Fixed-Budget Best-Arm Identification in Structured Bandits

MohammadJavad Azizi[1] , **Branislav Kveton**[2*] , **Mohammad Ghavamzadeh**[3]
[1]University of Southern California
[2]Amazon
[3]Google Research
azizim@usc.edu, bkveton@amazon.com, ghavamza@google.com

Abstract

Best-arm identification (BAI) in a fixed-budget setting is a bandit problem where the learning agent maximizes the probability of identifying the optimal (best) arm after a fixed number of observations. Most works on this topic study unstructured problems with a small number of arms, which limits their applicability. We propose a general tractable algorithm that incorporates the structure, by successively eliminating suboptimal arms based on their mean reward estimates from a joint generalization model. We analyze our algorithm in linear and generalized linear models (GLMs), and propose a practical implementation based on a G-optimal design. In linear models, our algorithm has competitive error guarantees to prior works and performs at least as well empirically. In GLMs, this is the first practical algorithm with analysis for fixed-budget BAI.

1 Introduction

Best-arm identification (BAI) is a *pure exploration* bandit problem where the goal is to identify the optimal arm. It has many applications, such as online advertising, recommender systems, and vaccine tests [Hoffman *et al.*, 2014; Lattimore and Szepesvári, 2020]. In *fixed-budget (FB)* BAI [Bubeck *et al.*, 2009; Audibert *et al.*, 2010], the goal is to accurately identify the optimal arm within a fixed budget of observations (arm pulls). This setting is common in applications where the observations are costly. However, it is more complex to analyze than the *fixed-confidence (FC)* setting, due to complications in budget allocation [Lattimore and Szepesvári, 2020, Section 33.3]. In FC BAI, the goal is to find the optimal arm with a guaranteed level of confidence, while minimizing the sample complexity.

Structured bandits are bandit problems in which the arms share a common structure, e.g., *linear* or *generalized linear* models [Filippi *et al.*, 2010; Soare *et al.*, 2014]. BAI in structured bandits has been mainly studied in the FC setting with the linear model [Soare *et al.*, 2014; Xu *et al.*, 2018; Degenne *et al.*, 2020]. The literature of FB BAI for linear bandits was limited to `BayesGap` [Hoffman *et al.*, 2014] for a long time. This algorithm does not explore sufficiently, and thus, performs poorly [Xu *et al.*, 2018]. [Katz-Samuels *et al.*, 2020] recently proposed `Peace` for FB BAI in linear bandits. Although this algorithm has desirable theoretical guarantees, it is computationally intractable, and its approximation loses the desired properties of the exact form. `OD-LinBAI` [Yang and Tan, 2021] is a concurrent work for FB BAI in linear bandits. It is a sequential halving algorithm with a special first stage, in which most arms are eliminated. This makes the algorithm inaccurate when the number of arms is much larger than the number of features, a common setting in structured problems. We discuss these three FB BAI algorithms in detail in Section 7 and empirically evaluate them in Section 8.

In this paper, we address the shortcomings of prior work by developing a general successive elimination algorithm that can be applied to several FB BAI settings (Section 3). The key idea is to divide the budget into multiple stages and allocate it *adaptively* for exploration in each stage. As the allocation is updated in each stage, our algorithm adaptively eliminates suboptimal arms, and thus, properly addresses the important trade-off between *adaptive* and *static* allocation in structured BAI [Soare *et al.*, 2014; Xu *et al.*, 2018]. We analyze our algorithm in *linear* bandits in Section 4. In Section 5, we extend our algorithm and analysis to *generalized linear models* (GLMs) and present the first BAI algorithm for these models. Our error bounds in Sections 4 and 5 motivate the use of a G-optimal allocation in each stage, for which we derive an efficient algorithm in Section 6. Using extensive experiments in Section 8, we show that our algorithm performs at least as well as a number of baselines, including `BayesGap`, `Peace`, and `OD-LinBAI`.

2 Problem Formulation

We consider a general stochastic bandit with K arms. The reward distribution of each arm $i \in \mathcal{A}$ (the set of K arms) has mean μ_i. Without loss of generality, we assume that $\mu_1 > \mu_2 \geq \cdots \geq \mu_K$; thus arm 1 is optimal. Let $x_i \in \mathbb{R}^d$ be the feature vector of arm i, such that $\sup_{i \in \mathcal{A}} ||x_i|| \leq L$ holds, where $||\cdot||$ is the ℓ_2-norm in $\mathbb{R}^d$. We denote the observed rewards of arms by $y \in \mathbb{R}$. Formally, the reward of arm i is $y = f(x_i) + \epsilon$, where ϵ is a σ^2-sub-Gaussian noise and $f(x_i)$ is any function of x_i, such that $\mu_i = f(x_i)$. In this paper, we

*This work started prior to joining Amazon.

focus on two instances of f: linear (Eq. (1)) and generalized linear (Eq. (4)).

We denote by B the fixed budget of arm pulls and by ζ the arm returned by the BAI algorithm. In the FB setting, the goal is to minimize the probability of error, i.e., $\delta = \Pr(\zeta \neq 1)$ [Bubeck *et al.*, 2009]. This is in contrast to the FC setting, where the goal is to minimize the sample complexity of the algorithm for a given upper bound on δ.

3 Generalized Successive Elimination

Successive elimination [Karnin *et al.*, 2013] is a popular BAI algorithm in multi-armed bandits (MABs). Our algorithm, which we refer to as Generalized Successive Elimination (`GSE`), generalizes it to structured reward models f. We provide the pseudo-code of `GSE` in Algorithm 1.

`GSE` operates in $s = \lceil \log_\eta K \rceil$ stages, where η is a tunable elimination parameter, usually set to be 2. The budget B is split evenly over s stages, and thus, each stage has budget $n = \lfloor B/s \rfloor$. In each stage $t \in [s]$, `GSE` pulls arms for n times and eliminates $1 - 1/\eta$ fraction of them. We denote the set of the remaining arms at the beginning of stage t by $\mathcal{A}_t$. By construction, only a single arm remains after s stages. Thus, $\mathcal{A}_1 = \mathcal{A}$ and $\mathcal{A}_{s+1} = \{\zeta\}$. In stage t, `GSE` performs the following steps:

Projection (Line 2): To avoid singularity issues, we project the remaining arms into their spanned subspace with $d_t \leq d$ dimensions. We discuss this more after Eq. (1).

Exploration (Line 3): The arms in $\mathcal{A}_t$ are sampled according to an allocation vector $\Pi_t \in \mathbb{N}^{\mathcal{A}_t}$, i.e., $\Pi_t(i)$ is the number of times that arm i is pulled in stage t. In Sections 4 and 5, we first report our results for general Π_t and then show how they can be improved if Π_t is an *adaptive* allocation based on the G-optimal design, described in Section 6.

Estimation (Line 4): Let $X_t = (X_{1,t}, \ldots, X_{n,t})$ and $Y_t = (Y_{1,t}, \ldots, Y_{n,t})$ be the feature vectors and rewards of the arms sampled in stage t, respectively. Given the reward model f, X_t, and Y_t, we estimate the mean reward of each arm i in stage t, and denote it by $\hat{\mu}_{i,t}$. For instance, if f is a linear function, $\hat{\mu}_{i,t}$ is estimated using linear regression, as in Eq. (1).

Elimination (Line 5): The arms in $\mathcal{A}_t$ are sorted in descending order of $\hat{\mu}_{i,t}$, their top $1/\eta$ fraction is kept, and the remaining arms are eliminated.

At the end of stage s, only one arm remains, which is *returned* as the optimal arm. While this algorithmic design is standard in MABs, it is not obvious that it would be near-optimal in structured problems, as this paper shows.

4 Linear Model

We start with the linear reward model, where $\mu_i = f(x_i) = x_i^\top \theta_*$, for an unknown reward parameter $\theta_* \in \mathbb{R}^d$. The estimate $\hat{\theta}_t$ of θ_* in stage t is computed using least-squares regression as $\hat{\theta}_t = V_t^{-1} b_t$, where $V_t = \sum_{j=1}^n X_{j,t} X_{j,t}^\top$ is the sample covariance matrix, and $b_t = \sum_{j=1}^n X_{j,t} Y_{j,t}$. This gives us the following mean estimate for each arm $i \in \mathcal{A}_t$,

$$\hat{\mu}_{i,t} = x_i^\top \hat{\theta}_t \,. \tag{1}$$

Algorithm 1 `GSE`: Generalized Successive Elimination

Input: Elimination hyper-parameter η, budget B
Initialization: $\mathcal{A}_1 \leftarrow \mathcal{A},\ t \leftarrow 1,\ s \leftarrow \lceil \log_\eta K \rceil$

1: **while** $t \leq s$ **do**
2: **Projection**: Project $\mathcal{A}_t$ to d_t dimensions, such that $\mathcal{A}_t$ spans $\mathbb{R}^{d_t}$
3: **Exploration**: Explore $\mathcal{A}_t$ using the allocation Π_t
4: **Estimation**: Calculate $(\hat{\mu}_{i,t})_{i \in \mathcal{A}_t}$ based on observed X_t and Y_t, using Eqs. (1) or (4)
5: **Elimination**: $\mathcal{A}_{t+1} = \underset{\mathcal{A} \subset \mathcal{A}_t : |\mathcal{A}| = \lceil \frac{|\mathcal{A}_t|}{\eta} \rceil}{\arg\max} \sum_{i \in \mathcal{A}} \hat{\mu}_{i,t}$
6: $t \leftarrow t + 1$
7: **end while**
8: **Output**: ζ such that $\mathcal{A}_{s+1} = \{\zeta\}$

The matrix V_t^{-1} is well-defined as long as X_t spans $\mathbb{R}^d$. However, since `GSE` eliminates arms, it may happen that the arms in later stages do not span $\mathbb{R}^d$. Thus, V_t could be singular and V_t^{-1} would not be well-defined. We alleviate this problem by projecting[1] the arms in $\mathcal{A}_t$ into their spanned subspace. We denote the dimension of this subspace by d_t. Alternatively, we can address the singularity issue by using the pseudo-inverse of matrices [Huang *et al.*, 2021]. In this case, we remove the projection step, and replace V_t^{-1} with its pseudo-inverse.

4.1 Analysis

In this section, we prove an error bound for `GSE` with the linear model. Although this error bound is a special case of that for GLMs (see Theorem 2), we still present it because more readers are familiar with linear bandit analysis than GLMs. To reduce clutter, we assume that all logarithms have base η. We denote by $\Delta_i = \mu_1 - \mu_i$, the sub-optimality gap of arm i, and by $\Delta_{\min} = \min_{i>1} \Delta_i$, the minimum gap, which by the assumption in Section 2 is just Δ_2.

Theorem 1. `GSE` *with the linear model (Eq.* (1)*) and any valid*[2] *allocation strategy* Π_t *identifies the optimal arm with probability at least* $1 - \delta$ *for*

$$\delta \leq 2\eta \log(K) \exp\left(\frac{-\Delta_{\min}^2 \sigma^{-2}}{4 \max\limits_{i \in \mathcal{A}, t \in [s]} ||x_i - x_1||_{V_t^{-1}}^2} \right). \tag{2}$$

where $||x||_V = \sqrt{x^\top V x}$ *for any* $x \in \mathbb{R}^d$ *and matrix* $V \in \mathbb{R}^{d \times d}$*. If we use the G-optimal design (Algorithm 2) for* Π_t*, then*

$$\delta \leq 2\eta \log(K) \exp\left(\frac{-B \Delta_{\min}^2}{4\sigma^2 d \log(K)} \right). \tag{3}$$

We sketch the proof in Section 4.2 and defer the detailed proof to Appendix A.

The error bound in (3) scales as expected. Specifically, it is tighter for a larger budget B, which increases the statistical power of `GSE`; and a larger gap $\Delta_{\min}$, which makes the

[1] The projection can be done by multiplying the arm features with the matrix whose columns are the orthonormal basis of the subspace spanned by the arms [Yang and Tan, 2021].

[2] Allocation strategy Π_t is valid if V_t is invertible.

optimal arm easier to identify. The bound is looser for larger K and d, which increase with the instance size; and larger reward noise σ, which increases uncertainty and makes the problem instance harder to identify. We compare this bound to the related works in Section 7.

There is no lower bound for FB BAI in structured bandits. Nevertheless, in the special case of MABs, our bound ((3)) matches the FB BAI lower bound $\exp\left(\frac{-B}{\sum_{i\in\mathcal{A}}\Delta_i^{-2}}\right)$ in Kaufmann *et al.* [2016], up to a factor of $\log K$. It also roughly matches the tight lower bound of Carpentier and Locatelli [2016], which is $\exp\left(\frac{-B}{\log(K)\sum_{i\in\mathcal{A}}\Delta_i^{-2}}\right)$. To see this, note that $\sum_{i\in\mathcal{A}}\Delta_i^{-2}\approx K\Delta_{\min}^{-2}$ and $d=K$, when we apply `GSE` to a K-armed bandit problem.

4.2 Proof Sketch

The key idea in analyzing `GSE` is to control the probability of eliminating the optimal arm in each stage. Our analysis is modular and easy to extend to other elimination algorithms. Let E_t be the event that the optimal arm is eliminated in stage t. Then, $\delta = \Pr(\cup_{t=1}^{s} E_t) \leq \sum_{t=1}^{s}\Pr(E_t|\bar{E}_1,\ldots,\bar{E}_{t-1})$, where $\bar{E}_t$ is the complement of event E_t. In Lemma 1, we bound the probability that a suboptimal arm has a higher estimated mean reward than the optimal arm. This is a novel concentration result for linear bandits in successive elimination algorithms.

Lemma 1. *In* `GSE` *with the linear model of Eq.* (1)*, the probability that any suboptimal arm i has a higher estimated mean reward than the optimal arm in stage t satisfies* $\Pr(\hat{\mu}_{i,t} > \hat{\mu}_{1,t}) \leq 2\exp\left(\frac{-\Delta_i^2\sigma^{-2}}{2||x_i-x_1||^2_{V_t^{-1}}}\right)$.

This lemma is proved using an argument mainly driven from a concentration bound. Next, we use it in Lemma 2 to bound the probability that the optimal arm is eliminated in stage t.

Lemma 2. *In* `GSE` *with the linear model (Eq.* (1)*), the probability that the optimal arm is eliminated in stage t satisfies* $\Pr(\tilde{E}_t) \leq 2\eta\exp\left(\frac{-\Delta_{\min,t}^2\sigma^{-2}}{2\max_{i\in\mathcal{A}_t}||x_i-x_1||^2_{V_t^{-1}}}\right)$, *where* $\Delta_{\min,t} = \min_{i\in\mathcal{A}_t\setminus\{1\}}\Delta_i$ *and $\tilde{E}_t$ is a shorthand for event* $E_t|\bar{E}_1,\ldots,\bar{E}_{t-1}$.

This lemma is proved by examining how another arm can dominate the optimal arm and using Markov's inequality. Finally, we bound δ in Theorem 1 using a union bound. We obtain the second bound in Theorem 1 by the Kiefer-Wolfowitz Theorem [Kiefer and Wolfowitz, 1960] for the G-optimal design described in Section 6.

5 Generalized Linear Model

We now study FB BAI in *generalized linear models (GLMs)* [McCullagh and Nelder, 1989], where $\mu_i = f(x_i) = h(x_i^\top\theta_*)$, where h is a monotone function known as the *mean function*. As an example, $h(x) = (1+\exp(-x))^{-1}$ in logistic regression. We assume that the derivative of the mean function, h', is bounded from below, i.e., $c_{\min} \leq h'(x_i^\top\tilde{\theta}_t)$, for some $c_{\min}\in\mathbb{R}^+$ and all $i\in\mathcal{A}$. Here $\tilde{\theta}_t$ can be any convex combination of θ_* and its *maximum likelihood estimate* $\hat{\theta}_t$ in stage t. This assumption is standard in GLM bandits [Filippi *et al.*, 2010; Li *et al.*, 2017]. The existence of $c_{\min}$ can be guaranteed by performing forced exploration at the beginning of each stage with the sampling cost of $O(d)$ [Kveton *et al.*, 2020]. As $\hat{\theta}_t$ satisfies $\sum_{j=1}^{n}\left(Y_{j,t}-h(X_{j,t}^\top\hat{\theta}_t)\right)X_{j,t}=0$, it can be computed efficiently by *iteratively reweighted least squares* [Wolke and Schwetlick, 1988]. This gives us the following mean estimate for each arm $i\in\mathcal{A}_t$,

$$\hat{\mu}_{i,t} = h(x_i^\top\hat{\theta}_t)\,. \tag{4}$$

5.1 Analysis

In Theorem 2, we prove similar bounds to the linear model. The proof and its sketch are presented in Appendix B. These are the first BAI error bounds for GLM bandits.

Theorem 2. `GSE` *with the GLM (Eq.* (4)*) and any valid Π_t identifies the optimal arm with probability at least $1-\delta$ for*

$$\delta \leq 2\eta\log(K)\exp\left(\frac{-\Delta_{\min}^2\sigma^{-2}c_{\min}^2}{8\max\limits_{i\in\mathcal{A},t\in[s]}||x_i||^2_{V_t^{-1}}}\right). \tag{5}$$

If we use the G-optimal design (Algorithm 2) for Π_t, then

$$\delta \leq 2\eta\log(K)\exp\left(\frac{-B\Delta_{\min}^2c_{\min}^2}{8\sigma^2 d\log(K)}\right). \tag{6}$$

The error bounds in Theorem 2 are similar to those in the linear model (Section 4.1), since $\max_{i\in\mathcal{A},t\in[s]}||x_i-x_1||_{V_t^{-1}} \leq 2\max_{i\in\mathcal{A},t\in[s]}||x_i||_{V_t^{-1}}$ .The only major difference is in factor $c_{\min}^2$, which is 1 in the linear case. This factor arises because GLM is a linear model transformed through some non-linear mean function h. When $c_{\min}$ is small, h can have flat regions, which makes the optimal arm harder to identify. Therefore, our GLM bounds become looser as $c_{\min}$ decreases. Note that the bounds in Theorem 2 depend on all other quantities same as the bounds in Theorem 1 do.

The novelty in our GLM analysis is in how we control the estimation error of θ_* using our assumptions on the existence of $c_{\min}$. The rest of the proof follows similar steps to those in Section 4.2 and are postponed to Appendix B.

6 G-Optimal Allocation

The stochastic error bounds in (2) and (5) can be optimized by minimizing $2\max_{i\in\mathcal{A},t\in[s]}||x_i||_{V_t^{-1}}$ with respect to V_t, in particular, with respect to X_t. In each stage t, let $g_t(\pi,x_i) = ||x_i||^2_{V_t^{-1}}$, where $V_t = n\sum_{i\in\mathcal{A}_t}\pi_i x_i x_i^\top$ and $\sum_{i\in\mathcal{A}_t}\pi_i = 1$. Then, optimization of V_t is equivalent to solving $\min_\pi\max_{i\in\mathcal{A}_t} g_t(\pi,x_i)$. This leads us to the G-optimal design [Kiefer and Wolfowitz, 1960], which minimizes the maximum variance along all x_i.

We develop an algorithm based on the Frank-Wolfe (FW) method [Jaggi, 2013] to find the G-optimal design. Algorithm 2 contains the pseudo-code of it, which we refer to as `FWG`. The G-optimal design is a convex relaxation of

Algorithm 2 Frank-Wolfe G-optimal allocation (FWG)

```
Input: Stage budget $n$, $N$ number of iterations
Initialization: $\pi_0 \leftarrow (1, \dots, 1)/|\mathcal{A}_t| \in \mathbb{R}^{|\mathcal{A}_t|}$, $i \leftarrow 0$
while $i < N$ do
    $\pi'_i \leftarrow \arg\min_{\pi' : ||\pi'||_1 = 1} \nabla_\pi g_t(\pi_i)^\top \pi'$   {Surrogate}
    $\gamma_i \leftarrow \arg\min_{\gamma \in [0,1]} g_t(\pi_i + \gamma(\pi'_i - \pi_i))$   {Line search}
    $\pi_{i+1} \leftarrow \pi_i + \gamma_i(\pi'_i - \pi_i)$   {Gradient step}
    $i \leftarrow i + 1$
end while
Output: $\Pi_t = \mathrm{ROUND}(n, \pi_N)$   {Rounding}
```

the G-optimal allocation; an allocation is the (integer) number of samples per arm while a design is the proportion of n for each arm. Defining $g_t(\pi) = \max_{i \in \mathcal{A}_t} g_t(\pi, x_i)$, by Danskin's theorem [Danskin, 1966], we know $\nabla_{\pi_j} g_t(\pi) = -n(x_j^\top V_t^{-1} x_{\max})^2$, where $x_{\max} = \arg\max_{i \in \mathcal{A}_t} g_t(\pi, x_i)$. This gives us the derivative of the objective function so we can use it in a FW algorithm. In each iteration, FWG first minimizes the 1st-order surrogate of the objective, and then uses line search to find the best step-size and takes a gradient step. After N iterations, it extracts an allocation (integral solution) from π_N using an efficient rounding procedure from Allen-Zhu *et al.* [2017], which we call it $\mathrm{ROUND}(n, \pi)$. This procedure takes budget n, design π_N, and returns an allocation Π_t.

In Appendix C, we show that the error bounds of Theorems 1 and 2 still hold for large enough N, if we use Algorithm 2 to obtain the allocation strategy Π_t at the exploration step (Line 3 of Algorithm 1). This results in the deterministic bounds in (3) and (6) in these theorems.

7 Related Work

To the best of our knowledge, there is no prior work on FB BAI for GLMs and our results are the first in this setting. However, there are three related algorithms for FB BAI in linear bandits that we discuss them in detail here. Before we start, note that there is no matching upper and lower bound for FB BAI in any setting [Carpentier and Locatelli, 2016]. However, in MABs, it is known that *successive elimination* is near-optimal [Carpentier and Locatelli, 2016].

BayesGap [Hoffman *et al.*, 2014] is a Bayesian version of the gap-based exploration algorithm in Gabillon *et al.* [2012]. This algorithm models correlations of rewards using a Gaussian process. As pointed out by Xu *et al.* [2018], BayesGap does not explore enough and thus performs poorly. In Appendix D.1, we show under few simplifying assumptions that the error probability of BayesGap is at most $KB \exp\left(\frac{-B\Delta^2_{\min}}{32K}\right)$. Our error bound in Eq. (3) is at most $2\eta \log(K) \exp\left(\frac{-B\Delta^2_{\min}}{4d\log(K)}\right)$. Thus, it improves upon BayesGap by reducing dependence on the number of arms K, from linear to logarithmic; and on budget B, from linear to constant. We provide a more detailed comparison of these bounds in Appendix D.1. Our experimental results in Section 8 support these observations and show that our algorithm always outperforms BayesGap in the linear setting.

Peace [Katz-Samuels *et al.*, 2020] is mainly a FC BAI algorithm based on a transductive design, which is modified to be used in the FB setting. It minimizes the *Gaussian-width* of the remaining arms with a progressively finer level of granularity. However, Peace cannot be implemented exactly because the Gaussian width does not have a closed form and is computationally expensive to minimize. To address this, Katz-Samuels *et al.* [2020] proposed an approximation to Peace, which still has some computational issues (see Remark D.2 and Section 8.1). The error bound for Peace, although is competitive, only holds for a relatively large budget (Theorem 7 in [Katz-Samuels *et al.*, 2020]). We discuss this further in Remark D.1. Although the comparison of their bound to ours is not straightforward, we show in Appendix D.2 that each bound can be superior in certain regimes that depend mainly on the relation of d and K. In particular, we show two cases: (i) Based on few claims in Katz-Samuels *et al.* [2020] that are not rigorously proved (see (i) in Appendix D.2 for more details), their error bound is at most $2\lceil\log(d)\rceil \exp\left(\frac{-B\Delta^2_{\min}}{\max_{i \in \mathcal{A}} ||x_i - x_1||_{V^{-1}} \log(d)}\right)$ which is better than our bound (Eq. (2)) only if $K > \exp(\exp(\log(d)\log\log(d)))$. (ii) We can also show that their bound is at most $2\lceil\log(d)\rceil \exp\left(\frac{-B\Delta^2_{\min}}{d\log(K)\log(d)}\right)$ under the G-optimal design, which is worse than our error bound (Eq. (3)).

In our experiments with Peace in Section 8, we implemented its approximation and it never performed better than our algorithm. We also show in Section 8.1 that approximate Peace is much more computationally expensive compared to our algorithm.

OD-LinBAI [Yang and Tan, 2021] uses a G-optimal design in a sequential elimination framework for FB BAI. In the first stage, it eliminates all the arms except $\lceil d/2 \rceil$. This makes the algorithm prone to eliminating the optimal arm in the first stage, especially when the number of arms is larger than d. It also adds a linear (in K) factor to the error bound. In Appendix D.3, we provide a detailed comparison between the error bound of OD-LinBAI and ours, and show that similar to the comparison with Peace, there are regimes where each bound is superior. However, we show that our bound is tighter in the more practically relevant setting of $K = \Omega(d^2)$. In particular, we show that their error is at most $\left(\frac{4K}{d} + 3\log(d)\right) \exp\left(\frac{(d^2 - B)\Delta^2_{\min}}{32d\log(d)}\right)$. Now assuming $K = d^q$ for some $q \in \mathbb{R}$, if we divide our bound (Eq. (3)) with theirs, we obtain $O\left(\frac{q\log(d)}{d^{q-1} + \log(d)} \exp\left(\frac{-d^2\Delta^2_{\min}}{d\log(d)}\right)\right)$, which is less than 1, so in this case *our error bound is tighter*. However, for $K < d(d+1)/2$, their bound is tighter. Finally, we note that our experiments in Section 8 and Appendix E.3 support these observations.

8 Experiments

In this section, we compare GSE to several baselines including all linear FB BAI algorithms: Peace, BayesGap, and OD-LinBAI. Others are variants of cumulative regret (CR) bandits and FC BAI algorithms. For CR algorithms, the baseline

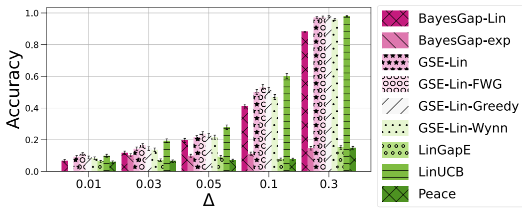

Figure 1: Static allocation.

stops at the budget limit and returns the most pulled arm.[3] We use LinUCB [Li *et al.*, 2010] and UCB-GLM [Li *et al.*, 2017], which are the state-of-the-art for linear and GLM bandits, respectively. LinGapE (a FC BAI algorithm) [Xu *et al.*, 2018] is used with its stopping rule at the budget limit. We tune its δ using a grid search and only report the best result. In Appendix F, we derive proper error bounds for these baselines to further justify the variants.

The *accuracy* is an estimate of $1-\delta$, as the fraction of 1000 Monte Carlo replications where the algorithm finds the optimal arm. We run GSE with linear model and uniform exploration (GSE-Lin), with FWG (GSE-Lin-FWG), with sequential G-optimal allocation of Soare *et al.* [2014] (GSE-Lin-Greedy), and with Wynn's G-optimal method (GSE-Lin-Wynn). For Wynn's method, see Fedorov [1972]. We set $\eta = 2$ in all experiments, as this value tends to perform well in successive elimination [Karnin *et al.*, 2013]. For LinGapE, we evaluate the Greedy version (LinGapE-Greedy) and show its results only if it outperforms LinGapE. For LinGapE-Greedy, see [Xu *et al.*, 2018]. In each experiment, we fix K, B/K, or d; depending on the experiment to show the desired trend. Similar trends can be observed if we fix the other parameters and change these. For further detail of our choices of kernels for BayesGap and also our real-world data experiments, see Appendix E.

8.1 Linear Experiment: Adaptive Allocation

We start with the example in Soare *et al.* [2014], where the arms are the canonical d-dimensional basis $e_1, e_2, \ldots, e_d$ plus a disturbing arm $x_{d+1} = (\cos(\omega), \sin(\omega), 0, \ldots, 0)^\top$ with $\omega = 1/10$. We set $\theta_* = e_1$ and $\epsilon \sim \mathcal{N}(0, 10)$. Clearly the optimal arm is e_1, however, when the angle ω is as small as 1/10, the disturbing arm is hard to distinguish from e_1. As argued in Soare *et al.* [2014], this is a setting where an adaptive strategy is optimal (see Appendix G.1 for further discussion on Adaptive vs. Static strategies).

Fig. 2 shows that GSE-Lin-FWG is the second-best algorithm for smaller K and the best for larger K. BayesGap-Lin performs poorly here, and thus, we omit it. We *conjecture* that BayesGap-Lin fails because it uses Gaussian processes and there is a very low correlation between the arms in this experiment. LinGapE wins mostly for smaller K and loses for larger K. This could be because its regret is linear in K (Appendix D). Peace has lower accuracy than several other algorithms. We could only simulate Peace for $K \leq 16$, since its computational cost is high for larger values of K. For instance, at $K = 16$, Peace completes 100 runs in 530 seconds; while it only takes 7 to 18 seconds for the other algorithms. At $K = 32$, Peace completes 100 runs in 14 hours (see Appendix E.1).

[3] In Appendix D, we argue that this is a reasonable stopping rule.

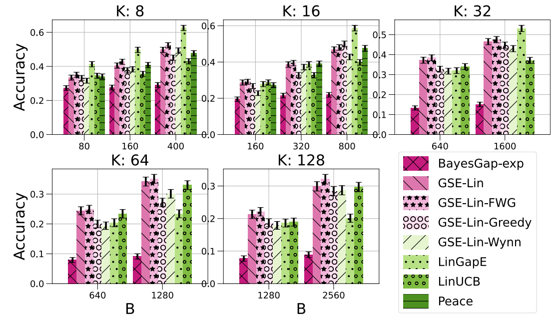

Figure 2: Adaptive instance for $d = K - 1$.

In this experiment, $K \approx d$ and both OD-LinBAI and GSE have $\log(K)$ stages and perform similarly. Therefore, we only report the results for GSE. This also happens in Section 8.2.

8.2 Linear Experiment: Static Allocation

As in Xu *et al.* [2018], we take arms $e_1, e_2, ..., e_{16}$ and $\theta_* = (\Delta, 0, \ldots, 0)$, where $K = d = 16$ and $B = 320$. In this experiment, knowing the rewards does not change the allocation strategy. Therefore, a static allocation is optimal [Xu *et al.*, 2018]. The goal is to evaluate the ability of the algorithm to adapt to a static situation.

Our results are reported in Fig. 1. We observe that LinUCB performs the best when Δ is small (harder instances). This is expected since suboptimal arms are well away from the optimal one, and CR algorithms do well in this case (Appendix D). Our algorithms are the second-best when Δ is sufficiently large, converging to the optimal static allocation. BayesGap-exp, LinGapE, and Peace cannot take advantage of larger Δ, probably because they adapt to the rewards too early. This example demonstrates how well our algorithms adjust to a static allocation, and thus, properly address the tradeoff between static and adaptive allocation.

8.3 Linear Experiment: Randomized

In this experiment, we use the example in Tao *et al.* [2018] and [Yang and Tan, 2021]. For each bandit instance, we generate i.i.d. arms sampled from the unit sphere centered at the origin with $d = 10$. We let $\theta_* = x_i + 0.01(x_j - x_i)$, where x_i and x_j are the two closest arms. As a consequence, x_i is the optimal arm and x_j is the disturbing arm. The goal is to evaluate the expected performance of the algorithms for a random instance to avoid bias in choosing the bandit instances.

We fix B/K in Fig. 3 and compare the performance for different K. GSE-Lin-FWG has competitive performance with other algorithms. We can see that G-optimal policies have similar expected performance while FWG is slightly better. Again, LinGapE performance degrades as K increases and Peace underperforms our algorithms. Moreover, the performance of OD-LinBAI worsens as K increases, especially for $K > \frac{d(d+1)}{2}$. We report more experiments in this setting, comparing GSE to OD-LinBAI, in Appendix E.3.

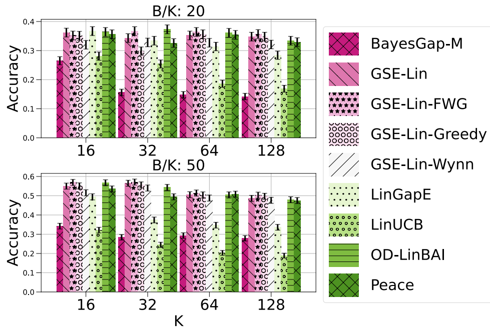

Figure 3: Randomized linear experiment.

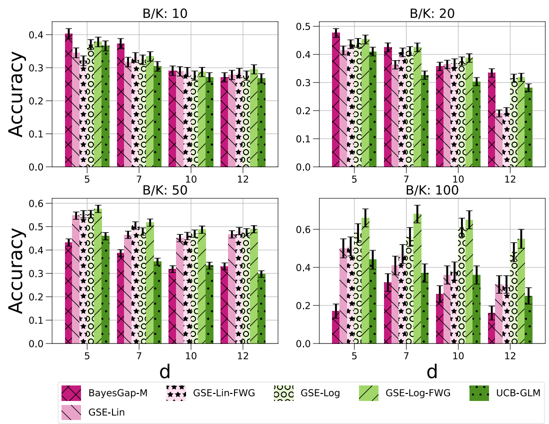

Figure 4: Logistic bandit experiment for $K = 8$.

8.4 GLM Experiment

As an instance of GLM, we study a *logistic bandit*. We generate i.i.d. arms from uniform distribution on $[-0.5, 0.5]^d$ with $d \in \{5, 7, 10, 12\}$, $K = 8$, and $\theta_* \sim N(0, \frac{3}{d} I_d)$, where I_d is a $d \times d$ identity matrix. The reward of arm i is defined as $y_i \sim \text{Bern}(h(x_i^\top \theta_*))$, where $h(z) = (1 + \exp(-z))^{-1}$ and $\text{Bern}(z)$ is a Bernoulli distribution with mean z. We use GSE with a logistic regression model (GSE-Log) and also with the linear models to evaluate the robustness of GSE to *model misspecification*. For exploration, we only use FWG (GSE-Log-FWG), as it performs better than the other G-optimal allocations in earlier experiments. We also use a modification of UCB-GLM [Li *et al.*, 2017], a state-of-the-art GLM CR algorithm, for FB BAI.

The results in Fig. 4 show GSE with logistic models outperforms linear models, and FWG improves on uniform exploration in the GLM case. These experiments also show the robustness of GSE to model misspecification, since the linear model only slightly underperforms the logistic model. UCB-GLM results confirm that CR algorithms could fail in BAI. BayesGap-M falls short for $B/K \geq 50$; the extra B in their error bound also suggests failure for large B. In contrast, the performance of GSE keeps improving as B increases.

9 Conclusions

In this paper, we studied fixed-budget best-arm identification (BAI) in linear and generalized linear models. We proposed the GSE algorithm, which offers an adaptive framework for structured BAI. Our performance guarantees are near-optimal in MABs. In generalized linear models, our algorithm is the first practical fixed-budget BAI algorithm with analysis. Our experiments show the efficiency and robustness (to model misspecification) of our algorithm. Extending our GSE algorithm to more general models could be a future direction (see Appendix H).

References

[Abbasi-yadkori *et al.*, 2011] Yasin. Abbasi-yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.

[Allen-Zhu *et al.*, 2017] Zeyuan Allen-Zhu, Yuanzhi Li, Aarti Singh, and Yining Wang. Near-optimal design of experiments via regret minimization. In *International Conference on Machine Learning*, pages 126–135, 2017.

[Audibert *et al.*, 2010] Jean-Yves Audibert, Sébastien Bubeck, and Rémi Munos. Best Arm Identification in Multi-Armed Bandits. In *Proceedings of the 23th Conference on Learning Theory*, 2010.

[Berthet and Perchet, 2017] Quentin Berthet and Vianney Perchet. Fast rates for bandit optimization with upper-confidence frank-wolfe. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[Bubeck *et al.*, 2009] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in multi-armed bandits problems. In *International conference on Algorithmic learning theory*, pages 23–37, 2009.

[Carpentier and Locatelli, 2016] Alexandra Carpentier and Andrea Locatelli. Tight (lower) bounds for the fixed budget best arm identification bandit problem. In *Conference on Learning Theory*, pages 590–604, 2016.

[Damla Ahipasaoglu *et al.*, 2008] Selin Damla Ahipasaoglu, Peng Sun, and Michael J Todd. Linear convergence of a modified Frank–Wolfe algorithm for computing minimum-volume enclosing ellipsoids. *Optimization Methods and Software*, 23(1):5–19, 2008.

[Danskin, 1966] John M. Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

[Degenne *et al.*, 2019] Rémy Degenne, Wouter M Koolen, and Pierre Ménard. Non-asymptotic pure exploration by solving games. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[Degenne *et al.*, 2020] 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, 2020.

[Fedorov, 1972] Valerii Vadimovich Fedorov. *Theory of Optimal Experiments*. Probability and Mathematical Statistics. Academic Press, 1972.

[Filippi *et al.*, 2010] Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, 2010.

[Gabillon *et al.*, 2012] Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Advances in Neural Information Processing Systems*, pages 3212–3220, 2012.

[Hoffman *et al.*, 2014] Matthew Hoffman, Bobak Shahriari, and Nando Freitas. On correlation and budget constraints in model-based bandit optimization with application to automatic machine learning. In *Artificial Intelligence and Statistics*, pages 365–374, 2014.

[Huang *et al.*, 2021] Ruiquan Huang, Weiqiang Wu, Jing Yang, and Cong Shen. Federated linear contextual bandits. *Advances in Neural Information Processing Systems*, 34, 2021.

[Jaggi, 2013] Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pages 427–435, 2013.

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

[Karnin *et al.*, 2013] 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*, page 1238–1246, 2013.

[Katz-Samuels *et al.*, 2020] 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. In *Advances in Neural Information Processing Systems*, pages 10371–10382, 2020.

[Kaufmann *et al.*, 2016] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *The Journal of Machine Learning Research*, 17(1):1–42, 2016.

[Khachiyan, 1996] Leonid G Khachiyan. Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research*, 21(2):307–320, 1996.

[Kiefer and Wolfowitz, 1960] Jack Kiefer and Jacob Wolfowitz. The equivalence of two extremum problems. *Canadian Journal of Mathematics*, 12:363–366, 1960.

[Kumar and Yildirim, 2005] Piyush Kumar and E Alper Yildirim. Minimum-volume enclosing ellipsoids and core sets. *Journal of Optimization Theory and Applications*, 126(1):1–21, 2005.

[Kveton *et al.*, 2020] Branislav Kveton, Manzil Zaheer, Csaba Szepesvari, Lihong Li, Mohammad Ghavamzadeh, and Craig Boutilier. Randomized exploration in generalized linear bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 2066–2076, 2020.

[Lattimore and Szepesvári, 2020] Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

[Li *et al.*, 2010] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. *International Conference on World Wide Web*, 2010.

[Li *et al.*, 2017] Lihong Li, Yu Lu, and Dengyong Zhou. Provably optimal algorithms for generalized linear contextual bandits. In *International Conference on Machine Learning*, pages 2071–2080, 2017.

[McCullagh and Nelder, 1989] Peter McCullagh and John A Nelder. *Generalized Linear Models*. Chapman & Hall, 1989.

[Riquelme *et al.*, 2018] Carlos Riquelme, George Tucker, and Jasper Snoek. Deep bayesian bandits showdown: An empirical comparison of bayesian deep networks for Thompson sampling. In *International Conference on Learning Representations*, 2018.

[Soare *et al.*, 2014] Marta Soare, Alessandro Lazaric, and Rémi Munos. Best-arm identification in linear bandits. In *Advances in Neural Information Processing Systems*, pages 828–836, 2014.

[Tao *et al.*, 2018] Chao Tao, Saúl Blanco, and Yuan Zhou. Best arm identification in linear bandits with linear dimension dependency. In *International Conference on Machine Learning*, pages 4877–4886, 2018.

[Tripuraneni *et al.*, 2021] Nilesh Tripuraneni, Chi Jin, and Michael Jordan. Provable meta-learning of linear representations. In *International Conference on Machine Learning*, pages 10434–10443, 2021.

[Vershynin, 2019] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics, 2019.

[Wolke and Schwetlick, 1988] Robert. Wolke and Hartmut Schwetlick. Iteratively reweighted least squares: algorithms, convergence analysis, and numerical comparisons. *SIAM Journal on Scientific and Statistical Computing*, 9(5):907–921, 1988.

[Xu *et al.*, 2018] 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, 2018.

[Yang and Tan, 2021] Junwen Yang and Vincent YF Tan. Towards minimax optimal best arm identification in linear bandits. *arXiv preprint arXiv:2105.13017*, 2021.

A Linear Model Proofs

We let $K = \eta^l$ for some integer $l \geq 2$ so $s = \log(K) = l$. This is for ease of reading in the analysis and proofs. We can deal with the cases in which l is not an integer using rounding operators.

Proof of Lemma 1. Fix stage t. Since t is fixed, we drop it in the rest of the proof. We start with

$$\Pr(\hat{\mu}_i > \hat{\mu}_1) = \Pr(x_i^\top \hat{\theta} > x_1^\top \hat{\theta}) \tag{7}$$

$$\begin{aligned} &= \Pr((x_i - x_1)^\top \hat{\theta} > 0) \\ &= \Pr((x_i - x_1)^\top V^{-1} \sum_{j=1}^n X_j Y_j > 0) \\ &= \Pr((x_i - x_1)^\top V^{-1} (V^\top \theta_* + \sum_{j=1}^n X_j \epsilon_j) > 0) \\ &= \Pr((x_i - x_1)^\top V^{-1} \sum_{j=1}^n X_j \epsilon_j > (x_1 - x_i)^\top \theta_*) \ . \end{aligned} \tag{8}$$

Now since $\{\epsilon_j\}_{j\in[n]}$ are independent, mean zero, σ^2-sub-Gaussian random variables, if we define

$$\begin{aligned} W_i &= (W_{i,1}, \ldots, W_{i,n}) \\ &= (x_i - x_1)^\top V^{-1} \Big(X_1, \cdots, X_n \Big) \in \mathbb{R}^n \ , \end{aligned}$$

then by Hoeffding's inequality (Theorem 2.6.3 in [Vershynin, 2019]) we can write and bound Eq. (8) as follows

$$\begin{aligned} \Pr(\sum_{j=1}^n W_{i,j} \epsilon_j > (x_1 - x_i)^\top \theta_*) &\leq 2 \exp\left(\frac{-((x_1 - x_i)^\top \theta_*)^2}{2\sigma^2 \, ||W_i||_2^2} \right) \\ &= 2 \exp\left(\frac{-\Delta_i^2}{2\sigma^2 \, ||W_i||_2^2} \right) \ . \end{aligned} \tag{9}$$

It is important that the noise and features are independent. This is how we do not need adaptivity in each stage. Now since

$$\begin{aligned} ||W_i||_2^2 = W_i W_i^\top &= (x_i - x_1)^\top V^{-1} V V^{-1} (x_i - x_1) \\ &= (x_i - x_1)^\top V^{-1} (x_i - x_1) \ . \end{aligned}$$

We can bound Eq. (9) as follows

$$2 \exp\left(\frac{-\Delta_i^2}{2\sigma^2 \, ||W_i||_2^2} \right) \leq 2 \exp\left(\frac{-\Delta_i^2}{2\sigma^2 \, ||x_i - x_1||_{V^{-1}}^2} \right) \ .$$

□

Proof of Lemma 2. Here we want to bound $\Pr(\tilde{E}_t)$. Let N_t denote the number of arms in $\mathcal{A}_t$ whose $\hat{\mu}$ is larger than $\hat{\mu}_1$. Then by Lemma 1, we have

$$\begin{aligned} \mathbf{E}(N_t) &= \sum_{i \in \mathcal{A}_t} \Pr(\hat{\mu}_{i,t} > \hat{\mu}_{1,t}) \\ &\leq \sum_{i \in \mathcal{A}_t} 2 \exp\left(\frac{-\Delta_i^2}{2\sigma^2 \, ||x_i - x_1||_{V_t^{-1}}^2} \right) \\ &\leq \sum_{i \in \mathcal{A}_t} 2 \exp\left(\frac{-\Delta_{\min,t}^2}{2\sigma^2 \, ||x_i - x_1||_{V_t^{-1}}^2} \right) \\ &\leq 2|\mathcal{A}_t| \exp\left(\frac{-\Delta_{\min,t}^2}{2\sigma^2 \max_{i\in\mathcal{A}_t} ||x_i - x_1||_{V_t^{-1}}^2} \right) \ . \end{aligned}$$

Now Markov inequality gives

$$\begin{aligned} \Pr(\tilde{E}_t) &= \Pr(N_t \geq \frac{1}{\eta} |\mathcal{A}_t|) \\ &\leq \frac{\eta \mathbf{E}(N_t)}{|\mathcal{A}_t|} \\ &\leq 2\eta \exp\left(\frac{-\Delta_{\min,t}^2}{2\sigma^2 \max_{i\in\mathcal{A}_t} ||x_i - x_1||_{V_t^{-1}}^2} \right) \ . \end{aligned}$$

□

Proof of Theorem 1. By Lemma 2 the optimal arm is eliminated in one of the $s = \log(K)$ stages with probability at most

$$\begin{aligned} \delta &\leq \sum_{t=1}^s \Pr(\tilde{E}_t) \\ &\leq 2 \sum_{t=1}^s \eta \exp\left(\frac{-\Delta_{\min,t}^2}{2\sigma^2 \max_{i\in\mathcal{A}_t} ||x_i - x_1||_{V_t^{-1}}^2} \right) \\ &\leq 2\eta \log(K) \exp\left(\frac{-\Delta_{\min}^2}{2\sigma^2 \max_{i\in\mathcal{A}_t, t\in[s]} ||x_i - x_1||_{V_t^{-1}}^2} \right) \\ &\leq 2\eta \log(K) \exp\left(\frac{-\Delta_{\min}^2}{4\sigma^2 \max_{i\in\mathcal{A}_t, t\in[s]} ||x_i||_{V_t^{-1}}^2} \right) \ . \end{aligned}$$

where we used Cauchy-Schwarz and Triangle inequality in the last inequality. Now by Kiefer-Wolfowitz Theorem [Kiefer and Wolfowitz, 1960], we know that under the G-optimal or D-optimal design

$$g_t(\pi^*) = \max_{i\in\mathcal{A}_t} ||x_i||_{V_t^{-1}}^2 = d_t/n \leq \frac{d \log(K)}{B}, \quad \forall t \ .$$

Therefore, $\max_{i\in\mathcal{A}_t, t\in[s]} ||x_i||_{V_t^{-1}}^2 \leq d_t/n \leq \frac{d\log(K)}{B}$ and

$$\begin{aligned} 2\eta \log(K) &\exp\left(\frac{-\Delta_{\min}^2}{4\sigma^2 \max_{i\in\mathcal{A}, t\in[s]} ||x_i||_{V_t^{-1}}^2} \right) \\ &\leq 2\eta \log(K) \exp\left(\frac{-B\Delta_{\min}^2}{4d\sigma^2 \log(K)} \right) \ . \end{aligned} \tag{10}$$

□

B GLM Proofs

The novelty in our GLM analysis is in how we control the estimation error of θ_* using our assumptions on the existence of $c_{\min}$. The rest of the proof follows similar steps to those in the linear model. The key idea is to obtain error bounds in each stage. First, in Lemma B.1, we bound the probability that a suboptimal arm has a higher estimated mean reward than the optimal arm.

Lemma B.1. *In* GSE *with the GLM of Eq.* (4)*, the probability that any suboptimal arm i has a higher estimated mean reward than the optimal arm in stage t satisfies*

$$\Pr(\hat{\mu}_{i,t} > \hat{\mu}_{1,t}) \leq \exp\left(\frac{-\Delta_i^2\sigma^{-2}c_{\min}^2}{8\,||x_i||^2_{V_t^{-1}}}\right) + \exp\left(\frac{-\Delta_i^2\sigma^{-2}c_{\min}^2}{8\,||x_1||^2_{V_t^{-1}}}\right).$$

We prove this lemma using the assumption on h' and Hoeffding's inequality.

Proof of Lemma B.1. Since t is fixed we drop it in the rest of this proof. Let

$$\Lambda = \sum_{j=1}^n h'(X_j^\top \tilde{\theta}) X_j X_j^\top ,$$

where $\tilde{\theta}$ is some convex combination of θ_* and $\hat{\theta}$ then

$$\begin{aligned}
&\Pr(x_i^\top \hat{\theta} > x_1^\top \hat{\theta}) \\
&= \Pr(x_i^\top \hat{\theta} - x_i^\top \theta_* - \frac{\Delta_i}{2} > x_1^\top \hat{\theta} - x_1^\top \theta_* + \frac{\Delta_i}{2}) \\
&\leq \Pr(x_i^\top \hat{\theta} - x_i^\top \theta_* - \frac{\Delta_i}{2} > 0) \\
&\qquad + \Pr(x_1^\top \hat{\theta} - x_1^\top \theta_* + \frac{\Delta_i}{2} < 0) \\
&= \Pr(x_i^\top (\hat{\theta} - \theta_*) > \frac{\Delta_i}{2}) + \Pr(x_1^\top (\theta_* - \hat{\theta}) > \frac{\Delta_i}{2}) \\
&= \Pr(x_i^\top \Lambda^{-1} \sum_{j=1}^n X_j \epsilon_j > \frac{\Delta_i}{2}) \\
&\qquad + \Pr(-x_1^\top \Lambda^{-1} \sum_{j=1}^n X_j \epsilon_j > \frac{\Delta_i}{2}) .
\end{aligned}$$

Note that we use Δ_i for the gap before the mean function transformation. In the last inequality, we used Lemma 1 in [Kveton *et al.*, 2020]. Now if we define

$$W_i = (W_{i,1}, \dots, W_{i,n}) = x_i^\top \Lambda^{-1} \Big(X_1, \cdots, X_n \Big) \in \mathbb{R}^n$$
$$\Rightarrow ||W_i||_2^2 = W_i W_i^\top = x_i^\top \Lambda^{-1} V \Lambda^{-1} x_i ,$$

then by Hoeffding's inequality, we get

$$\begin{aligned}
\Pr(x_i^\top \Lambda^{-1} \sum_{j=1}^n X_j \epsilon_j > \frac{\Delta_i}{2}) &= \Pr(\sum_{j=1}^n W_{i,j} \epsilon_j > \frac{\Delta_i}{2}) \\
&\leq \exp\left(\frac{-\Delta_i^2}{8\sigma^2\,||W_i||_2^2}\right) , \\
\Pr(-x_1^\top \Lambda^{-1} \sum_{j=1}^n X_j \epsilon_j > \frac{\Delta_i}{2}) &= \Pr(-\sum_{j=1}^n W_{1,j} \epsilon_j > \frac{\Delta_i}{2}) \\
&\leq \exp\left(\frac{-\Delta_i^2}{8\sigma^2\,||W_1||_2^2}\right) .
\end{aligned}$$

Since $\tilde{\theta}$ is not known in the process, we need to dig deeper. By assumption, we know $c_{\min} \leq h'(x_i^\top \tilde{\theta})$ for some $c_{\min} \in \mathbb{R}^+$ and for all $i \in \mathcal{A}$, therefore $c_{\min}^{-1} V^{-1} \succeq \Lambda^{-1}$ by definition of Λ, and

$$\begin{aligned}
||W_i||_2^2 &= x_i^\top \Lambda^{-1} V \Lambda^{-1} x_i \\
&\preceq x_i^\top c_{\min}^{-1} V^{-1} V c_{\min}^{-1} V^{-1} x_i \\
&= c_{\min}^{-2}\, ||x_i||^2_{V^{-1}} ,
\end{aligned}$$

so

$$\begin{aligned}
&\Pr(x_i^\top \hat{\theta} > x_1^\top \hat{\theta}) \\
&\leq \exp\left(\frac{-\Delta_i^2 c_{\min}^2}{8\sigma^2\,||x_i||^2_{V^{-1}}}\right) + \exp\left(\frac{-\Delta_i^2 c_{\min}^2}{8\sigma^2\,||x_1||^2_{V^{-1}}}\right) .
\end{aligned}$$

□

Next, we bound the error probability at each stage in Lemma B.2.

Lemma B.2. *In* GSE *with the GLM of Eq.* (4)*, the probability that the optimal arm is eliminated in stage t satisfies*

$$\Pr(\tilde{E}_t) \leq 2\eta \exp\left(\frac{-\Delta_{\min,t}^2\ \sigma^{-2}\ c_{\min}^2}{8 \max_{i \in \mathcal{A}_t}\ ||x_i||^2_{V_t^{-1}}}\right) .$$

Proof of Lemma B.2. By Lemma B.1 we have

$$\begin{aligned}
\mathbf{E}(N_t) &= \sum_{i \in \mathcal{A}_t} \Pr(\hat{\mu}_{i,t} > \hat{\mu}_{1,t}) \\
&\leq \sum_{i \in \mathcal{A}_t} \exp\left(\frac{-\Delta_i^2 c_{\min}^2}{8\sigma^2\,||x_i||^2_{V_t^{-1}}}\right) + \exp\left(\frac{-\Delta_i^2 c_{\min}^2}{8\sigma^2\,||x_1||^2_{V_t^{-1}}}\right) \\
&\leq 2|\mathcal{A}_t| \exp\left(\frac{-\Delta_{\min,t}^2 c_{\min}^2}{8\sigma^2 \max_{i \in \mathcal{A}_t} ||x_i||^2_{V_t^{-1}}}\right) .
\end{aligned}$$

Now Markov inequality gives

$$\begin{aligned}
\Pr(\tilde{E}_t) &= \Pr(N_t > 1/\eta|\mathcal{A}_t|) \\
&\leq 2\eta \exp\left(\frac{-\Delta_{\min,t}^2 c_{\min}^2}{8\sigma^2 \max_{i \in \mathcal{A}_t} ||x_i||^2_{V_t^{-1}}}\right) .
\end{aligned}$$

□

Finally, we bound the probability of error and conclude the proof of Theorem 2 by using this result together with a union bound and the Kiefer-Wolfowitz Theorem.

Proof of Theorem 2. By Lemma B.2 we know that the optimal arm is eliminated in one of the $s = \log(K)$ stages with a probability that satisfies

$$\begin{aligned}\delta &\le \sum_{t=1}^{s} 2\eta \exp\left(\frac{-\Delta_{\min,t}^2 c_{\min}^2}{8\sigma^2 \max_{i\in\mathcal{A}_t} ||x_i||^2_{V_t^{-1}}}\right)\\ &\le \sum_{t=1}^{s} 2\eta \exp\left(\frac{-\Delta_{\min}^2 c_{\min}^2}{8\sigma^2 \max_{i\in\mathcal{A}_t} ||x_i||^2_{V_t^{-1}}}\right)\\ &\le 2\eta\log(K) \exp\left(\frac{-\Delta_{\min}^2 c_{\min}^2}{8\sigma^2 \max_{i\in\mathcal{A}_t, t\in[s]} ||x_i||^2_{V_t^{-1}}}\right) .\end{aligned}$$

Now by the Kiefer-Wolfowitz Theorem in [Kiefer and Wolfowitz, 1960], we know $g_t(\pi^*) = \max_{i\in\mathcal{A}_t, t\in[s]} ||x_i||^2_{V_t^{-1}} = d_t/n \le \frac{d\log(K)}{B}$ under the G-optimal or D-optimal design, so with this design we have

$$\begin{aligned}\delta &\le 2\eta\log(K) \exp\left(\frac{-\Delta_{\min}^2 c_{\min}^2}{8\sigma^2 \max_{i\in\mathcal{A}_t, t\in[s]} ||x_i||^2_{V_t^{-1}}}\right)\\ &\le 2\eta\log(K) \exp\left(\frac{-B\Delta_{\min}^2 c_{\min}^2}{8\sigma^2 d\log(K)}\right) .\end{aligned}$$

□

C Frank Wolfe G-optimal Design

In this section, we develop more details for our FW G-optimal algorithm. Let $\pi : \mathcal{A}_t \to [0, 1]$ be a distribution on $\mathcal{A}_t$, so $\sum_{i\in\mathcal{A}_t} \pi(i) = 1$. For instance, based on Kiefer and Wolfowitz [1960] (or Theorem 21.1 (Kiefer–Wolfowitz) and equation 21.1 from [Lattimore and Szepesvári, 2020]) we should sample arm i in stage t, w_i times, in which

$$w_i = \left\lceil \frac{\pi(i) g_t(\pi)}{\varepsilon^2} \log(1/\delta) \right\rceil ,$$

where $g_t(\pi) = \max_{i\in\mathcal{A}} ||x_i||_{V_t^{-1}}$ and we know that $g_t(\pi^*) = d_t$ by the same Theorem. Finding π^* is a convex problem for finite number of arms and can be solved using Frank-Wolfe algorithm (read note 3 from section 21 in [Lattimore and Szepesvári, 2020]). After we get the optimal design π^*, we can get the optimal allocation using a randomized rounding. There are algorithms that avoid randomized rounding by starting with an allocation problem (see Section 7). We develop yet another efficient algorithm for the optimal design in Section 6.

Khachiyan in Khachiyan [1996] showed that if we run the FW algorithm for $O(d \log\log(K + d))$ iterations, we get $g_t(\hat{\pi}) \le d_t$, where $\hat{\pi}$ is the FW solution. More precisely, we get the following error bounds as a corollary.

Corollary C.1. *If we use* GSE *with* FWG *for Exploration, for* $N = O(d \log\log(K + d))$ *iterations, then*

$$\delta \le 2\eta\log(K) \exp\left(\frac{-B\Delta_{\min}^2}{4d\sigma^2\log(K)}\right) .$$

Proof. As in Eq. (10) we just use $g(\hat{\pi}) \le d_t/n \le \frac{d\log_\eta K}{B}$, where $\hat{\pi}$ is derived according to the algorithm in Khachiyan [1996]. □

Also, Kumar and Yildirim [2005] suggested an initialization of the FW algorithm which achieves a bound independent of the number of arms. In particular, we get the following corollary.

Corollary C.2. *If we use the* GSE *using FW algorithm to find a G-optimal design with* $N = O(d \log\log(d))$ *iterations starting from the initialization advised in Kumar and Yildirim [2005], then the accuracy is lower bounded as below;*

$$\delta \le 2\eta\log(K) \exp\left(\frac{-B\Delta_{\min}^2}{8d\sigma^2\log(K)}\right) .$$

Proof. By Theorem 2.3 in Damla Ahipasaoglu *et al.* [2008] or note 21.2 in Lattimore and Szepesvári [2020] we get $g(\hat{\pi}) \le 2d/n$. The rest is same as in Corollary C.1. □

The same sort of Corollaries hold for GLM as well by starting from Theorem 2.

It is worth noting that based on the connection between D-optimal and G-optimal through the Kiefer-Wolfowitz Theorem [Kiefer and Wolfowitz, 1960], we can also look at the FW algorithm for a D-optimal allocation. In D-optimal design, we seek to minimize $\det(V_t^{-1})$ which is equal to minimizing $h_t(\pi) = -\det(V_t)$ and we get

$$\begin{aligned}\nabla_{\pi_i} h_t(\pi) &= -\det(V_t)\mathrm{tr}\left(V_t^{-1}\frac{\partial V_t}{\partial \pi_i}\right)\\ &= -\det(V_t)\mathrm{tr}\left(V_t^{-1} x_i x_i^\top\right) .\end{aligned}$$

We can use this in FWG algorithm to implement the D-optimal allocation. The experimental results show similar performance using both G and D optimal allocation.

D Error Bound Comparison

In this section, we compare our analytical bounds to those in the related works. We try to derive similar bounds from their performance guarantee so that it is comparable to ours.

D.1 BayesGap

Based on Theorem 1 in Hoffman *et al.* [2014] we know if $\theta_* \sim \mathcal{N}(0, \eta_b^2 I)$ then BayesGap simple regret has the following bound for any $\epsilon > 0$

$$\Pr(\mu_1 - \mu_\zeta \ge \epsilon) \le KB \exp\left(-\left(\frac{B-K}{\sigma^2} + \frac{\kappa_b}{\eta_b^2}\right)/8H_\epsilon\right),$$

where $\kappa_b = \sum_{i\in\mathcal{A}} ||x_i||^{-2}$ and $H_\epsilon = \sum_{i\in\mathcal{A}} \big(\max(0.5(\Delta_i + \epsilon), \epsilon)\big)^{-2}$. We use the following assumptions to transform the bounds in way comparable to our bounds. Namely we use $\epsilon = 0^+$, $\eta_b^2 = 1$, $\sigma^2 = 1$ where 0^+ is a very small positive number and for simplicity we assume $\max(0.5(\Delta_i + 0^+), 0^+) = 0.5\Delta_i$. We know that $\sum_{i\in\mathcal{A}} \Delta_i^{-2} \le K\Delta_{\min}^{-2}$, $\kappa_b \ge KL^{-2} =$

K where $L = 1$. In this setting, the error bound satisfy the following

$$\begin{aligned}\Pr(\mu_1 - \mu_\zeta > 0) &\simeq \Pr(\mu_1 - \mu_\zeta > 0^+) \\ &\leq KB \exp\left(\frac{-B\Delta_{\min}^2}{32K}\right) . \qquad (11)\end{aligned}$$

This bound is comparable to our bounds in Theorem 1. With the same assumptions, the error (regret) bound of `GSE` with G-optimal design is

$$\Pr(\mu_1 - \mu_\zeta \geq 0) \leq 2\eta \log(K) \exp\left(\frac{-B\Delta_{\min}^2}{4d \log(K)}\right) . \qquad (12)$$

Comparing Eq. (11) and Eq. (12) we can see the improvements that `GSE` has over `BayesGap`. In terms of K, we improve the linear dependence to log factors while in B, we improve by eliminating the linear factor to the constant outside of the exponent.

In summary, We know for `BayesGap`, $\Pr(\mu_1 \geq \mu_\zeta) \leq KB \exp\left(\frac{-B\Delta_{\min}^2}{32K}\right)$ while `GSE` satisfies $\Pr(\mu_1 \geq \mu_\zeta) \leq 2\eta \log(K) \exp\left(\frac{-B\Delta_{\min}^2}{4d \log(K)}\right)$. The improvements in its dependence on K and B are obvious.

D.2 Peace

We compare our error bound Eq. (2) (or Eq. (3)) to the performance guarantee of FB Peace in Theorem 7 of Katz-Samuels *et al.* [2020]. We consider different cases and claims as follows, however, it is not easy to compare our bounds to `Peace` bounds since their bounds are defined using quite different approaches.

(i) If we accept few claims in the Peace paper (see below), their error bound is as follows:

$$2\lceil \log(d) \rceil \exp\left(\frac{-B\Delta_{\min}^2}{c \max_{i\in\mathcal{A}} ||x_i - x_1||_{V^{-1}} \log(d)}\right) .$$

which is better than our error bound Eq. (2) if $\boxed{K > \exp(\exp(\log(d) \log\log(d)))}$, and worse otherwise.

Proof. The error bound for `Peace` in Theorem 7 of Katz-Samuels *et al.* [2020] is as follows;

$$\Pr(\zeta \neq 1) \leq 2\lceil \log(\gamma(\mathcal{X})) \rceil \exp\left(\frac{-B}{c'(\rho^* + \gamma^*) \log(\gamma(\mathcal{X}))}\right) . \qquad (13)$$

for $B \geq c \max\left((\rho^* + \gamma^*), d\right) \log(\gamma(\mathcal{X}))$ where $\mathcal{X}$ is the set of all arms $\{x_i\}_{i\in\mathcal{A}}$, c' is a constant and

$$\gamma(X) := \min_{\lambda\in\Delta} \mathbf{E}_{\xi\sim N(0,I_d)} \left(\sup_{x,x'\in X} (x - x')^\top V^{-1/2}(\lambda)\xi\right) .$$

with Δ being a probability simplex and $X \subset \mathcal{X}$ is any subset of arms, $V(\lambda) = \sum_{i\in\mathcal{A}} \lambda_i x_i x_i^\top$ and $V^{-1/2} = (V^{-1})^{1/2}$ i.e. $V^{-1/2}V^{-1/2} = V^{-1}$. Also

$$\rho^* := \inf_{\lambda\in\Delta} \rho^*(\lambda) ,$$

where

$$\rho^* := \sup_{x\in\mathcal{X}\setminus\{x_1\}} \frac{||x_1 - x||_{V(\lambda)^{-1}}^2}{(\theta_*^\top (x_1 - x))^2} , \qquad (14)$$

$$\gamma^* := \inf_{\lambda\in\Delta} \gamma^*(\lambda) ,$$

where

$$\gamma^*(\lambda) := \mathbf{E}_{\xi\sim N(0,I_d)} \left(\sup_{z\in\mathcal{Z}\setminus\{z_*\}} \frac{(z_* - z)^\top V(\lambda)^{-1/2}\xi}{\theta^\top (z_* - z)}\right)^2 .$$

By Proposition 1 in Katz-Samuels *et al.* [2020], we know $\gamma^* \geq c\rho^*$ since $\inf_{x\neq x_1} \inf_{\lambda\in\Delta} \frac{||x_1 - x||_{V(\lambda)^{-1}}}{(\theta_*^\top (x_1 - x))^2} \ll \rho^*$. This claim is not proved in Katz-Samuels *et al.* [2020] though. As such we can give advantage to `Peace` and assume the bound Eq. (13) is rather

$$\begin{aligned}&\Pr(\zeta \neq 1) \\ &\leq 2\lceil \log(\gamma(\mathcal{X})) \rceil \exp\left(\frac{-B}{c\rho^* \log(\gamma(\mathcal{X}))}\right) \\ &\leq 2\lceil \log(\gamma(\mathcal{X})) \rceil \exp\left(\frac{-B\Delta_{\min}^2 (\log(\gamma(\mathcal{X})))^{-1}}{c \max_{i\in\mathcal{A}} ||x_i - x_1||_{V^{-1}}}\right) .\end{aligned}$$

where we used Eq. (14) in the last inequality. By the claim after Theorem 7 in Katz-Samuels *et al.* [2020] $\log(\gamma(\mathcal{X})) = O(\log(d))$ for linear bandits. This claim is not proved in Katz-Samuels *et al.* [2020] neither. Now the bound is

$$2\lceil \log(d) \rceil \exp\left(\frac{-B\Delta_{\min}^2}{c \max_{i\in\mathcal{A}} ||x_i - x_1||_{V^{-1}} \log(d)}\right) .$$

Bringing all the elements to the exponent, this is of order

$$\exp\left(\frac{-B\Delta_{\min}^2}{c \max_{i\in\mathcal{A}} ||x_i - x_1||_{V^{-1}} \log(d) \log\log(d)}\right) , \qquad (15)$$

while Eq. (2) is

$$\exp\left(\frac{-B\Delta_{\min}^2}{c \max_{i\in\mathcal{A}} ||x_i - x_1||_{V^{-1}} \log\log(K)}\right) , \qquad (16)$$

as such the comparison boils down to comparing

$$\boxed{\log\log(K) \text{ for ours vs } \log(d)\log\log(d) \text{ for theirs}}$$

. Therefore, Eq. (15) is better than Eq. (16) if $K > \exp(\exp(\log(d) \log\log(d)))$ and worse otherwise. □

(ii) We can also show that their bound is

$$2\lceil \log(d) \rceil \exp\left(\frac{-B\Delta_{\min}^2}{c' d \log(K) \log(d)}\right) ,$$

under the G-optimal design, which is <u>worse than our error bound Eq. (3).</u>

Proof. On the other hand, by definition of γ we know

$$\begin{aligned}\gamma^* &\leq \inf_{\lambda\in\Delta} \mathbf{E}_{\xi\sim N(0,I)} \frac{\sup_{x\in\mathcal{X}\setminus\{x_1\}} (x_1 - x)^\top V(\lambda)^{-1/2}\xi}{\sup_{x\in\mathcal{X}} \left(\theta^\top (x_1 - x)\right)^2} \\ &= \gamma(\mathcal{X})/\Delta_{\min}^2 . \qquad (17)\end{aligned}$$

Also by Proposition 1 in Katz-Samuels *et al.* [2020] $\rho^* \leq \rho^* \log(K)$ so we can rewrite Eq. (13) as

$$\Pr(\zeta \neq 1) \leq 2\lceil \log(\gamma(\mathcal{X}))\rceil \exp\left(\frac{-B}{c'\rho^*(1+\log(K))\log(\gamma(\mathcal{X}))}\right) \leq 2\lceil\log(d)\rceil \exp\left(\frac{-B\Delta_{\min}^2}{c'\max_{i\in\mathcal{A}}||x_i - x_1||_{V^{-1}}\log(K)\log(d)}\right),$$

which under G-optimal design is

$$2\lceil\log(d)\rceil \exp\left(\frac{-B\Delta_{\min}^2}{c'd\log(K)\log(d)}\right).$$

Now we can compare this with Eq. (3) and notice the extra $\log(d)$ in the exponent of `Peace` bound. Again this can be written orderwise as

$$\exp\left(\frac{-B\Delta_{\min}^2}{c'd\log(K)\log(d)\log(\log(d))}\right).$$

which by Eq. (16) the comparison simplifies to

$$\boxed{\log\log(K) \text{ for ours vs } d\log(K)\log(d)\log(\log(d)) \text{ for theirs}}$$

which is always in the favor of our algorithm. □

Remark D.1 (Budget Requirement of `Peace`). *Similar to above, using Proposition 1 and the claims in Katz-Samuels* et al. *[2020], for the lower bound required for* `Peace` *budget we have*

$$\begin{aligned} B &\geq c\max\left([\rho^* + \gamma^*], d\right)\log(\gamma(\mathcal{Z})) \\ &\gtrsim c\left[\rho^* + \gamma^*\right]\log(d) \\ &\geq c\rho^*\log(d) \\ &= c\log(d)\frac{\max_{i\in\mathcal{A}}||x_i - x_1||_{V^{-1}}^2}{\Delta_{\min}^2}, \end{aligned}$$

wherein the second inequality we used the claim that $\log(\gamma(\mathcal{Z})) = O(\log(d))$ so we used $\gtrsim$ to account for that.

Remark D.2 (Implementational Issues of the Approximation for Peace). *We note that the "Computationally Efficient Algorithm for Combinatorial Bandits" in the Peace paper is proposed for the FC setting and it is not easy to derive it for the FB setting. Nonetheless, it still has some computational issues. To name a few, consider the calculating the gradient $\nabla_\lambda g(\lambda;\eta;\bar{z})$ in estimateGradient subroutine, Algorithm 8 which could be cumbersome as it is done B times. Also the maximization in subroutine,* $\arg\max_{z\in\mathcal{Z}} g(\lambda;\eta;\mathit{MAX\text{-}VAL};z)$ *could be hard to track since the geometric properties of $\mathcal{Z}$ come into play.*

D.3 OD-LinBAI

Here we compare the error upper bounds of our algorithm with OD-LinBAI. We show that their bound could simplify to

$$\Pr(\zeta \neq 1) \leq \left(\frac{4K}{d} + 3\log(d)\right)\exp\left(\frac{(d^2 - B)\Delta_{\min}^2}{32d\log(d)}\right).$$

Now assume $K = d^q$ for some $q \in \mathbb{R}$, if we divide our bound Eq. (3) with theirs we get

$$O\left(\frac{q\log(d)}{d^{q-1} + \log(d)}\exp\left(\frac{-d^2\Delta_{\min}^2}{d\log(d)}\right)\right). \tag{18}$$

which is less than 1 and in this case **our error bound is tighter**. Nevertheless, in the case of $K < d(d+1)/2$, their bound is tighter.

Proof. Our error bound for the linear case is (with $\eta = 2$ and $\sigma^2 = 1$)

$$\Pr(\zeta \neq 1) \leq 4\log(K)\exp\left(\frac{-B\Delta_{\min}^2}{4d\log(K)}\right),$$

while the error bound of OD-LinBAI in theorem 2 is

$$P(\zeta \neq 1) \leq \left(\frac{4K}{d} + 3\log(d)\right)\exp\left(\frac{-m}{32H_{2,\text{lin}}}\right),$$

where $H_{2,\text{lin}} = \max_{2\leq i\leq d}\frac{i}{\Delta_i^2}$ and

$$m = \frac{B - \min\left(K, \frac{d(d+1)}{2}\right) - \sum_{r=1}^{\lceil\log(d)\rceil - 1}\left[\frac{d}{2^r}\right]}{\lceil\log(d)\rceil}$$

$$\leq \begin{cases} \frac{B - d(d+1)/2}{\log(d)} \leq \frac{B - d^2}{\log(d)} & K \geq d(d+1)/2 \\ \frac{B-K}{\log(d)} & K < d(d+1)/2 \end{cases}.$$

Thus their bound is

$$\begin{aligned} &\Pr(\zeta \neq 1) \\ &\leq \left(\frac{4K}{d} + 3\log(d)\right)\exp\left(\frac{-m}{32H_{2,\text{lin}}}\right) \\ &\geq \begin{cases} \left(\frac{4K}{d} + 3\log(d)\right)\exp\left(\frac{d^2 - B}{32H_{2,\text{lin}}\log(d)}\right) & K \geq d(d+1)/2 \\ \left(\frac{4K}{d} + 3\log(d)\right)\exp\left(\frac{B-K}{32H_{2,\text{lin}}\log(d)}\right) & K < d(d+1)/2 \end{cases}. \end{aligned}$$

In the case when $K \geq d(d+1)/2$ and K is large, e.g. when $K = d^q$ for some $q \geq 2$, the top d gaps would be approximately equal and we can roughly claim $H_{2,\text{lin}} \simeq \frac{d}{\Delta_{\min}^2}$. Substituting these in the above error bound of OD-LinBAI yields

$$\Pr(\zeta \neq 1) \leq \left(\frac{4K}{d} + 3\log(d)\right)\exp\left(\frac{(d^2 - B)\Delta_{\min}^2}{32d\log(d)}\right),$$

where we assumed a smaller upper bound for `OD-LinBAI` in its favor. Now if we divide our bound with theirs we get

$$O\left(\frac{q\log(d)}{d^{q-1} + \log(d)}\exp\left(\frac{-d^2\Delta_{\min}^2}{d\log(d)}\right)\right),$$

which is less than 1 and in this case **our error bound is tighter**. Nevertheless, in the case of $K < d(d+1)/2$, it seems their bound is tighter. □

Corner cases: There are special corner cases where digging into different cases of our bounds we can compare them based on Eq. (18). We can see that our error bound is worse than OD-LinBAI in a particular setting where d is small, $q > 8$ is fixed, and $B \to \infty$. However, in the same setting, if $q < 8$, then the conclusion would be the opposite. We also list below several additional cases where our bound improves upon OD-LinBAI:

1. B is fixed, d is fixed, and $q \to \infty$.
2. B is fixed, and $d, q \to \infty$.
3. $B \to \infty$, while d is small and $q < 8$ is fixed (the case we described above).
4. $B \to \infty$ while $d \geq 3$ is fixed, and q grows at the same rate as B. In this case, the term with d^{q-1} grows faster than the exponent, because $e^x \approx 2.7^x$.

E More Experiments

First note that we set $\eta = 2$ in the experiments and in the paper for ease of reading, but note that if we set $\eta = K$ such that the error bound in Eq. (3) is

$$\delta \leq 2K \exp\left(\frac{-B\Delta_{\min}^2}{4\sigma^2 d}\right).$$

which could be order-wise better than Eq. (3) under a specific regime of K and d. The problem is this would make the algorithm totally static and the adaptivity is lost and deteriorates the performance in instances like Section 8.1. As such, there is a trade-off and $\eta = 2$ seems the best for adaptive experiments.

E.1 Details of the Experimental Results

Our preliminary experiments showed that `BayesGap` is sensitive to the choice of the kernel. Therefore, we tune `BayesGap` in each experiment and choose the best kernel from a set of kernels. Note that this gives `BayesGap` an advantage. The best performing kernels are linear, exponential, and Matérn kernels. `BayesGap-Lin` stands for the linear, `BayesGap-exp` for the exponential, and `BayesGap-M` for the Matérn kernels.

We used a combination of computing resources. The main resource we used is the USC Center for Advanced Research Computing (https://carc.usc.edu/). Their typical compute node has dual 8 to 16 core processors and resides on a 56 gigabit FDR InfiniBand backbone, each having 16 GB memory. We also used a PC with 16 GB memory and Intel(R) Core(TM) i7-10750H CPU. For the `Peace` runs we even tried Google Cloud c2-standard-60 instances with 60 CPUs and 240 GB memory.

E.2 Real-World Data Experiments

For this experiment, we use the "Automobile Dataset"[4], which has features of different cars and their prices. We assume the car prices are not readily in hand, rather we get samples of them where the price of car i is $N(p_i, 0.1)$ where p_i is the price of car i in the dataset. The dataset includes 205 cars and we use the most informative subset of features namely 'curb weight', 'width', 'engine size', 'city mpg', and 'highway mpg' so $d = 5$. All the features are rescaled to $[0, 1]$. We want to find the most expensive car by sampling. In each replication, we sample K cars and run the algorithms with the given budget on them. The purpose is to evaluate the performance of the algorithms on a real-world dataset and test their robustness to model misspecification.

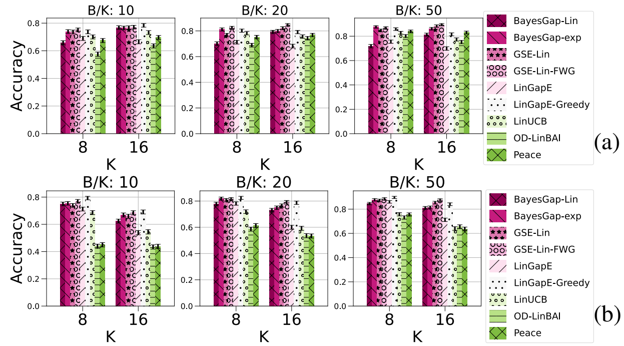

Figure 5: Automobile (5a) and electric motor (5b) datasets.

The other dataset is the "Electric Motor temperature"[5] and we want to find the highest engine temperature. Again, we take samples from the temperature distribution $N(\tau_i, 0.1)$ where τ_i is the temperature of motor i in the dataset. All the features are rescaled to [0,1] and d is 11. The dataset includes $\sim 998k$ data points.

Fig. 5a and Fig. 5b show the results indicating `GSE` variants outperform others or have competing performance with `LinGapE-Greedy`. The experimental results in Fig. 5b show that our algorithm has the highest accuracy in most of the cases despite the fact that we use a linear model for a real-world dataset. This experiment also shows how well all the linear BAI algorithms could generalize to real-world situations.

E.3 More OD-LinBAI Experiments

In this section, we include further experiments to compare `GSE` with `OD-LinBAI`. First, we illustrate that our algorithm outperforms `OD-LinBAI` [Yang and Tan, 2021]. Fig. 6 shows the results for the same experiment as in Section 8.3 but with $\sigma^2 = 1$ (to imitate Yang and Tan [2021]) for different B and K. We can observe for small budgets increasing K our algorithm outperforms `OD-LinBAI` more and more. While if B is extremely large like 500 we can see `OD-LinBAI` outperform `GSE`.

Fig. 7 shows the corner case experiment of Section 5.1 in Yang and Tan [2021] where $\sigma^2 = 1$, $d = 2$, $\theta_* = e_1 = x_1$, $x_K = (\cos(3\pi/4), \sin(3\pi/4))^\top$, and $x_i = (\cos(\pi/4 + \phi_i), \sin(\pi/4 + \phi_i))^\top$ for $i = 2, \cdots, K-1$ where $\phi_i \sim N(0, 0.09^2)$ are i.i.d. samples. In this experiment `OD-LinBAI` outperforms our algorithm. Since `OD-LinBAI` only has 1

[4] J. Schlimmer, Automobile Dataset, https://archive.ics.uci.edu/ml/datasets/Automobile, accessed: 01.05.2021, 1987

[5] J. Bocker, Electric Motor Temperature, https://www.kaggle.com/wkirgsn/electric-motor-temperature, 2019, accessed: 01.05.2021

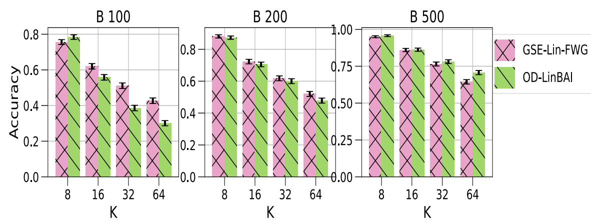

Figure 6: Randomized experiment, comparison with `OD-LinBAI`

stage and simplifies to a G-optimal design, it seems that the specific setting of this experiment makes a G-optimal design most effective in only 1 stage.

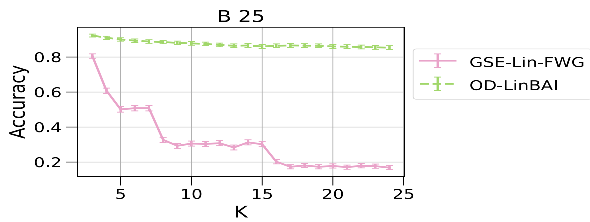

Figure 7: The corner case experiment

F Cumulative Regret and Fixed-Confidence Baselines

By Proposition 33.2 in Lattimore and Szepesvári [2020] there is a connection between the policies designed for minimizing the cumulative regret and BAI policies. However, by the discussion after corollary 33.2 in Lattimore and Szepesvári [2020] the cumulative regret algorithms could under-perform for BAI depending on the bandit instance. This is because these algorithms mostly sample the optimal arm and play suboptimal arms barely enough to ensure they are not optimal. In BAI, this leads to a highly suboptimal performance with asymptotically polynomial simple regret. However, we can compare our BAI algorithm with cumulative regret bandit algorithms as a sanity check or potential competitor. We modify the cumulative regret algorithms into a BAI using a heuristic recommendation policy, mainly by returning the most frequently played arm.

We take `LinUCB` algorithm [Li *et al.*, 2010] and let $x_t \in R^d$ be the pulled arm in stage t, and let the number of stages be equal B. `LinUCB` has the following upper bound on its expected B-stage regret. With probability at least $1-\delta$,

$$x_1^\top \theta_* - \left(\frac{1}{B}\sum_{t=1}^{B} x_t\right)^\top \theta_* \leq \tilde{O}\left(d\sqrt{\frac{\log(1/\delta)}{B}}\right) .$$

where $\tilde{O}$ hides additional log factors in B [Li *et al.*, 2010]. Note that this is a high-probability guarantee on the near-optimality of the average of played arms, the average feature vectors of all pulled arms. If we let this gap be smaller that $\Delta_{\min}$ then we get an error bound of

$$\tilde{O}\left(\exp\left(\frac{-B\Delta_{\min}^2}{d^2}\right)\right) .$$

In a finite arm case, like our setting, we take the most frequently played arm, call it ϱ, as the optimal arm. The reason is that a cumulative regret algorithm plays the potentially optimal arm the most. We conjecture this is the same as the average of played arms, since when we have large enough budget the mode (i.e. ϱ) converges to the mean by the Central Limit Theorem. If we set $d\sqrt{\log(1/\delta)/B} = \Delta_{\min}$ then $\delta = \exp\left(-B\Delta_{\min}^2/d^2\right)$ which is of same order (except we get d instead of d^2 which is better) as our bound in Theorem 1 also we used $\tilde{O}$ here.

For the GLM case, we employ `UCB-GLM` algorithm [Li *et al.*, 2017] which improves the results of ***GLM-UCB*** [Filippi *et al.*, 2010]. According to their most optimistic bound in Theorem 4 of Li *et al.* [2017], we know the `UCB-GLM` cumulative regret is less than $Cc_{\min}\sigma^2/\kappa_l\sqrt{dB\log(B/\delta)}$ for B samples, where κ_l lower bounds the local behavior of $h'(x)$ near θ_* and C is a positive universal constant. Now since simple regret is upper bounded by the cumulative regret, this is also a simple regret bound if we return ϱ, i.e.

$$\begin{aligned} h(x_1^\top\theta_*) - h(x_\varrho^\top\theta_*) &\leq \sum_{t=1}^{B} h(x_1^\top\theta_*) - h(x_t^\top\theta_*) \\ &\leq Cc_{\min}\sigma^2/\kappa_l\sqrt{dB\log(B/\delta)} . \end{aligned}$$

Now if we set this expression equal $\Delta_{\min}$ we have

$$\delta = B\exp\left(\frac{-\Delta_{\min}^2\kappa_l^2}{c_{\min}^2 dB}\right) ,$$

which is comparable to the bound in Theorem 2 but has a slower decrease in B.

We could also turn an FC BAI algorithm into a FB algorithm by stopping at the budget limit and using their recommendation rule. We do a grid search and use the best δ for them. `LinGapE` algorithm [Xu *et al.*, 2018] is the state-of-art algorithm that performs the best among many [Degenne *et al.*, 2020]. By their Theorem 2, we require access to $\{\Delta_i\}_{i\in\mathcal{A}}$ to access the bound and find a proper δ for a given B. This is not very desirable from a practical point of view since we need to know $\{x_i\}_{i\in\mathcal{A}}$ and θ_* beforehand. As such, in our experiments, we find the best δ by a grid search based on the empirical performance of the algorithm for FB BAI. We chose the best one for them in their favor.

G More on Related Work

G.1 Adaptive vs. Static BAI

As argued in Soare *et al.* [2014] and Xu *et al.* [2018], adaptive allocation is necessary for achieving optimal performance. However, it adds an extra $\sqrt{d}$ factor to the confidence bounds, and worsens the sample complexity. Soare *et al.* [2014] and [Xu *et al.*, 2018] propose their FC BAI algorithms $\mathcal{XY}$-adaptive and LinGapE as an attempt to address this issue. Unlike $\mathcal{XY}$-adaptive, LinGapE uses a transductive design and is shown to outperform $\mathcal{XY}$-adaptive. Our algorithm with G-optimal design applies an optimal allocation within each phase, which is different than the greedy and static allocation used by $\mathcal{XY}$-adaptive. We modify LinGapE to be applied to the FB BAI setting and compare it with our algorithm in Section 8. In most cases, our algorithm performed better. Therefore, we believe our algorithm is capable of properly balancing the trade-off between adaptive and static allocations. We further discuss the related work, such as those in the FC BAI setting and optimal design in Appendix G.

G.2 BAI with Successive Elimination

Successive elimination [Karnin *et al.*, 2013] is common in BAI; *Sequential Halving* from Karnin *et al.* [2013] is closely related to our work which is developed for the FB MAB problems. We extend these algorithms to the structured bandits and use an optimal static stage which adapts to the rewards between the stages.

G.3 Optimal Design

Finding the optimal distribution over arms is "optimal design" while finding the optimal number of samples per arm is called "optimal allocation". The exact optimization for many design problems is NP-Hard [Allen-Zhu *et al.*, 2017] Therefore, in the BAI literature, optimal allocation is usually treated as an implementation detail [Soare *et al.*, 2014; Degenne *et al.*, 2020]. and several heuristics are proposed [Khachiyan, 1996; Kumar and Yildirim, 2005]. However, most of these methods are greedy [Degenne *et al.*, 2020]; in particular, they start with one observation per arm, and then continue with the arm that reduces the uncertainty the most in some sense [Soare *et al.*, 2014]. Jedra and Jedra and Proutiere [2020] have a procedure (Lemma 5) that does not start with a sample for each arm but it works for FC setting. Tao *et al.* [2018] suggest that solving the convex relaxation of the optimal design along with a randomized estimator yields a better solution than the greedy methods. This motivates our `FWG` algorithm, which is based on FW fast-rate algorithms. Berthet and Perchet [2017] developed a UCB Frank-Wolfe method for *bandit optimization*. Their goal is to maximize an unknown smooth function with fast rates. Simple regret could be a special case of their framework, however, as of now, it is not clear how to use this method for a BAI bandit problem [Degenne *et al.*, 2019]. In our experiments, we tried several variants of optimal design techniques where the results confirm that `FWG` mostly outperforms the others.

G.4 Related Work on Fixed-Confidence Best Arm Identification

In this section, we discuss the works related to the FC setting in more detail. There are several BAI algorithms for linear bandits under the FC setting. We only discuss the main algorithms. Soare *et al.* [2014] proposed the $\mathcal{XY}$ algorithms based on transductive experimental design [Xu *et al.*, 2018]. In the static case, they fix all arm selections before observing any reward; as a result, it cannot estimate the near-optimal arms. The remedy is an algorithm that adapts the sampling based on the rewards history to assign most of the budget on distinguishing between the suboptimal arms. Abbasi-yadkori *et al.* [2011] introduced a confidence bound based on Azuma's inequality for adaptive strategies which is looser than the static bounds by a $\sqrt{d}$ factor. The $\mathcal{XY}$-adaptive is trying to avoid the extra $\sqrt{d}$ factor as a semi-adaptive algorithm. This algorithm improves the sample complexity in theory, but the algorithm must discard the history of previous phases to apply the bounds. This empirically degrades the performance as shown in Xu *et al.* [2018].

Xu *et al.* [2018] proposed the `LinGapE` algorithm which avoids the $\sqrt{d}$ factor by careful construction of the confidence bounds. However, the sample complexity of their algorithm is linear in K, which is not desirable. In this paper, we derive error bounds logarithmic in K for both linear and generalized linear models.

Jedra and Proutiere [2020] introduced an FC BAI algorithm which tracks an optimal proportion of arm draws and updates these proportions as rarely as possible to avoid compromising its theoretical guarantees. Lemma 5 in this paper provides a procedure that can help avoid greedy sampling of all the arms.

H Heuristic Exploration for More General Models

Here we design an optimal allocation For a general model in `GSE`. Let's start with the following example; consider $K+1$ arms such that $x_i = e_1 + e_1 * \xi$ where $\xi \sim N(0, 0.001)$ for $i \in \{1, \cdots, K\}$ and $x_{K+1} = e_2$. If we have $B = 2K$, in this example, an optimal design would be to sample each $\{x_1, \ldots, x_K\}$ one time and sample x_{K+1}, K times. This is very different than a uniform exploration and motivates our idea of a generalized optimal design as follows. In each stage, cluster the remaining arms into m clusters, e.g. using k-means on the x_i's, then divide the budget equally between the clusters. Now in each cluster do a uniform exploration. In this way, the arms in larger clusters get a smaller budget and we get an equal amount of information in all the directions.

For a general structured setting, we can embed the features to a lower dimension space and then apply the previous algorithms. The candidates include Principle Component Analysis, Method of Moments [Tripuraneni *et al.*, 2021], and encoding with Neural Networks [Riquelme *et al.*, 2018].