter $m \ge \lceil \log(1/\Delta_{(1)}) \rceil + 1$ to bound the total number of virtual phases, it is not difficult to choose a reasonable value for this parameter in a practical implementation. On the one hand, since its dependence on $\Delta_{(1)}^{-1}$ is only logarithmic, taking some moderate value such as $m = 25$ should safely satisfy $m \ge i_0$ for most practical scenarios; on the other hand,

in most real-world applications, a sub-optimal arm should always be acceptable as long as its gap is small enough. Indeed, if we take ϵ to be the largest acceptable sub-optimality gap and take $m \geq \lceil \log(1/\epsilon) \rceil + 1$, then P1-RAGE will output arm x_{J_T} that satisfies $\Delta_{J_T} \leq \max\{\epsilon, \Delta_{(1)}\}$ with high probability in pure stationary environments (Corollary 2 in Appendix C). That is, the output arm will either be an optimal arm if $\epsilon \leq \Delta_{(1)}$ or an arm with an acceptable suboptimality gap ϵ otherwise.

6 EXPERIMENTS

In this section, we present our experiment results on several stationary/non-stationary environments. Since to the best of our knowledge, we are the first to propose best-arm identification algorithms that tackle non-stationarity in linear bandits, the algorithms from other works that we compare with are all specifically designed for stationary environments. In particular, we will compare our algorithms with Peace, which is the first fixed-budget algorithm for linear bandits and also inspires our algorithmic design [Katz-Samuels et al., 2020], and OD-LinBAI, which is the most recent algorithm of this kind and is claimed to be minimax optimal [Yang and Tan, 2022].

Meanwhile, we also examine two additional heuristically designed algorithms for non-stationary environments. The first one is P1-Peace, which has the same design spirit as P1-RAGE but uses a different Peace-based virtual elimination subroutine; the second one is Mixed-Peace, which is a naive mixture of Peace and the G-optimal design. In particular, while P1-RAGE/P1-Peace combines G-optimal design with what RAGE/Peace would sample *in a full run*, Mixed-Peace simply mixes G-optimal design with what Peace in a stationary environment samples *at each time step*. The details of these two additional algorithms are summarized in Algorithm 4 and 6 in Appendix A.1, respectively. More implementation details and additional experiments can be found in Appendix D.[5]

Stationary benchmark example. First, as a sanity check, we consider the famous stationary benchmark example proposed in Soare et al. [2014]. In particular, we have $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_d, x'\}$, where $x' = \cos(\omega)\mathbf{e}_1 + \sin(\omega)\mathbf{e}_2$ with some small $\omega > 0$, and $\overline{\theta}_T = \theta^* = 2\mathbf{e}_1$ so that $x_{(1)} = \mathbf{e}_1$. An efficient algorithm should pick $\mathbf{e}_2$ frequently to reduce the variance in the direction of $\mathbf{e}_1 - x'$. In this example, we pick $d = 10$ and $\omega = 0.1$.

The results are shown in Figure 2. We can see that both our algorithms, P1-RAGE and P1-Peace, perform better than G-BAI and comparably with Peace, showing that our algorithms maintain good performance in stationary environments. Meanwhile, we also notice that Mixed-Peace has performance only comparable to G-BAI, showing that naively mixing the allocation strategy with the G-optimal design can downgrade the performance in stationary environments.

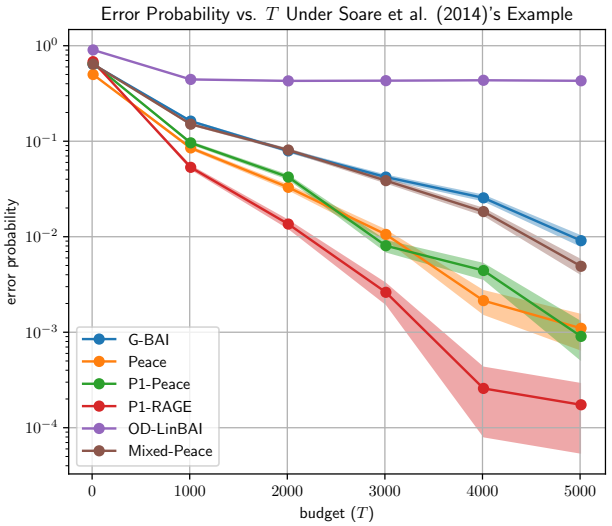

Figure 2: Each error probability is estimated through at least 2×10^4 independent trials. The vertical axis is on log scale and the shaded area represents the 95% confidence interval.

Non-stationary multivariate testing example. We consider a multivariate testing example from Fiez et al. [2019], which is also similar to the one discussed in Introduction. Considering a webpage with D distinct slots and suppose each slot has two content choices, where we represent each layout as an element $w \in \mathcal{W} = \{-1, 1\}^D$. We hope to maximize the click-through rate and we assume it linearly depends on a layout-determined arm $x \in \mathcal{X}$ in a form of

$$x^\top \theta^* = \theta_0^* + \alpha_1 \sum_{j=1}^{D} \theta_j^* w_j + \alpha_2 \sum_{k=1}^{D-1} \sum_{\ell=k+1}^{D} \theta_{k,\ell}^* w_k w_\ell.$$

Here θ_0^* is the common bias, θ_j^* is the weight of j-th slot and $\theta_{k,\ell}^*$ is the weight of the interaction between k-th and ℓ-th slots. Because of the periodic nature of people's life cycle, it is very likely that the real-world weights will periodically change. Therefore, to construct a non-stationary environment, we randomly oscillate the weights with scale s and period L to get

$$\theta_{t,i} = \theta_i^* + sI \left\|\theta^*\right\|_\infty \sin\left(\frac{2\pi t}{L} + \phi_i\right),$$
$$\text{where } I \sim \mathsf{Unif}(\{0,1\}), \phi_i \sim \mathsf{Unif}([0, 2\pi]).$$

[5] Code repository is available at https://github.com/FFTypeZero/bobw_linear.

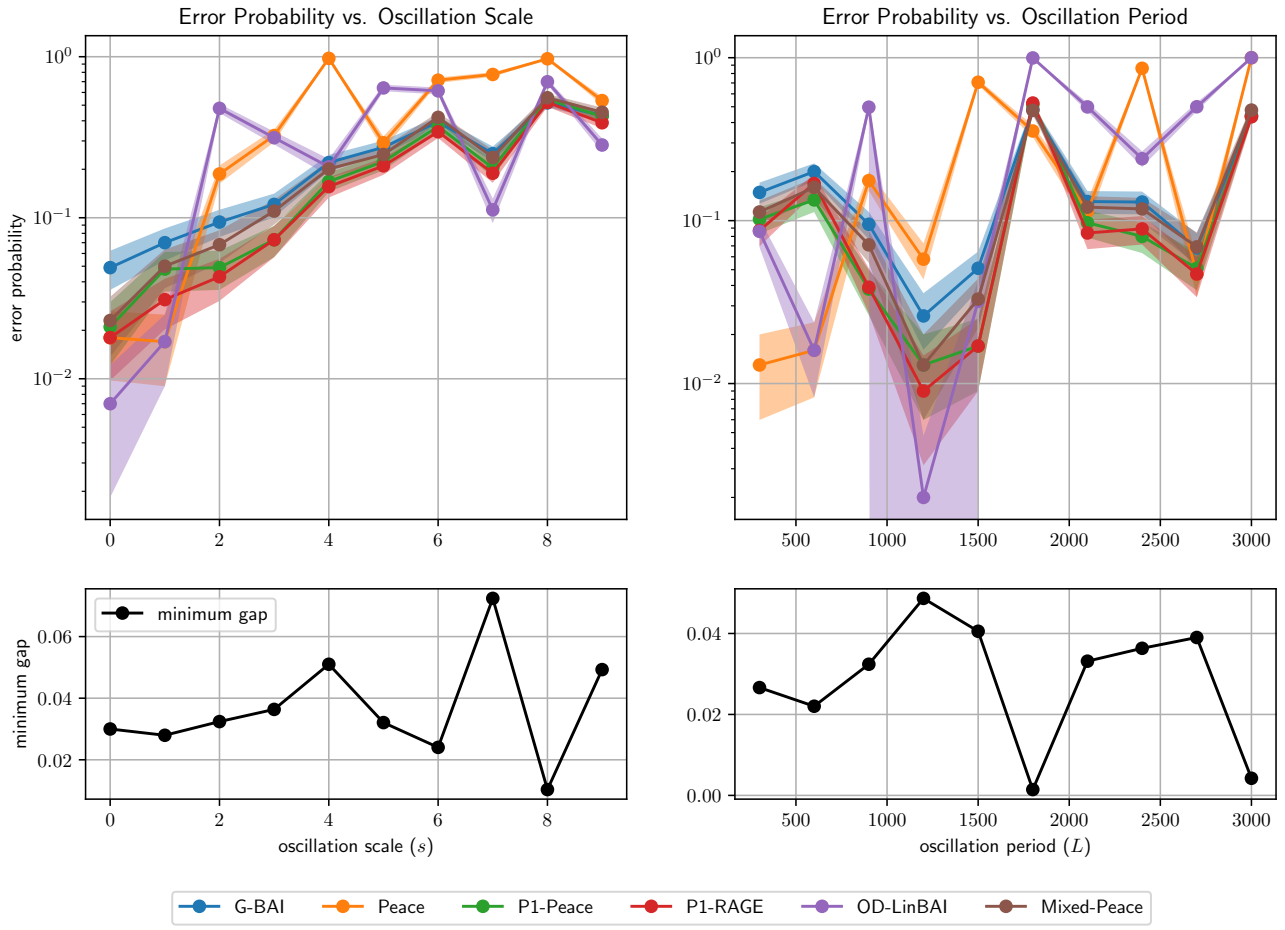

Figure 3: Each error probability is estimated through 1000 repeated trials. The bottom two plots give the minimum gap $\Delta_{(1)}$ of each instance as a function of oscillation scale s and oscillation period L.

Here, in the first series of instances, we fix $L = 900$ and take values $s \in \{0, 1, \ldots, 9\}$, and in the second series of instances, we fix $s = 2$ and take values $L \in \{300, 600, \ldots, 3000\}$. Finally, we take $\alpha_1 = 1$, $\alpha_2 = 0.5$, sample each component of θ^* uniformly in $[-0.1, 0.1]$ and guarantee that $\overline{\theta}_T$ has the same optimal arm as θ^*. We take $T = 10^4$ for all settings and the results are shown in Figure 3.

From the plots, we can see that the error probabilities of Peace and OD-LinBAI, algorithms designed for stationary environments, can range from near 0 to 1 in different non-stationary environments, which is quite unstable. Meanwhile, we can see that the performance of the other four algorithms, which all in certain way contain a G-optimal design, is relatively much more stable.[6] Furthermore, among these four algorithms, we can see that our algorithms P1-RAGE and P1-Peace consistently outperform (never worse than) G-BAI and Mixed-Peace.

Non-stationary click-through example. To create an instance using real-world data, we use the Yahoo! Webscope Dataset R6A [Yahoo!, 2011].[7] This dataset contains a fraction of user click log of Yahoo!'s news article from May 1st, 2009 to May 10th, 2009. For each click, we take the outer product between user and article features to get a vector in $\mathbb{R}^{36}$ and then we run a principle component analysis to get arm set $\mathcal{Z} \subset \mathbb{R}^{24}$. To create a non-stationary example, we take data from May 1st to May 7th and for each day's data, we fit a ridge regression with regularization 0.01, obtaining $\theta_1^*, \ldots, \theta_7^*$, which can be used to simulate user's weekly periodic behavior. Suppose we receive L visits each day, then, we can define a non-stationary environment where each period consists of $\theta_1^*, \ldots, \theta_1^*, \ldots, \theta_7^*, \ldots, \theta_7^*$ and each θ_i^* repeats for L times. Finally, we form our arm set $\mathcal{X}$ by picking the optimal arm from $\mathcal{Z}$ plus 23 randomly picked arms with gap at least 0.05

[6] All algorithms fluctuate in the upper right plot mainly because the minimum gaps also have large fluctuation.

[7] https://webscope.sandbox.yahoo.com/

so that $\text{span}(\mathcal{X}) = \mathbb{R}^{24}$. We take $T = 2.1 \times 10^4$ and the results are shown in Figure 4. Again, we can see that the performance of Peace and OD-LinBAI is very unstable and the performance of P1-RAGE and P1-Peace consistently outperforms the other two naive G-optimal-design-based algorithms, G-BAI and Mixed-Peace.

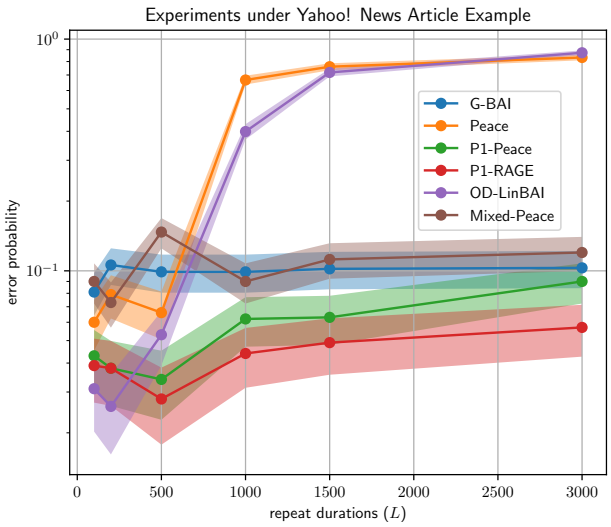

Figure 4: Each error probability is estimated through 1000 independent trials. The vertical axis is on log scale and the shaded area represents the 95% confidence interval.

7 CONCLUSIONS AND FUTURE WORK

To the best of our knowledge, in this paper, we present the first two novel robust linear bandits algorithm for fixed-budget best-arm identification, P1-RAGE and P1-Peace, that tackle stationary and non-stationary environments simultaneously while being agnostic to the environment. Theoretically, we prove error probability bounds of P1-RAGE in both stationary and non-stationary environments. Empirically, we show that in stationary settings, both P1-RAGE and P1-Peace perform comparably with algorithms designed for such environments, and in non-stationary settings, they consistently outperform naive algorithms based on G-optimal design.

Finally, several questions still remain open. Is the extra term in $H_{\text{P1-RAGE}}(\theta)$, as discussed in Section 5, necessary? What is the optimal complexity for this mixed stationary/non-stationary settings? Answering these questions can serve as promising future directions.

Acknowledgements

ZX sincerely thanks Anze Wang for his great favor offered during the time of paper writing. This work was supported in part by the NSF TRIPODS II grant DMS 2023166, NSF CCF 2007036, NSF CCF 2212261 and Microsoft Grant for Customer Experience Innovation.

References

Yasin Abbasi-Yadkori, Peter Bartlett, Victor Gabillon, Alan Malek, and Michal Valko. Best of both worlds: Stochastic & adversarial best-arm identification. In *Conference on Learning Theory*, pages 918–949. PMLR, 2018.

Jean-Yves Audibert, Sébastien Bubeck, and Rémi Munos. Best arm identification in multi-armed bandits. In *COLT*, pages 41–53, 2010.

Peter Auer and Chao-Kai Chiang. An algorithm with nearly optimal pseudo-regret for both stochastic and adversarial bandits. In *Conference on Learning Theory*, pages 116–120. PMLR, 2016.

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1): 48–77, 2002.

Peter Auer, Pratik Gajane, and Ronald Ortner. Adaptively tracking the best bandit arm with an unknown number of distribution changes. In *Conference on Learning Theory*, pages 138–158. PMLR, 2019.

Mohammad Javad Azizi, Branislav Kveton, and Mohammad Ghavamzadeh. Fixed-budget best-arm identification in structured bandits. *arXiv preprint arXiv:2106.04763*, 2021.

Sébastien Bubeck and Aleksandrs Slivkins. The best of both worlds: Stochastic and adversarial bandits. In *Conference on Learning Theory*, pages 42–1. JMLR Workshop and Conference Proceedings, 2012.

Yifang Chen, Chung-Wei Lee, Haipeng Luo, and Chen-Yu Wei. A new algorithm for non-stationary contextual bandits: Efficient, optimal and parameter-free. In *Conference on Learning Theory*, pages 696–726. PMLR, 2019.

Rémy Degenne, Pierre Ménard, Xuedong Shang, and Michal Valko. Gamification of pure exploration for linear bandits. In *International Conference on Machine Learning*, pages 2432–2442. PMLR, 2020.

Tanner Fiez, Lalit Jain, Kevin G Jamieson, and Lillian Ratliff. Sequential experimental design for transductive linear bandits. *Advances in neural information processing systems*, 32, 2019.

David A Freedman. On tail probabilities for martingales. *the Annals of Probability*, pages 100–118, 1975.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *Algorithmic Learning Theory: 22nd International Conference, ALT 2011, Espoo, Finland, October 5-7, 2011. Proceedings 22*, pages 174–188. Springer, 2011.

Daniel N Hill, Houssam Nassif, Yi Liu, Anand Iyer, and SVN Vishwanathan. An efficient bandit algorithm for realtime multivariate optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1813–1821, 2017.

Yassir Jedra and Alexandre Proutiere. Optimal best-arm identification in linear bandits. *Advances in Neural Information Processing Systems*, 33:10007–10017, 2020.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page III–1238–III–1246. JMLR.org, 2013.

Zohar S Karnin. Verification based solution for structured mab problems. *Advances in Neural Information Processing Systems*, 29, 2016.

Julian Katz-Samuels, Lalit Jain, Zohar Karnin, and Kevin Jamieson. An empirical process approach to the union bound: Practical algorithms for combinatorial and linear bandits. *Advances in Neural Information Processing Systems*, 33:10371–10382, 2020.

Ron Kohavi and Roger Longbotham. Unexpected results in online controlled experiments. *ACM SIGKDD Explorations Newsletter*, 12(2):31–35, 2011.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Chung-Wei Lee, Haipeng Luo, Chen-Yu Wei, Mengxiao Zhang, and Xiaojin Zhang. Achieving near instance-optimality and minimax-optimality in stochastic and adversarial linear bandits simultaneously. In *International Conference on Machine Learning*, pages 6142–6151. PMLR, 2021.

Optimizely. Stats accelerator – acceleration under time-varying signals. https://support.optimizely.com/hc/en-us/articles/5326213705101-Stats-Accelerator-Acceleration-Under-Time-Varying-Signals, May 2023.

Chao Qin and Daniel Russo. Adaptivity and confounding in multi-armed bandit experiments. *arXiv preprint arXiv:2202.09036*, 2022.

Yevgeny Seldin and Gábor Lugosi. An improved parametrization and analysis of the exp3++ algorithm for stochastic and adversarial bandits. In *Conference on Learning Theory*, pages 1743–1759. PMLR, 2017.

Yevgeny Seldin and Aleksandrs Slivkins. One practical algorithm for both stochastic and adversarial bandits. In *International Conference on Machine Learning*, pages 1287–1295. PMLR, 2014.

Marta Soare, Alessandro Lazaric, and Rémi Munos. Best-arm identification in linear bandits. *Advances in Neural Information Processing Systems*, 27, 2014.

Joe Suk and Samory Kpotufe. Tracking most significant arm switches in bandits, 2022.

Andrew Wagenmaker and Dylan J Foster. Instance-optimality in interactive decision making: Toward a non-asymptotic theory. *arXiv preprint arXiv:2304.12466*, 2023.

Chen-Yu Wei and Haipeng Luo. Non-stationary reinforcement learning without prior knowledge: An optimal black-box approach. In *Conference on Learning Theory*, pages 4300–4354. PMLR, 2021.

Chen-Yu Wei, Haipeng Luo, and Alekh Agarwal. Taking a hint: How to leverage loss predictors in contextual bandits? In *Conference on Learning Theory*, pages 3583–3634. PMLR, 2020.

Yuhang Wu, Zeyu Zheng, Guangyu Zhang, Zuohua Zhang, and Chu Wang. Non-stationary a/b tests. 2022.

Liyuan Xu, Junya Honda, and Masashi Sugiyama. A fully adaptive algorithm for pure exploration in linear bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 843–851. PMLR, 2018.

Yahoo! Yahoo! webscope dataset ydata-frontpage-todaymodule-clicks-v1_0, 2011. URL https://webscope.sandbox.yahoo.com/catalog.php?datatype=r.

Junwen Yang and Vincent Tan. Minimax optimal fixed-budget best arm identification in linear bandits. *Advances in Neural Information Processing Systems*, 35:12253–12266, 2022.

Junwen Yang and Vincent YF Tan. Towards minimax optimal best arm identification in linear bandits. *arXiv e-prints*, pages arXiv–2105, 2021.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]

(b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [No. While we provide a complete analysis of P1-RAGE, we also heuristically design two additional algorithms, P1-Peace and Mixed-Peace, for empirical comparisons without theoretical analysis.]

(c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:

(a) Statements of the full set of assumptions of all theoretical results. [Yes]

(b) Complete proofs of all theoretical results. [Yes]

(c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:

(a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]

(b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]

(c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]

(d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

(a) Citations of the creator If your work uses existing assets. [Yes]

(b) The license information of the assets, if applicable. [Yes]

(c) New assets either in the supplemental material or as a URL, if applicable. [Yes]

(d) Information about consent from data providers/curators. [Yes]

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

(a) The full text of instructions given to participants and screenshots. [Not Applicable]

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Contents

A ADDITIONAL ALGORITHMS IN IMPLEMENTATION

A.1 A Peace-based Robust Algorithm

In this section, we briefly explain how we design P1-Peace based on intuition similar to P1-RAGE and make it computationally efficient. First, we propose another subroutine, called Peace-Elimination, based on the elimination strategy in Peace Katz-Samuels et al. [2020], which has the same spirit as RAGE. Similar to RAGE-Elimination, Peace-Elimination also repeatedly computes $\mathcal{XY}$-allocation, but (virtually) eliminate arms so that the value of the remaining arms' optimal $\mathcal{XY}$-design is halved. In addition, in P1-Peace, we only update the sampling distribution λ_t after a period of time. The intuition is that if the environment is stationary, then we do not need to update our allocation probability frequently just like RAGE and Peace; if the environment is non-stationary, then the non-stationarity is handled by the mixed G-optimal design λ^*, which is fixed from the very beginning. Therefore, updating λ_t in a low frequency should not severely harm the performance. The new algorithm and elimination subroutine are summarized in Algorithm 4 and 5.

For convenience of presentation, for arm set $\mathcal{Z} \subset \mathbb{R}^d$ and distribution $\lambda \in \triangle_{\mathcal{X}}$, we define

$$\rho(\mathcal{Z}, \lambda) = \max_{x, x' \in \mathcal{Z}} \|x - x'\|^2_{A(\lambda)^{-1}} . \tag{5}$$

Algorithm 4 P1-Peace

Input: budget, $T \in \mathbb{N}$; arm set $\mathcal{X} \subset \mathbb{R}^d$
Compute epoch length $R \leftarrow \left\lfloor \frac{T}{\log_2(\inf_{\lambda \in \triangle_{\mathcal{X}}} \rho(\mathcal{X}, \lambda))} \right\rfloor$
Compute G-optimal design λ^* based on equation (3) and initialize $\lambda_1 = \lambda^*$
for $t = 1, 2, \ldots, T$ **do**
  Sample $x_t \sim \lambda_t$ and receive reward r_t
  Estimate $\widehat{\theta}_t \leftarrow \frac{1}{t} \sum_{s=1}^{t} \mathbb{E}_{x \sim \lambda_s} \left[x x^\top \right]^{-1} x_s r_s$
  $\lambda_{t+1} \leftarrow \lambda_t$
  if $t - 1 = cR$ for some integer c **then**
    Update $\lambda_{t+1} \leftarrow$Peace-Elimination$(\widehat{\theta}_t)$
  end if
end for
return $\operatorname{argmax}_{x \in \mathcal{X}} x^\top \widehat{\theta}_T$

Algorithm 5 Peace-Elimination

Input: arm set $\mathcal{X} \subset \mathbb{R}^d$; current estimate $\widehat{\theta}_t$
Find index $\widehat{(k)}_t$ such that $x_{\widehat{(1)}_t}^\top \widehat{\theta}_t \geq x_{\widehat{(2)}_t}^\top \widehat{\theta}_t \geq \cdots \geq x_{\widehat{(K)}_t}^\top \widehat{\theta}_t$
Initialize $\mathcal{X}_t^{(0)} \leftarrow \mathcal{X}$ and $i \leftarrow 0$
while $|\mathcal{X}_t^{(i)}| > 1$ **do**
  Compute $\lambda_t^{(i)} \leftarrow \operatorname{arginf}_{\lambda \in \triangle_{\mathcal{X}}} \rho(\mathcal{X}_t^{(i)}, \lambda)$
  Find the largest index k_i such that

$$\inf_{\lambda \in \triangle_{\mathcal{X}}} \rho\left(\{ x_{\widehat{(1)}_t}, \ldots, x_{\widehat{(k_i)}_t} \} \right) \leq \frac{1}{2} \cdot \inf_{\lambda \in \triangle_{\mathcal{X}}} \rho(\mathcal{X}_t^{(i)}, \lambda)$$

  Update $\mathcal{X}_t^{(i+1)} \leftarrow \left\{ x_{\widehat{(1)}_t}, \ldots, x_{\widehat{(k_i)}_t} \right\}$
  $i \leftarrow i + 1$
end while
return $(\bar{\lambda}_t + \lambda^*)/2$, where $\bar{\lambda}_t = \frac{1}{i} \sum_{i'=0}^{i-1} \lambda_t^{(i')}$

A.2 A Naive Baseline Mixed Algorithm

In this section, we present a naive mixture of Peace and the G-optimal design, called Mixed-Peace, which eliminates arms and computes design λ_k during each epoch exactly the same as Peace. The only differences are that Mixed-Peace uses IPS estimator and when pulling an arm, it will pull an arm by following $x_t \sim (\lambda_k + \lambda^*)/2$, where λ^* is the G-optimal design defined in equation (3). Its details are summarized in Algorithm 6.

Algorithm 6 Mixed-Peace

Input: budget, $T \in \mathbb{N}$; arm set $\mathcal{X} \subset \mathbb{R}^d$
Initialize $R \leftarrow \lceil \log_2 (\inf_{\lambda \in \triangle_{\mathcal{X}}} \rho(\mathcal{X}, \lambda)) \rceil$, $N \leftarrow \lfloor \frac{T}{R} \rfloor$, $\mathcal{X}_0 \leftarrow \mathcal{X}$, $\widehat{\theta}_0 \leftarrow \mathbf{0}$ and $t \leftarrow 1$
Compute G-optimal design λ^* using equation (3)
for $r = 0, \ldots, R$ **do**
  Find $\lambda_r \leftarrow (\operatorname{arginf}_{\lambda \in \triangle_{\mathcal{X}}} \rho(\mathcal{X}_r, \lambda) + \lambda^*)/2$
  while $t \le \min\{T, (r+1)N\}$ **do**
    Sample $x_t \sim \lambda_r$ and receive reward r_t
    Estimate $\widehat{\theta}_t \leftarrow \frac{t-1}{t} \cdot \widehat{\theta}_{t-1} + \frac{1}{t} \cdot \mathbb{E}_{x \sim \lambda_r} \left[x x^\top \right]^{-1} x_t r_t$
    $t \leftarrow t + 1$
  end while
  if $|\mathcal{X}_r| > 1$ **then**
    Reindex $\mathcal{X}_r$ such that $x_1^\top \widehat{\theta}_t \ge x_2^\top \widehat{\theta}_t \ge \cdots \ge x_{n_r}^\top \widehat{\theta}_t$, where $n_r = |\mathcal{X}_r|$
    Find the largest index k_r such that

$$\inf_{\lambda \in \triangle_{\mathcal{X}}} \rho(\{x_1, \ldots, x_{k_r}\}, \lambda) \le \frac{1}{2} \cdot \inf_{\lambda \in \triangle_{\mathcal{X}}} \rho(\mathcal{X}_r, \lambda)$$

    Update $\mathcal{X}_{r+1} \leftarrow \{x_1, \ldots, x_{k_r}\}$
  end if
end forreturn $\operatorname{argmax}_{x \in \mathcal{X}} x^\top \widehat{\theta}_T$

B ERROR PROBABILITY OF ALGORITHM 1 IN NON-STATIONARY ENVIRONMENTS

Theorem 1 (Error probability of G-BAI). *Fix time horizon T, arm set $\mathcal{X} \subset \mathbb{R}^d$ with $|\mathcal{X}| = K$ and arbitrary unknown parameters $\{\theta_t\}_{t=1}^T$. If we run Algorithm 1 in this non-stationary environment and obtain x_{J_T}, then it holds that*

$$\mathbb{P}_{\overline{\theta}_T} (J_T \neq (1)) \le K \exp\left(-\frac{T}{12 H_{\text{G-BAI}}(\overline{\theta}_T)} \right), \quad \text{where } H_{\text{G-BAI}}(\overline{\theta}_T) = \frac{d}{\Delta_{(1)}^2}.$$

Proof. Based on the recommendation rule $x_{J_T} = \operatorname{argmax}_{x \in \mathcal{X}} x^\top \widehat{\theta}_T$, we have

$$\begin{aligned}
\mathbb{P}(J_T \neq (1)) =& \mathbb{P}\left(\exists k \in [2:K] \text{ s.t. } x_{(k)}^\top \widehat{\theta}_T \ge x_{(1)}^\top \widehat{\theta}_T \right) \\
\le& \mathbb{P}\left(\exists k \in [2:K] \text{ s.t. } x_{(k)}^\top \widehat{\theta}_T - x_{(k)}^\top \overline{\theta}_T \ge \frac{\Delta_{(k)}}{2} \text{ or } x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \overline{\theta}_T \le -\frac{\Delta_{(1)}}{2} \right) \\
\le& \mathbb{P}\left(x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \overline{\theta}_T \le -\frac{\Delta_{(1)}}{2} \right) + \sum_{k=2}^{K} \mathbb{P}\left(x_{(k)}^\top \widehat{\theta}_T - x_{(k)}^\top \overline{\theta}_T \ge \frac{\Delta_{(k)}}{2} \right). \qquad (6)
\end{aligned}$$

The above terms can be bounded by Bernstein's inequality. In particular, for the first term, we have

$$\mathbb{P}\left(x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \overline{\theta}_T \le -\frac{\Delta_{(1)}}{2} \right) = \mathbb{P}\left(\sum_{t=1}^{T} x_{(1)}^\top \left(A(\lambda^*)^{-1} x_t r_t - \theta_t \right) \le -\frac{T \Delta_{(1)}}{2} \right).$$

Since IPS estimator is unbiased, $x_{(1)}^\top \left(A(\lambda^*)^{-1}x_t r_t - \theta_t\right)$ is a zero-mean random variable. Based on our bounded reward assumption, we have

$$\left|x_{(1)}^\top \left(A(\lambda^*)^{-1}x_t r_t - \theta_t\right)\right| \le \left|x_{(1)}^\top A(\lambda^*)^{-1}x_t\right| + 2 \le \left\|x_{(1)}\right\|_{A(\lambda^*)^{-1}} \left\|x_t\right\|_{A(\lambda^*)^{-1}} + 2 \le d + 2 \le 3d,$$

where we use the property of G-optimal design $\max_{x\in\mathcal{X}} \|x\|^2_{A(\lambda^*)^{-1}} \le d$. We can similarly bound its variance by

$$\begin{aligned}
\mathbb{E}\left[\left(x_{(1)}^\top \left(A(\lambda^*)^{-1}x_t r_t - \theta_t\right)\right)^2\right] \le& \mathbb{E}\left[\left(x_{(1)}^\top A(\lambda^*)^{-1}x_t\right)^2\right] \\
=& x_{(1)}^\top A(\lambda^*)^{-1}\mathbb{E}\left[x_t x_t^\top\right] A(\lambda^*)^{-1}x_{(1)} \\
=& x_{(1)}^\top A(\lambda^*)^{-1}A(\lambda^*)A(\lambda^*)^{-1}x_{(1)} \qquad \text{(Since } x_t \sim \lambda^* \text{ by algorithm)} \\
=& \left\|x_{(1)}\right\|^2_{A(\lambda^*)^{-1}} \le d
\end{aligned}$$

Thus, by Bernstein's inequality, we have

$$\mathbb{P}\left(x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \overline{\theta}_T \le -\frac{\Delta_{(1)}}{2}\right) \le \exp\left(-\frac{T^2\Delta_{(1)}^2/8}{Td + Td\Delta_{(1)}/2}\right) \le \exp\left(-\frac{T\Delta_{(1)}^2}{12d}\right),$$

where the last inequality uses the assumption that $\Delta_{(1)} \le 1$. By similarly applying Bernstein's inequality to other terms in (6), we can then have

$$\begin{aligned}
\mathbb{P}\left(J_T \ne x_{(1)}\right) \le& \mathbb{P}\left(x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \overline{\theta}_T \le -\frac{\Delta_{(1)}}{2}\right) + \sum_{k=2}^{K} \mathbb{P}\left(x_{(k)}^\top \widehat{\theta}_T - x_{(k)}^\top \overline{\theta}_T \ge \frac{\Delta_{(k)}}{2}\right) \\
\le& \sum_{k=1}^{K} \exp\left(-\frac{T\Delta_{(k)}^2}{12d}\right) \\
\le& K \exp\left(-\frac{T\Delta_{(1)}^2}{12d}\right).
\end{aligned}$$

□

C ERROR PROBABILITY OF ALGORITHM 2

C.1 Stationary Environments

We first prove an error probability of Algorithm 2 in stationary environments that contains unspecified parameters from the virtual phases. Without loss of generality, assume that the arms $x_1, \ldots, x_K$ are ordered such that $\theta^\top x_1 > \theta^\top x_2 \ge \cdots \ge \theta^\top x_K$ and $\Delta_1 = \Delta_2 \le \Delta_3 \le \cdots \le \Delta_K$.

Throughout this section, we will the following definitions: $i_0 = \lceil \log_2(1/\Delta_1) \rceil + 1$, $\mathcal{A}_i = \{x \in \mathcal{X} \mid \Delta_x \le 2 \cdot 2^{-i}\}$, $\bar{i}(k) = \max\{i \in [i_0 - 1] \mid \Delta_k \le 2^{-i}\}$ and

$$f(\mathcal{A}_i) = \min_{\lambda \in \triangle_\mathcal{X}} \max_{x, x' \in \mathcal{A}_i} \|x - x'\|^2_{A(\lambda)^{-1}}.$$

Theorem 3. *Let $\mathcal{D} = \{\mathbf{a} \in [0,1]^{i_0+1} \mid 0 = a_0 < a_1 \le a_2 \le \ldots \le a_{i_0} = 1\}$. Then, if $m \ge i_0$, The error probability of Algorithm 2 in a stationary environment with parameter θ is bounded as*

$$\begin{aligned}
\mathbb{P}_\theta\left(J_T \ne 1\right) \le& 2i_0 K T \exp\left(-\frac{T}{\overline{H}_{\textit{P1-RAGE}}(\theta)}\right), \\
\overline{H}_{\textit{P1-RAGE}}(\theta) =& \min_{\mathbf{a}\in\mathcal{D}} \max_{k\in[K]} \frac{48m \sum_{i'=1}^{\bar{i}(k)} (a_{i'} - a_{i'-1}) f(\mathcal{A}_{i'-2}) + 8(m\sqrt{df(\mathcal{X})} + 1)a_{\bar{i}(k)}\Delta_k}{3a^2_{\bar{i}(k)}\Delta_k^2}.
\end{aligned} \tag{7}$$

Proof. With $0 = n_0 < n_1 \le n_2 \le \ldots \le n_{i_0} = T$.[8] we define the event ξ_i with $i \ge 1$ as follows: after n_i samples all the arms with true gap smaller than $2 \cdot 2^{-i}$ are estimated with precision $2^{-i}/2$, which is

$$\xi_i = \{\forall t \ge n_i, \forall k \in [K] \text{ s.t. } \Delta_k \le 2 \cdot 2^{-i} \implies |\Delta_k - \widehat{\Delta}_k^{(t)}| < 2^{-i}/2\},$$

[8] We do not specify the values of $n_1, \ldots, n_{i_0 - 1}$ for now.

where $\widehat{\Delta}_k^{(t)} = (x_1 - x_k)^\top \widehat{\theta}^{(t)}$ for $k > 1$ and $\widehat{\Delta}_1^{(t)} = (x_1 - x_2)^\top \widehat{\theta}^{(t)}$. We first show how these events $\{\xi_i\}_{i=1}^{i_0}$ relate the correctness of Algorithm 2.

Correctness. If $\bigcap_{i=1}^{i_0} \xi_i$ holds then the algorithm successfully identifies the best arm. Indeed, if we assume it does not, then there must exist non-optimal arm k_0 such that $\widehat{\Delta}_{k_0}^{(T)} < 0$. As $\bigcap_{i=1}^{i_0} \xi_i$ holds, for some $i' \le i_0$, it holds that $2^{-i'} < \Delta_{k_0} \le 2 \cdot 2^{-i'}$ and then $|\Delta_{k_0} - \widehat{\Delta}_{k_0}^{(T)}| < 2^{-i'}/2$. Therefore, we have $2^{-i'} < \Delta_{k_0} \le \Delta_{k_0} - \widehat{\Delta}_{k_0}^{(T)} \le |\Delta_{k_0} - \widehat{\Delta}_{k_0}^{(T)}| \le 2^{-i'}/2$, which is a contradiction.

Thus, the error probability is upper bounded by $\mathbb{P}\left(\bigcup_{i=1}^{i_0} \xi_i^c\right)$, which gives us

$$
\begin{aligned}
\mathbb{P}\left(J_T \neq 1\right) \le & \mathbb{P}\left(\bigcup_{i=1}^{i_0} \xi_i^c\right) = \mathbb{P}\left(\bigcup_{i=1}^{i_0} \left(\xi_i^c \setminus \bigcup_{j=1}^{i-1} \xi_j^c\right)\right) \le \sum_{i=1}^{i_0} \mathbb{P}\left(\xi_i^c \setminus \bigcup_{j=1}^{i-1} \xi_j^c\right) \\
= & \sum_{i=1}^{i_0} \mathbb{P}\left(\xi_i^c \cap \left(\bigcup_{j=1}^{i-1} \xi_j^c\right)^c\right) = \sum_{i=1}^{i_0} \mathbb{P}\left(\xi_i^c \cap \left(\bigcap_{j=1}^{i-1} \xi_j\right)\right) \\
\le & \sum_{i=1}^{i_0} \mathbb{P}\left(\xi_i^c \,\middle|\, \bigcap_{j=1}^{i-1} \xi_j\right).
\end{aligned}
$$

Bernstein's inequality. Now, we just need to find an upper bound of $\mathbb{P}\left(\xi_i^c \,\middle|\, \bigcap_{j=1}^{i-1} \xi_j\right)$. Assume $\exists t \ge n_i, \exists k \in [K]$ s.t. $\Delta_k \le 2 \cdot 2^{-i}$.[9] Then, we have

$$
\begin{aligned}
& \mathbb{P}(|\Delta_k - \widehat{\Delta}_k^{(t)}| \ge 2^{-i}/2) \\
= & \mathbb{P}(|(\theta - \widehat{\theta}_t)^\top (x_1 - x_k)| \ge 2^{-i}/2) \\
= & \mathbb{P}\left(\left|\sum_{s=1}^{t} \left(\theta - A(\lambda_s)^{-1} x_s r_s\right)^\top (x_1 - x_k)\right| \ge 2^{-i} t/2\right) \\
\overset{(a)}{\le} & 2\exp\left(-\frac{2^{-2i} t^2/8}{2\sum_{s=1}^{t} \|x_1 - x_k\|^2_{A(\bar{\lambda}_s)^{-1}} + \left(\sqrt{d}\max_{s\in[1:t]} \|x_1 - x_k\|_{A(\bar{\lambda}_s)^{-1}} + 1\right) t 2^{-i}/3}\right) \\
& \qquad \text{(By Bernstein's inequality for martingale differences Freedman [1975])} \\
\le & 2\exp\left(-\frac{2^{-2i} t^2/8}{\text{term I}}\right),
\end{aligned}
\tag{8}
$$

$$
\begin{aligned}
\text{where term I} = & 2\sum_{i'=1}^{i} \sum_{s=n_{i'-1}+1}^{n_{i'}} \|x_1 - x_k\|^2_{A(\bar{\lambda}_s)^{-1}} + 2\sum_{s=n_i+1}^{t} \|x_1 - x_k\|^2_{A(\bar{\lambda}_s)^{-1}} \\
& + \left(\sqrt{d}\max_{s\in[1:t]} \|x_1 - x_k\|_{A(\bar{\lambda}_s)^{-1}} + 1\right) \cdot \frac{t 2^{-i}}{3}.
\end{aligned}
$$

Here, to use Bernstein's inequality for martingale differences in the inequality (a) above, we need to bound the variance and magnitude of $\left(\theta - A(\lambda_s)^{-1} x_s r_s\right)^\top (x_1 - x_k)$ condition on λ_s.[10] In particular, we have

$$
\begin{aligned}
\left|\left(\theta - A(\lambda_s)^{-1} x_s r_s\right)^\top (x_1 - x_k)\right| \le & \left|(x_1 - x_k)^\top A(\lambda_s)^{-1} x_s\right| + \Delta_k \\
\le & \|x_1 - x_k\|_{A(\lambda_s)^{-1}} \|x_s\|_{A(\lambda_s)^{-1}} + 2 \\
\le & 2\sqrt{d} \|x_1 - x_k\|_{A(\bar{\lambda}_s)^{-1}} + 2. \\
& \text{(Since } \lambda_s = (\bar{\lambda}_s + \lambda^*)/2 \text{ and } \lambda \mapsto \|x_1 - x_k\|^2_{A(\lambda)^{-1}} \text{ is convex in } \lambda)
\end{aligned}
$$

[9] Otherwise, ξ_i is vacuously true and $\mathbb{P}\left(\xi_i^c\right) = 0$.

[10] Since IPS estimator is unbiased and λ_s is determined by the history prior to time s, we have $\mathbb{E}\left[\left(\theta - A(\lambda_s)^{-1} x_s r_s\right)^\top (x_1 - x_k) \mid \mathcal{H}_{s-1}\right] = 0$, which implies that it is a martingale difference sequence.

$$
\begin{aligned}
&\mathbb{E}\left[\left(\left(\theta - A(\lambda_s)^{-1}x_s r_s\right)^\top (x_1 - x_k)\right)^2 \mid \lambda_s\right] \\
\leq &\mathbb{E}\left[\left((x_1 - x_k)^\top A(\lambda_s)^{-1} x_s\right)^2 \mid \lambda_s\right] \\
= &(x_1 - x_k)^\top A(\lambda_s)^{-1} \mathbb{E}\left[x_s x_s^\top \mid \lambda_s\right] A(\lambda_s)^{-1}(x_1 - x_k) \\
= &\|x_1 - x_k\|^2_{A(\lambda_s)^{-1}} \\
\leq &2\|x_1 - x_k\|^2_{A(\bar{\lambda}_s)^{-1}}. \qquad \text{(Since } \lambda_s = (\bar{\lambda}_s + \lambda^*)/2\text{)}
\end{aligned}
$$

Single-term error probability. Now, we need to use the property of the subroutine RAGE-Elimination (Line 3 of Algorithm 2) that generates λ_s. That is, by Lemma 3, since $x_k \in \mathcal{A}_i \subseteq \mathcal{A}_{i'}$ for $i' \leq i$ and $m \geq i_0$, for $s \in [n_{i'-1}+1, n_{i'}]$, we have $\|x_1 - x_k\|^2_{A(\bar{\lambda}_s)^{-1}} \leq m \inf_{\lambda \in \triangle_\mathcal{X}} \max_{x,x' \in \mathcal{A}_{i'-2}} \|x - x'\|^2_{A(\lambda)^{-1}} \stackrel{\text{def}}{=} m f(\mathcal{A}_{i'-2})$. Thus, we have

$$
\begin{aligned}
&\mathbb{P}(|\Delta_k - \widehat{\Delta}_k^{(t)}| \geq 2^{-i}/2) \\
\leq &2\exp\left(-\frac{2^{-2i}t^2/8}{2m\sum_{i'=1}^{i}(n_{i'} - n_{i'-1})f(\mathcal{A}_{i'-2}) + 2m(t - n_i)f(\mathcal{A}_{i-1}) + (m\sqrt{df(\mathcal{X})} + 1)t2^{-i}/3}\right) \\
\leq &2\exp\left(-\frac{2^{-2i}n_i^2/8}{2m\sum_{i'=1}^{i}(n_{i'} - n_{i'-1})f(\mathcal{A}_{i'-2}) + (m\sqrt{df(\mathcal{X})} + 1)n_i 2^{-i}/3}\right),
\end{aligned}
$$

where the last inequality above holds because of $t \geq n_i$ and a simple fact that $t \mapsto \frac{t^2}{at+b}$ is an increasing function when $t \geq 0$ if $a > 0$ and $b > 0$.

Final error probability. Then, with the union bound over all $t \geq n_i$ and $k \in [K]$, it holds for any $0 < n_1 \leq n_2 \ldots \leq n_i \leq T$ that

$$
\begin{aligned}
\mathbb{P}\left(\xi_i^c \,\middle|\, \bigcap_{j=1}^{i-1}\xi_j\right) &\leq 2KT\exp\left(-\frac{2^{-2i}n_i^2/8}{2m\sum_{i'=1}^{i}(n_{i'} - n_{i'-1})f(\mathcal{A}_{i'-2}) + (m\sqrt{df(\mathcal{X})} + 1)n_i 2^{-i}/3}\right) \\
&\leq 2KT\max_{k\in[K]}\exp\left(-\frac{3n_{\bar{i}(k)}^2\Delta_k^2}{48m\sum_{i'=1}^{\bar{i}(k)}(n_{i'} - n_{i'-1})f(\mathcal{A}_{i'-2}) + 8(m\sqrt{df(\mathcal{X})} + 1)n_{\bar{i}(k)}\Delta_k}\right),
\end{aligned}
$$

where $\bar{i}(k) = \max\{i \in [i_0 - 1] \mid \Delta_k \leq 2^{-i}\}$. Here, the last inequality use the same simple fact that $t \mapsto \frac{t^2}{at+b}$ is an increasing function when $t \geq 0$ if $a > 0$ and $b > 0$.

With values of $0 = n_0 < n_1 \leq n_2 \leq \cdots \leq n_{i_0} = T$, we can define $a_i = \frac{n_i}{T}$, which implies $0 = a_0 < a_1 \leq a_2 \leq \cdots \leq a_{i_0} = 1$. Since the choice of values $\mathbf{a} \in \mathcal{D}$ is arbitrary, the final error probability can be bounded as

$$
\begin{aligned}
\mathbb{P}(J_T \neq 1) &\leq \sum_{i=1}^{i_0}\mathbb{P}\left(\xi_j^c \,\middle|\, \bigcap_{j=1}^{i-1}\xi_j\right) \\
&\leq 2i_0 KT\min_{\mathbf{a}\in\mathcal{D}}\max_{k\in[K]}\exp\left(-\frac{3Ta_{\bar{i}(k)}^2\Delta_k^2}{48m\sum_{i'=1}^{\bar{i}(k)}(a_{i'} - a_{i'-1})f(\mathcal{A}_{i'-2}) + 8(m\sqrt{df(\mathcal{X})} + 1)a_{i(k)}\Delta_k}\right),
\end{aligned}
$$

which completes the proof □

C.1.1 Properties of RAGE-Elimination

In this section, we prove some properties of the RAGE-Elimination algorithm that will be useful for proving Theorem 3.

Lemma 1. *Assume $t \geq n_i$. Then, under $\bigcap_{j=1}^{i-1}\xi_j$, when running RAGE-Elimination (line 3 in Algorithm 2), it holds that*

$$
\mathcal{X}_t^{(i+1)} \subseteq \left\{x \in \mathcal{X} \mid \widehat{\Delta}_x^{(t)} \leq 2^{-i}\right\} \subseteq \mathcal{A}_i.
$$

Proof. To show $\mathcal{X}_t^{(i+1)} \subseteq \left\{x \in \mathcal{X} \mid \widehat{\Delta}_x^{(t)} \le 2^{-i}\right\}$, let $x_{\widehat{(1)}_t} = \operatorname{argmax}_{x\in\mathcal{X}} \left\langle \widehat{\theta}_t, x \right\rangle$. Then, for some arm x, if we have $\left\langle \widehat{\theta}^{(t)}, x_{\widehat{(1)}_t} - x \right\rangle \le 2^{-i}$, it holds that

$$\left\langle \widehat{\theta}^{(t)}, x_1 - x \right\rangle = \underbrace{\left\langle \widehat{\theta}^{(t)}, x_1 - x_{\widehat{(1)}_t} \right\rangle}_{\le 0} + \underbrace{\left\langle \widehat{\theta}^{(t)}, x_{\widehat{(1)}_t} - x \right\rangle}_{\le 2^{-i}} \le 2^{-i},$$

which implies $x \in \{x \in \mathcal{X} \mid \widehat{\Delta}_x^{(t)} \le 2^{-i}\}$.

To show $\left\{x \in \mathcal{X} \mid \widehat{\Delta}_x^{(t)} \le 2^{-i}\right\} \subseteq \mathcal{A}_i$, let $\widehat{\Delta}_x^{(t)} \le 2^{-i}$ for some x and assume for the sake of a contradiction that $\Delta_x > 2 \cdot 2^{-i}$. As $\Delta_x > 2 \cdot 2^{-i}$, there must exist $\tilde{i} \le i-1$ such that $2^{-\tilde{i}} < \Delta_x \le 2 \cdot 2^{-\tilde{i}}$. Then $|\Delta_x - \widehat{\Delta}_x^{(t)}| < 2^{-\tilde{i}}/2$ since event $\xi_{\tilde{i}}$ holds. Meanwhile, we have $\widehat{\Delta}_x^{(t)} \le 2^{-i} \le 2^{-\tilde{i}}/2$ since $\tilde{i} \le i-1$. Now, this leads to the contradiction

$$2^{-\tilde{i}}/2 = 2^{-\tilde{i}} - 2^{-\tilde{i}}/2 \le \Delta_x - \widehat{\Delta}_x^{(t)} \le |\Delta_x - \widehat{\Delta}_j^{(t)}| < 2^{-\tilde{i}}/2.$$

Thus, under $\bigcap_{j=1}^{i-1} \xi_j$, we have

$$\left\{x \in \mathcal{X} \mid \widehat{\Delta}_x^{(t)} \le 2^{-i}\right\} \subseteq \left\{x \in \mathcal{X} \mid \Delta_x \le 2 \cdot 2^{-i}\right\} = \mathcal{A}_i.$$

□

Lemma 2. *Assume $t \ge n_i$. Then, under $\bigcap_{j=1}^{i-1} \xi_j$, when running RAGE-Elimination, if $x \in \mathcal{A}_i$, then $x \in \mathcal{X}_t^{(i-1)}$.*

Proof. If $x \in \mathcal{A}_i$, then $\langle \theta, x_1 - x \rangle \le 2 \cdot 2^{-i}$. Again, let $x_{\widehat{(1)}_t} = \operatorname{argmax}_{x\in\mathcal{X}} \left\langle \widehat{\theta}_t, x \right\rangle$ and we have

$$\begin{aligned}
\left\langle \widehat{\theta}_t, \hat{x}_1^{(t)} - x \right\rangle &= \left\langle \widehat{\theta}_t, x_{\widehat{(1)}_t} - x_1 \right\rangle + \left\langle \widehat{\theta}_t, x_1 - x \right\rangle \\
&= \left\langle \widehat{\theta}_t, x_{\widehat{(1)}_t} - x_1 \right\rangle + \left\langle \widehat{\theta}_t - \theta, x_1 - x \right\rangle + \underbrace{\langle \theta, x_1 - x \rangle}_{\le 2\cdot 2^{-i}} \\
&\le \left\langle \widehat{\theta}_t, x_{\widehat{(1)}_t} - x_1 \right\rangle + |\widehat{\Delta}_x^{(t)} - \Delta_x| + 2 \cdot 2^{-i} \\
&\le \left\langle \widehat{\theta}_t, x_{\widehat{(1)}_t} - x_1 \right\rangle + 2^{-i} + 2 \cdot 2^{-i} && \text{(Since } \xi_{i-1} \text{ holds)} \\
&= -\widehat{\Delta}_{x_{\widehat{(1)}_t}}^{(t)} + 2^{-i} + 2 \cdot 2^{-i} \\
&\le 2^{-i} + 2^{-i} + 2 \cdot 2^{-i} \\
&= 4 \cdot 2^{-i}.
\end{aligned}$$

The last inequality above holds because under $\bigcap_{j=1}^{i-1} \xi_j$, by Lemma 1, we have $x_{\widehat{(1)}_t} \in \mathcal{A}_i$, meaning that $|\widehat{\Delta}_{x_{\widehat{(1)}_t}}^{(t)} - \Delta_{x_{\widehat{(1)}_t}}| < 2^{-i} \implies \widehat{\Delta}_{x_{\widehat{(1)}_t}}^{(t)} > \Delta_{x_{\widehat{(1)}_t}} - 2^{-i} > -2^{-i}$. □

Lemma 3. *Assume $t \ge n_i$ and $\bigcap_{j=1}^{i-1} \xi_j$ holds. When running RAGE-Elimination, If $x_k \in \mathcal{A}_i$, then*

$$\|x_1 - x_k\|^2_{A(\bar{\lambda}_t)^{-1}} \le m \min_{\lambda \in \triangle_{\mathcal{X}}} \max_{x,x' \in \mathcal{A}_{i-2}} \|x - x'\|^2_{A(\lambda)^{-1}}.$$

Proof. By Lemma 2, we have $x_1, x_k \in \mathcal{A}_i \implies x_1, x_k \in \mathcal{X}_t^{(i-1)}$, which means that $\left|\mathcal{X}_t^{(i-1)}\right| \ge 2$ and $\bar{\lambda}_t = \frac{1}{i_t} \sum_{i'=1}^{i_t} \lambda_t^{(i')}$ for some i_t satisfying $i-1 \le i_t \le m$. Thus, We have

$$\begin{aligned}
\|x_1 - x_k\|^2_{A(\bar{\lambda}_t)^{-1}} &\le m \|x_1 - x_k\|^2_{A\left(\lambda_t^{(i-1)}\right)^{-1}} \\
&\le m \max_{x,x' \in \mathcal{X}_t^{(i-1)}} \|x - x'\|^2_{A\left(\lambda_t^{(i-1)}\right)^{-1}} && \text{(Since } x_1, x_k \in \mathcal{X}_{i-1}^{(t)}\text{)}
\end{aligned}$$

$$\overset{\text{(i)}}{\leq} m \min_{\lambda \in \triangle_{\mathcal{X}}} \max_{x,x' \in \mathcal{A}_{i-2}} \|x - x'\|^2_{A(\lambda)^{-1}}.$$

Here, the above inequality (i) holds because by Lemma 1, we have $\mathcal{X}_t^{(i-1)} \subseteq \mathcal{A}_{i-2}$ and by algorithm construction, we have $\lambda_t^{(i-1)} \in \operatorname{argmin}_{\lambda \in \triangle_{\mathcal{X}}} \max_{x,x' \in \mathcal{X}_t^{(i-1)}} \|x - x'\|^2_{A(\lambda)^{-1}}$. □

C.1.2 Simplified Stationary Complexity and its Relation to Multi-armed Bandits

In this section, we simplify the complexity of Algorithm 2 obtained in Theorem 3 by appropriately choosing values $\mathbf{a} \in \mathcal{D}$. In particular, we have the following theorem.

Theorem 4. *For $\overline{H}_{\textit{P1-RAGE}}(\theta)$ defined in equation (7), we have*

$$\overline{H}_{\textit{P1-RAGE}}(\theta) \leq \frac{1024 m i_0}{\Delta_1} \inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{x \neq x_1} \frac{\|x - x_1\|^2_{A(\lambda)^{-1}}}{\Delta_x} + \frac{16m\sqrt{d}}{3\Delta_1} \inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{x \neq x_1} \|x - x_1\|_{A(\lambda)^{-1}} + \frac{1}{3\Delta_1}.$$

Proof. For $i \in \{1, \ldots, i_0 - 1\}$, we take $a_i = \frac{\Delta_1}{\Delta_{\bar{k}(i)}}$, where $\bar{k}(i) = \min\left\{k \in [K] \mid \Delta_k \geq \frac{2^{-i}}{2}\right\}$. Then, since $\bar{i}(k) = \max\{i \in [i_0 - 1] \mid \Delta_k \leq 2^{-i}\}$, for any $k \in [K]$, we have $\frac{2^{-\bar{i}(k)}}{2} \leq \Delta_{\bar{k}(\bar{i}(k))} \leq \Delta_k$, which further implies

$$a_{\bar{i}(k)} \Delta_k = \frac{\Delta_1}{\Delta_{\bar{k}(\bar{i}(k))}} \cdot \Delta_k \geq \Delta_1.$$

Then, for $\overline{H}_{\textsf{P1-RAGE}}(\theta)$ (defined in equation (7)), we have

$$\begin{aligned}
\overline{H}_{\textsf{P1-RAGE}}(\theta) &\leq \max_{k \in [K]} \left\{ \frac{16m \sum_{i'=1}^{\bar{i}(k)} (a_{i'} - a_{i'-1}) f(\mathcal{A}_{i'-2})}{a^2_{\bar{i}(k)} \Delta_k^2} + \frac{8(m\sqrt{df(\mathcal{X})} + 1)}{3 a_{\bar{i}(k)} \Delta_k} \right\} \\
&\leq \frac{16m}{\Delta_1} \max_{k \in [K]} \left\{ \frac{f(\mathcal{A}_{-1})}{\Delta_{\bar{k}(1)}} + \sum_{i'=2}^{\bar{i}(k)} \left(\frac{1}{\Delta_{\bar{k}(i')}} - \frac{1}{\Delta_{\bar{k}(i'-1)}} \right) f(\mathcal{A}_{i'-2}) \right\} + \frac{8(m\sqrt{df(\mathcal{X})} + 1)}{3\Delta_1}. \\
&\qquad \text{(Since } a_0 = 0 \text{ by definition)}
\end{aligned}$$

For the second term, using the definition of $f(\mathcal{X})$, we simply have

$$\begin{aligned}
\frac{8(m\sqrt{df(\mathcal{X})} + 1)}{3\Delta_1} &= \frac{8m\sqrt{d}}{3\Delta_1} \inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{x,x' \in \mathcal{X}} \|x - x_1 + x_1 - x'\|_{A(\lambda)^{-1}} + \frac{1}{3\Delta_1} \\
&\leq \frac{16m\sqrt{d}}{3\Delta_1} \inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{x \neq x_1} \|x - x_1\|_{A(\lambda)^{-1}} + \frac{1}{3\Delta_1}. \qquad (9)
\end{aligned}$$

For the first term, by fixing arm index $k \in [K]$ and defining $j \in \operatorname{argmax}_{\ell \in [\bar{i}(k)]} \frac{f(\mathcal{A}_{\ell-2})}{\Delta_{\bar{k}(\ell)}}$, we have

$$\begin{aligned}
&\frac{f(\mathcal{A}_{-1})}{\Delta_{\bar{k}(1)}} + \sum_{i'=2}^{\bar{i}(k)} \left(\frac{1}{\Delta_{\bar{k}(i')}} - \frac{1}{\Delta_{\bar{k}(i'-1)}} \right) f(\mathcal{A}_{i'-2}) \\
=& \frac{f(\mathcal{A}_{\bar{i}(k)-2})}{\Delta_{\bar{k}(\bar{i}(k))}} + \sum_{i'=1}^{\bar{i}(k)-1} \frac{f(\mathcal{A}_{i'-2}) - f(\mathcal{A}_{i'-1})}{\Delta_{\bar{k}(i')}} \\
\overset{\text{(a)}}{\leq}& \frac{f(\mathcal{A}_{j-2})}{\Delta_{\bar{k}(j)}} \left(1 + \sum_{i'=1}^{\bar{i}(k)-1} \frac{f(\mathcal{A}_{i'-2}) - f(\mathcal{A}_{i'-1})}{f(\mathcal{A}_{i'-2})} \right) \\
\leq& \bar{i}(k) \frac{f(\mathcal{A}_{j-2})}{\Delta_{\bar{k}(j)}} && \text{(Since } f(\mathcal{A}_{i'-2}) \geq f(\mathcal{A}_{i'-1})\text{)} \\
\leq& i_0 \max_{\ell \in [\bar{i}(k)]} \frac{f(\mathcal{A}_{\ell-2})}{\Delta_{\bar{k}(\ell)}} && \text{(Since } \bar{i}(k) \leq i_0 \text{ for any } k \in [K]\text{)}
\end{aligned}$$

$$
\begin{aligned}
=& i_0 \max_{\ell\in[\bar{i}(k)]} \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x,x'\in\mathcal{A}_{\ell-2}} \frac{\|x-x'\|^2_{A(\lambda)^{-1}}}{\Delta_{\bar{k}(\ell)}} \\
\leq& i_0 \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{\ell\in[\bar{i}(k)]} \max_{x,x'\in\mathcal{A}_{\ell-2}} \frac{\|x-x'\|^2_{A(\lambda)^{-1}}}{\Delta_{\bar{k}(\ell)}} && \text{(By the weak duality inequality)} \\
\leq& 64 i_0 \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{\ell\in[\bar{i}(k)]} \max_{x\in\mathcal{A}_{\ell-2},x\neq x_1} \frac{\|x-x_1\|^2_{A(\lambda)^{-1}}}{16\Delta_{\bar{k}(\ell)}} && \text{(By reasoning similar to equation (9))} \\
\overset{\text{(b)}}{\leq}& 64 i_0 \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{\ell\in[\bar{i}(k)]} \max_{x\in\mathcal{A}_{\ell-2},x\neq x_1} \frac{\|x-x_1\|^2_{A(\lambda)^{-1}}}{\Delta_x} \\
\leq& 64 i_0 \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x\neq x_1} \frac{\|x-x_1\|^2_{A(\lambda)^{-1}}}{\Delta_x}.
\end{aligned}
$$

Here, the inequality (a) above holds because $f(\mathcal{A}_{i'-2}) \geq f(\mathcal{A}_{i'-1})$ and by definition of j, we have $\frac{f(\mathcal{A}_{\ell-2})}{\Delta_{\bar{k}(\ell)}} \leq \frac{f(\mathcal{A}_{j-2})}{\Delta_{\bar{k}(j)}}$. The inequality (b) above holds because by definitions of $\bar{k}(\ell) = \min\left\{k\in[K] \mid \Delta_k \geq \frac{2^{-i}}{2}\right\}$ and $\mathcal{A}_{\ell-2} = \left\{x\in\mathcal{X} \mid \Delta_x \leq 2\cdot 2^{-(\ell-2)}\right\}$, we have $16\Delta_{\bar{k}(\ell)} \geq \Delta_x$ for any $x\in\mathcal{A}_{\ell-2}$.

Therefore, by plugging the bound of both terms back, we have

$$
\overline{H}_{\text{P1-RAGE}}(\theta) \leq \frac{1024 m i_0}{\Delta_1} \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x\neq x_1} \frac{\|x-x_1\|^2_{A(\lambda)^{-1}}}{\Delta_x} + \frac{16m\sqrt{d}}{3\Delta_1} \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x\neq x_1} \|x-x_1\|_{A(\lambda)^{-1}} + \frac{1}{3\Delta_1}.
$$

□

In the following corollary, we show that the above simplified complexity is in a same order (up to logarithmic factors) of H_{BOB} proposed in Abbasi-Yadkori et al. [2018].

Corollary 1. *In multi-armed bandits, meaning $d = K$ and $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_K\}$, for $H_{\text{P1-RAGE}}(\theta)$ (defined in equation (4)), if $m = i_0$, we then have*

$$
H_{\text{P1-RAGE}}(\theta) \leq \frac{2i_0\left(i_0\log(2K)+1\right)}{\Delta_{(1)}} \max_{k\in[K]} \frac{k}{\Delta_{(k)}} = 2i_0\left(i_0\log(2K)+1\right) H_{\text{BOB}}(\theta).
$$

Proof. When in multi-armed bandits, for the first term in $H_{\text{P1-RAGE}}(\theta)$, we have

$$
\inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x\neq x_{(1)}} \frac{\|x-x_{(1)}\|^2_{A(\lambda)^{-1}}}{\Delta_x} \leq 2\sum_{k=1}^{K} \frac{1}{\Delta_k} \leq 2\log(2K) \max_{k\in[K]} \frac{k}{\Delta_{(k)}},
$$

where the first inequality above comes from Soare et al. [2014] and the second inequality comes from Audibert et al. [2010]. For the second term in $H_{\text{P1-RAGE}}(\theta)$, we have

$$
\inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x\neq x_{(1)}} \|x-x_{(1)}\|_{A(\lambda)^{-1}} = \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{k\neq(1)} \sqrt{\frac{1}{\lambda_{(1)}} + \frac{1}{\lambda_k}} = \sqrt{2K},
$$

which then gives us $\frac{\sqrt{K}\cdot\sqrt{2K}}{\Delta_{(1)}} \leq \frac{2K}{\Delta_{(1)}\Delta_{(K)}} \leq \frac{2}{\Delta_{(1)}} \max_{k\in[K]} \frac{k}{\Delta_{(k)}}$.

Finally, by plugging these inequalities back into $H_{\text{P1-RAGE}}(\theta)$ (defined in equation (4)), we have

$$
\begin{aligned}
H_{\text{P1-RAGE}}(\theta) =& \frac{mi_0}{\Delta_{(1)}} \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x\neq x_{(1)}} \frac{\|x-x_{(1)}\|^2_{A(\lambda)^{-1}}}{\Delta_x} + \frac{m\sqrt{d}}{\Delta_{(1)}} \inf_{\lambda\in\triangle_{\mathcal{X}}} \max_{x\neq x_{(1)}} \|x-x_{(1)}\|_{A(\lambda)^{-1}} \\
\leq& \frac{2i_0^2\log(2K)}{\Delta_{(1)}} \max_{k\in[K]} \frac{k}{\Delta_{(k)}} + \frac{2i_0}{\Delta_{(1)}} \max_{k\in[K]} \frac{k}{\Delta_{(k)}} \\
=& \frac{2i_0\left(i_0\log(2K)+1\right)}{\Delta_{(1)}} \max_{k\in[K]} \frac{k}{\Delta_{(k)}}.
\end{aligned}
$$

□

C.1.3 Approximate BAI of Algorithm 2

Corollary 2. *Fix arm set $\mathcal{X} \subset \mathbb{R}^d$ with $|\mathcal{X}| = K$ and budget T. For a stationary environment with unknown parameter θ, if $m \geq i_0(\epsilon) = \lceil \log_2(1/\epsilon) \rceil + 1$ for some $\epsilon \geq \Delta_1$, then there exists absolute constant $c > 0$ such that the error probability of P1-RAGE satisfies*

$$\mathbb{P}_\theta\left(J_T \notin \mathcal{A}(\epsilon)\right) \leq 2i_0(\epsilon)KT \exp\left(-\frac{cT}{H_{\text{P1-RAGE}}(\theta, \epsilon)}\right),$$

where $\mathcal{A}(\epsilon) = \{x \in \mathcal{X} \mid \Delta_x \leq \epsilon\}$ and $H_{\text{P1-RAGE}}(\theta, \epsilon)$ is defined as replacing i_0 by $i_0(\epsilon)$ in $H_{\text{P1-RAGE}}(\theta)$ (defined in Eq. (7)).

Proof. The proof is the same as Theorem 2 through simply replacing i_0 by $i_0(\epsilon)$. □

C.2 Non-stationary Environments

In this section, we prove the error probability of Algorithm 2 in general non-stationary environments.

Theorem 5. *Fix time horizon T, arm set $\mathcal{X} \subset \mathbb{R}^d$ with $|\mathcal{X}| = K$ and arbitrary unknown parameters $\{\theta_t\}_{t=1}^T$. If we run Algorithm 2 in this non-stationary environment and obtain x_{J_T}, then it holds that*

$$\mathbb{P}_{\bar{\theta}_T}\left(J_T \neq (1)\right) \leq K \exp\left(-\frac{3T\Delta_{(1)}^2}{64d}\right).$$

Proof. The proof will basically resemble the one for Theorem 1. In particular, by the same reasoning to obtain equation 6, we have

$$\mathbb{P}\left(J_T \neq (1)\right) \leq \mathbb{P}\left(x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \bar{\theta}_T \leq -\frac{\Delta_{(1)}}{2}\right) + \sum_{k=2}^K \mathbb{P}\left(x_{(k)}^\top \widehat{\theta}_T - x_{(k)}^\top \bar{\theta}_T \geq \frac{\Delta_{(k)}}{2}\right),$$

$$\text{where } \mathbb{P}\left(x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \bar{\theta}_T \leq -\frac{\Delta_{(1)}}{2}\right) = \mathbb{P}\left(\sum_{t=1}^T x_{(1)}^\top \left(A(\lambda_t)^{-1} x_t r_t - \theta_t\right) \leq -\frac{T\Delta_{(1)}}{2}\right).$$

Since $\lambda_t = \frac{\bar{\lambda}_t + \lambda^*}{2}$ and $\lambda \mapsto \|x\|^2_{A(\lambda)^{-1}}$ is convex in λ, to use the Berstein's inequality for martingale differences [Freedman, 1975], we have

$$\left|x_{(1)}^\top \left(A(\lambda_t)^{-1} x_t r_t - \theta_t\right)\right| \leq 2\left\|x_{(1)}\right\|_{A(\lambda^*)^{-1}} \|x_t\|_{A(\lambda^*)^{-1}} + 2 \leq 2d + 2 \leq 4d,$$

$$\mathbb{E}\left[\left(x_{(1)}^\top \left(A(\lambda_t)^{-1} x_t r_t - \theta_t\right)\right)^2 \mid \lambda_t\right] = \left\|x_{(1)}\right\|^2_{A(\lambda_t)^{-1}} \leq 2\left\|x_{(1)}\right\|^2_{A(\lambda^*)^{-1}} \leq 2d.$$

Therefore, we have

$$\mathbb{P}\left(x_{(1)}^\top \widehat{\theta}_T - x_{(1)}^\top \bar{\theta}_T \leq -\frac{\Delta_{(1)}}{2}\right) \leq \exp\left(-\frac{T\Delta_{(1)}^2/8}{2d + 2d\Delta_{(1)}/3}\right) \leq \exp\left(-\frac{3T\Delta_{(1)}^2}{64d}\right).$$

By applying the same inequality to other terms, we have

$$\mathbb{P}\left(J_T \neq (1)\right) \leq K \exp\left(-\frac{3T\Delta_{(1)}^2}{64d}\right).$$

□

D IMPLEMENTATION DETAILS AND ADDITIONAL EXPERIMENTS

In this section, we provide more implementation details and additional experiment results. Experiments are executed through Python 3.10 and paralleled by a Mac M1 Pro chip with 6 cores.

First, we notice that an algorithm for stationary environments usually determines a batch of arms to pull at once during each epoch, while in non-stationary environment, the order of pulling these arms will affect the rewards it will receive. Therefore, when applying stationary algorithms (Peace and OD-LinBAI) into a non-stationary environment, we use a random permutation to determine the order of pulling for each batch of arms.

When implementing P1-RAGE, to be computationally efficient, we update λ_t in the same frequency as P1-Peace, which is summarized in Algorithm 4. We take $m = 15$ for P1-RAGE, which, based on Theorem 2, is valid as long as $\Delta_{(1)} \geq 2^{-13} \approx 1.22 \times 10^{-4}$. Furthermore, when implementing Peace, for simplicity, we use $\inf_{\lambda \in \triangle_{\mathcal{X}}} \rho(\mathcal{Z}, \lambda)$, defined in equation (5), to replace all $\gamma(\mathcal{Z})$ used in Katz-Samuels et al. [2020]. Since the paper of OD-LinBAI does not provide code, we implement it based on the pseudocode in Yang and Tan [2022]. Finally, we use Frank-Wolfe algorithm with stepsize $\frac{1}{2(i+2)}$ in i-th iteration to solve all optimization problems in a form of $\inf_{\lambda \in \triangle_{\mathcal{X}}} \max_{y \in \mathcal{Y}} \|y\|^2_{A(\lambda)^{-1}}$.

As for code snippets reference, we use part of the code from Katz-Samuels et al. [2020] to implement the rounding procedure used in Peace[11] and part of the code from Fiez et al. [2019] to generate the base stationary instance for the multivariate testing example.[12] We also use code from Xu et al. [2018] to preprocess the Yahoo! Webscope dataset.[13]

D.1 Additional Experiments

Here, we provide experiment results on some additional examples to corroborate our theoretical findings.

Malicious non-stationary example Because of the nature of arm elimination, algorithms designed for stationary environment can fail easily in some malicious non-stationary environments. Here, we pick the same $\mathcal{X}$ as Soare et al. [2014]'s stationary benchmark example and set $\omega = 0.5$. Then, we take

$$\theta_t = \begin{cases} \begin{bmatrix} 0 & 1 & 1 & \dots & 1 \end{bmatrix}^\top & \text{for } t = 1, \dots, \frac{T}{3}, \\ \begin{bmatrix} 2 & 0 & 0 & \dots & 0 \end{bmatrix}^\top & \text{for } t = \frac{T}{3} + 1, \dots, T. \end{cases}$$

We can see that the overall best arm is still $x_{(1)} = \mathbf{e}_1$. However, because of the θ_t in the first $1/3$ rounds, algorithms like Peace and OD-LinBAI will eliminate $\mathbf{e}_1$ in its initial phase; on the other hand, our algorithms will be robust to this non-stationarity. Here, we take $T = 10^4$ and the results are shown in right plot of Figure 5.

Stationary multivariate testing example We also test the performance of these algorithms in multivariate testing example when there is no non-stationarity, i.e. $\theta_t = \theta^*$ for all t. Here, we also take $T = 10^4$ and the results are shown in Figure 6. We can see that our robust algorithm P1-RAGE again performs better than G-BAI and comparably with Peace.

Non-stationary benchmark example In this example, we add non-stationarity to Soare et al. [2014]'s stationary benchmark example in a more structured instead of malicious way. In particular, we keep the arm set $\mathcal{X}$ the same, take $\omega = 0.5$ and set

$$\theta_t = \begin{bmatrix} 0.3 & 0 & 0 & \dots & -s \sin\left(\frac{2\pi t}{L}\right) + 0.5 \end{bmatrix}^\top,$$

where s is the oscillation scale and L is the oscillation period, In the first series of instances, we fix $L = 200$ and take values $m \in \{0, 1, \dots, 9\}$; in the second series of instances, we fix $m = 1$ and take values $L \in \{300, 600, \dots, 3000\}$. All non-stationary instances have the same optimal arm as their stationary counterparts and we take $T = 10^4$ for

[11] No license information.
[12] Under MIT License.
[13] No license information.

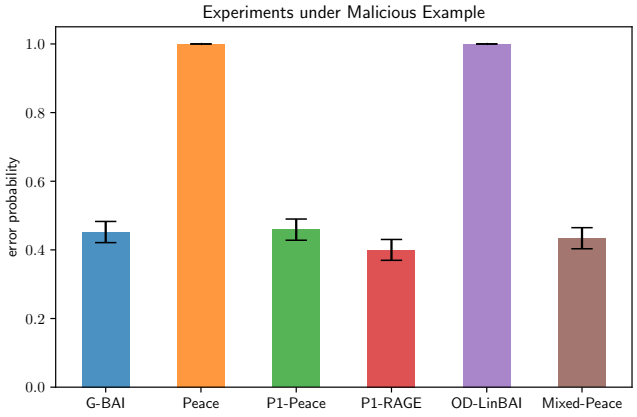

Figure 5: The error probabilities are estimated through 1000 repeated trials and the error bars represent 95% confidence intervals.

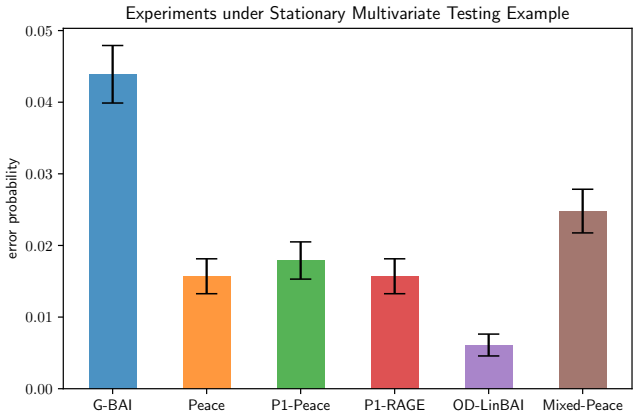

Figure 6: The error probabilities are estimated through 10^4 repeated trials and the error bars represent 95% confidence intervals.

all of these instances. The results are shown in Figure 7, from which we can see similar phenomenon as in Figure 3. In particular, algorithms designed for stationary environments, Peace and OD-LinBAI, are very unstable in face of non-stationarity. Meanwhile, among the other four relatively robust algorithms, our algorithms P1-RAGE and P1-Peace consistently outperform the other two.

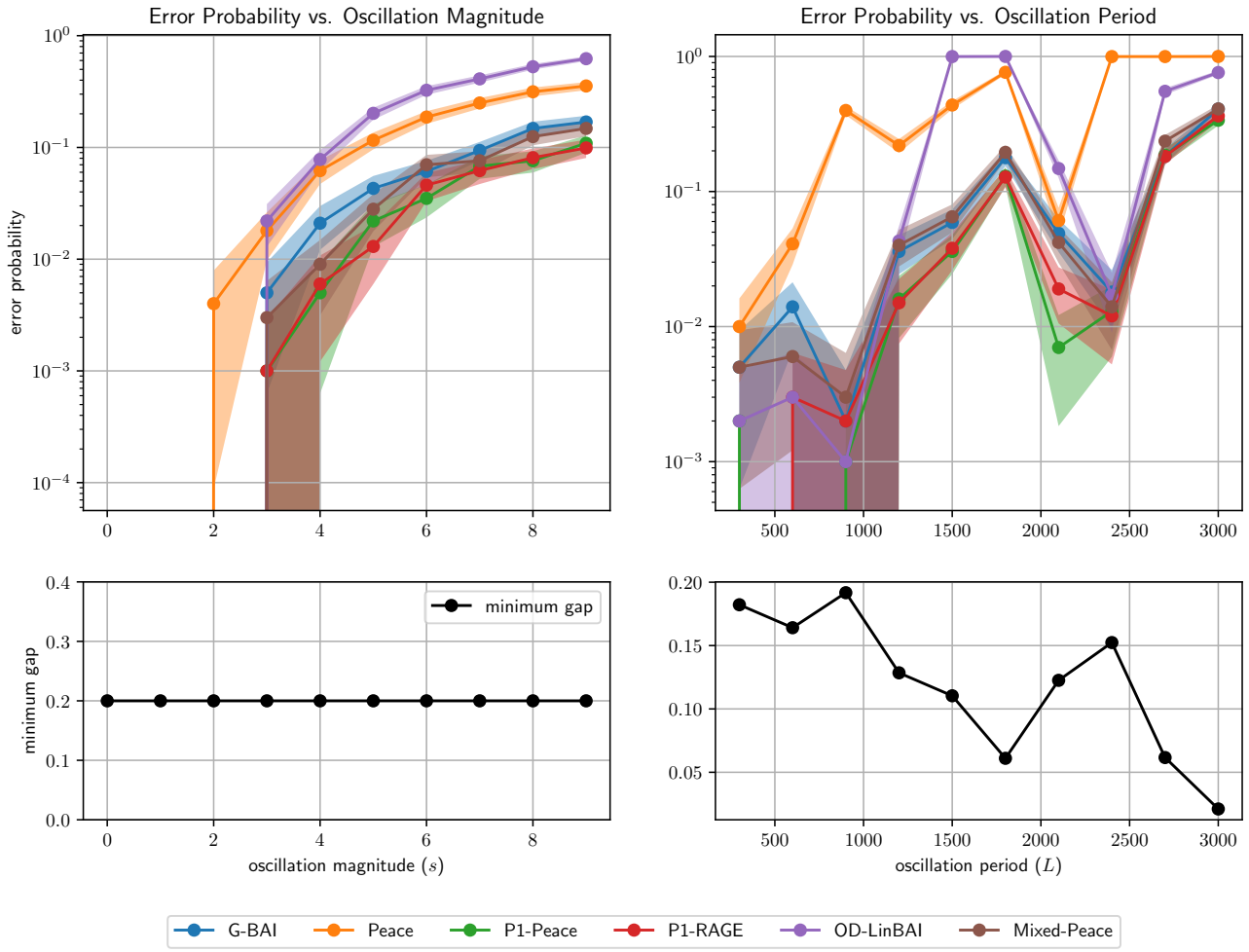

Figure 7: The vertical axis (error probability) is in log scale. The shaded area represents the 95% confidence interval. Each error probability is estimated through 1000 repeated trials. The bottom two plots give the minimum gap $\Delta_{(1)}$ of each instance that algorithms run over