Feasible Bidding Strategies through Pure-Exploration Bandits

Julian Katz-Samuels*
University of Michigan
Ann Arbor, Michigan
jkatzsam@umich.edu

Abraham Bagherjeiran
Amazon.com
Palo Alto, California
abagher@amazon.com

ABSTRACT

We apply and extend recent results in feasible arm identification to quickly find a small set of bidding strategies that can simultaneously meet multiple business objectives. We formulate this as an any-m feasible arm identification problem, a pure exploration multi-armed bandit problem where each arm is a D-dimensional distribution represented by a mean vector. The goal is to identify m feasible arms, meaning they satisfy a set of multiple criteria, represented by a polyhedron $P = \{\boldsymbol{x} : A\boldsymbol{x} \leqslant b\} \subset \mathbb{R}^D$. This problem has many applications beyond advertising to online A/B testing, crowdsourcing, clinical trials, and hyperparameter optimization. We propose a new formal algorithm and explore a heuristic improvement through synthetic and real-world datasets.

ACM Reference Format:
Julian Katz-Samuels and Abraham Bagherjeiran. 2019. Feasible Bidding Strategies through Pure-Exploration Bandits. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Multi-armed bandits have a long history starting from [16], in which an agent seeks to maximize an objective function by selecting arms which represents a design parameter (in A/B testing), a dosage for a drug, or a candidate to select for a task (crowdsourcing). Maximizing the objective function requires an agent to continually balance the value of exploitation by simply picking the best arm so far against the cost of exploration–trying new arms. Known as the explore-exploit tradeoff, a multi-armed bandit expects to achieve better objective values by gradually exploiting more and exploring less.

A pure exploration multi-armed bandit addresses the explore-exploit tradeoff sequentially. Rather than maximizing the objective fully, an agent seeks to explore arms for a while and then stop when it finds an arm or a subset of arms nearly optimal as described in [11, 15]. This is determined by setting a fixed termination criteria in advance. Upon reaching the exploration termination point, the agent could focus fully on exploitation [11]. Pure exploration multi-armed bandits follow more closely the common experimentation practice of running an initial trial of many options and then scaling up the best one or a few options.

Pure exploration bandits tend to follow two overall regimes for selecting the termination condition. In the *fixed-confidence* setting [1, 3, 10, 12, 14], a confidence threshold bound to the objective is selected in advance. These methods tend to consume considerable exploration rounds to obtain minute gains in the objective function. In modern online A/B testing environments where experiments cost time and money, it can be impractical to run the experiment long enough to reach a fixed confidence threshold even with guarantees on the threshold [10]. Having a clear time- or value-based budget reflects the practical reality that experiments are always resource-bound. We consider here the *fixed-budget* setting [13], where the exploration budget is determined in advance and the best solution at that time is used.

Enumerating all arms which meet the criteria, though possible, is unnecessarily expensive in terms of number of exploration rounds [3, 13]. In many applications, we seek only a fixed number, say m, arms which meet the criteria, i.e., are feasible. This simplification, as we demonstrate, leads to as much as a 10X reduction in error probability and consequently reduces the number of exploration rounds needed. We find that this setting is quite common, for example, in online advertising where hundreds of parameters combinations can be evaluated on a small amount of traffic to quickly weed out those which we know do not meet all the criteria. Then, a longer test can evaluate a few that do meet the criteria.

In online advertising, bandit algorithms have been used in several areas, such as bidding behavior and mechanism design [6, 8, 20]. Bidding strategies benefit from the exploration inherent to bandit algorithms because bidders only receive feedback when they win an auction. However, long-running exploration introduces risk, so a pure-exploration bandit with a fixed budget offers a good balance between the need to explore and the long-term risk.

We introduce *any-m feasible arm identification*, which identifies m arms that simultaneously satisfy multiple criteria. We consider the setting where each arm, i, has a multi-dimensional distribution represented by a mean vector $\boldsymbol{\mu}_i$ and the criteria are described by a multi-dimensional polyhedron, P. A feasible arm is one whose mean lies within the polyhedron, $\boldsymbol{\mu}_i \in P$. We propose a new algorithm MD-APT-ANY for the fixed budget setting, prove an upper bound, and, finally, demonstrate its performance in experiments on several public datasets.

2 RELATED WORK

To our knowledge, ours is the first study of the any-m feasible arm identification problem under a fixed budget setting.

There has been a recent stream of research on the problem of best arm identification, [2, 4, 5, 9, 15]. In this setting, we want to

*Work completed during internship at Amazon.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

find the best of k feasible arms. Most work assumes one dimensional arms and is not applicable to our setting of multi-dimensional arms. Recently, [1] considered the problem of Pareto front identification for multi-dimensional arms. Their work differs from the current paper since any-m feasible arm identification and Pareto front identification are quite different and our work considers the fixed budget setting versus fixed confidence in [1].

Best arm identification assumes that all arms are already feasible. Feasible arm identification considers the large problem to find the feasible arms in the first place, suggesting a simple two-step approach of finding the feasible arms and then find the best. Chen et al. [3] consider a very general problem of finding the best set of arms, of which any-m feasible arm identification problem is a related subproblem which has two major differences. First, they and their extensions such as [12] consider the fixed confidence setting whereas the current paper deals with the fixed budget setting. Second, their proposed algorithm (see Algorithm 7 in [3]) is impractical since it begins by uniformly sampling the arms until the confidence bound is achieved (based on confidence level $\delta = 0.01$) of the arms imply that a particular set of m arms have means belonging to the polyhedron. On the other hand, we propose practical, fully adaptive algorithms.

Recently, [14] proposed the thresholding bandit problem, in which there are K scalar-valued distributions and a threshold τ and the goal is to determine for each arm i, whether $\boldsymbol{\mu}_i \geqslant \tau$. This problem was generalized in [13] so that there are K vector-valued distributions and a polyhedron P and proposed algorithms MD-APT and MD-SAR to determine for each arm i, whether $\boldsymbol{\mu}_i \in P$. Under a fixed -budget scheme, both MD-APT and MD-SAR spend considerable budget on enumerating *all* feasible arms. In contrast, we demonstrate that there is considerable gain to emitting only m, any-m feasible arms rather than all of them.

The any-m feasible arm identification problem under a fixed-budget setting is more practically relevant than traditional feasible arm identification since in many applications, it is sufficient to find a certain number of arms that do satisfy the criteria rather than having to determine whether each and every arm satisfies them. It is unnecessary to determine for the other arms whether they belong to the polyhedron. Furthermore, algorithms developed for traditional feasible arm identification problem waste much of their sampling effort to arms with arms near the boundary. This leads to poor performance when used for the any-m feasible arm identification problem.

3 SETUP

In this section, we formulate the any-m feasible arm identification problem. To begin, we define some notation. For all $n \in \mathbb{N}$, define $[n] = \{1, \ldots, n\}$. For any $A \subset \mathbb{R}^D$ and $\boldsymbol{x} \in \mathbb{R}^D$, define $\text{dist}(\boldsymbol{x}, A) = \inf_{\boldsymbol{y} \in A} \|\boldsymbol{x} - \boldsymbol{y}\|_2$. Let ∂A denote the boundary of A, i.e., $\partial A = \bar{A} \backslash A^\circ$ where $\bar{A}$ denotes the closure of A and A° denotes the interior of A. Let $\mathbf{1}\{\cdot\}$ denote the indicator function and $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^D$. Define $\mathcal{S}^{D-1} = \{\boldsymbol{x} \in \mathbb{R}^D : \|\boldsymbol{x}\|_2 = 1\}$. Let U be a finite set and f be a scalar-valued function with domain containing U, and define

$$\max_{x \in U}^{(l)} f(x) \coloneqq \begin{cases} \max_{x \in U | \{y \in U : f(y) \geqslant f(x)\}| \geqslant l-1} f(x) & |U| \geqslant l \\ -\infty & \text{otherwise} \end{cases}$$

In words, $\max_{x \in U}^{(l)} f(x)$ is the value of the lth largest $x \in U$ under $f(\cdot)$ and if $|U| < l$, then it is $-\infty$.

Suppose we are given K stochastic arms. When the ith arm is pulled, a realization is drawn i.i.d. from a D-dimensional distribution ν_i. Denote $\boldsymbol{\mu}_i = \mathbb{E}_{\boldsymbol{X} \sim \nu_i} \boldsymbol{X}$. We assume that the agent is given a polyhedron $P = \{\boldsymbol{x} : A\boldsymbol{x} \leqslant \boldsymbol{b}\}$ where $A \in \mathbb{R}^{M \times D}$ such that

$$A = \left(\boldsymbol{a}_1^t, \ldots, \boldsymbol{a}_M^t\right)^T$$

and $\boldsymbol{b} \in \mathbb{R}^M$. By dividing each constraint j by $\|\boldsymbol{a}_j\|_2$, we can assume without loss of generality that $\|\boldsymbol{a}_j\|_2 = 1$ for all $j \in [M]$. For simplicity, we assume that P has positive volume and that $\text{diam}(P) \leqslant 2B$.

In the fixed budget setting, the game is as follows: there are T rounds and at each round t, the agent chooses an arm $I_t \in [K]$ and observes a realization $\boldsymbol{X}_t \sim \nu_{I_t}$. The goal in the game is to identify m arms with means $\boldsymbol{\mu}_i \in P$. In the degenerate case where there are fewer than m arms in the polyhedron, our algorithm identifies at least $|\{i \in [K] : \boldsymbol{\mu}_i \in P\}|$ arms with means $\boldsymbol{\mu}_i$ that satisfy $\text{dist}(\boldsymbol{\mu}_i, P) \leqslant 2\epsilon$, where $\epsilon > 0$ is a specified tolerance.

Our analysis assumes that each ν_i is a multi-dimensional sub-Gaussian distribution, whose definition we repeat here (see [19] for more details). Let X be a scalar random variable. We say that X is *R-sub-Gaussian* if $\mathbb{E}\exp(\frac{X^2}{R^2}) \leqslant 2$. The sub-Gaussian norm of X is the smallest R that satisfies the above requirement, as follows:

$$\|X\|_{\psi_2} = \inf\{R > 0 : \mathbb{E}\exp(\frac{X^2}{R^2}) \leqslant 2\}$$

A random vector $\boldsymbol{X} \in \mathbb{R}^D$ is sub-Gaussian if $\boldsymbol{X}^t \boldsymbol{a}$ is sub-Gaussian for all $\boldsymbol{a} \in \mathbb{R}^D$. The sub-Gaussian norm of $\boldsymbol{X}$ is defined as

$$\|\boldsymbol{X}\|_{\psi_2} = \sup_{\boldsymbol{a} \in \mathcal{S}^{D-1}} \left\|\boldsymbol{X}^t \boldsymbol{a}\right\|_{\psi_2}.$$

We say that a random vector $\boldsymbol{X}$ is R-sub-Gaussian if $\|\boldsymbol{X}\|_{\psi_2} \leqslant R$. Henceforth, we assume that $\nu_1, \ldots, \nu_K$ are R-sub-Gaussian. See [18, 19] for further details on sub-Gaussian distributions.

4 ALGORITHM

In this section, we present our algorithm MD-APT-ANY. First, we introduce some notation. Let I_t denote the index of the arm chosen at time t and $T_i(t) = \sum_{s=1}^{t-1} \mathbf{1}\{I_s = i\}$ denote the number of pulls of arm i at round t. Let $X_{i,j,t}$ denote the tth realization of the jth coordinate of ν_i and $\widehat{\boldsymbol{\mu}}_{i,t}$ denote the estimate of $\boldsymbol{\mu}_i$ after t samples, i.e., $\widehat{\boldsymbol{\mu}}_{i,t} = (\widehat{\mu}_{i,1,t}, \ldots, \widehat{\mu}_{i,D,t})^t$ where $\widehat{\mu}_{i,j,t} = \frac{1}{t}\sum_{s=1}^{t} X_{i,j,s}$. Define the distance to the polyhedron P of arm i at time t, $\widehat{\Delta}_{i,t}^{(\epsilon)}$, as follows:

$$\widehat{\Delta}_{i,t}^{(\epsilon)} = \begin{cases} \min_{j \in [M]} b_j - \boldsymbol{a}_j^t \widehat{\boldsymbol{\mu}}_{i,t} + \epsilon & : \widehat{\boldsymbol{\mu}}_{i,t} \in P \\ \text{dist}(\widehat{\boldsymbol{\mu}}_{i,t}, P) + \epsilon & : \widehat{\boldsymbol{\mu}}_{i,t} \notin P \end{cases}$$

where $\widehat{\Delta}_{i,t}^{(\epsilon)}$ is the estimator of the margin between feasible and infeasible region, or the distance to the closest side if the mean is within P and the distance to the closest point inside P if the mean is outside P.

Background. Algorithm 1, introduced as MD-APT by [13] generalizes APT from [14] to multi-dimensional distributions. MD-APT is a nearly optimal algorithm for feasible arm identification; MD-APT(ϵ) determines correctly for each arm i for which $\text{dist}(\boldsymbol{\mu}_i, \partial P) \geqslant \epsilon$, whether $\boldsymbol{\mu}_i \in P$, and its probability of error decays as $\exp(-c\frac{T}{H_\epsilon})$ where $H_\epsilon = \sum_{i\in[K]}[\,\text{dist}(\boldsymbol{\mu}_i, \partial P) + \epsilon]^{-2}$. Its key idea is to sample each arm proportionally to $[\,\text{dist}(\mu_i, P) + \epsilon]^{-2}$. The main drawback of MD-APT for any-m feasible arm identification is that it wastes budget sampling arms near the boundary to find all feasible arms rather than finding only m feasible arms.

The Algorithm. We propose MD-APT-ANY, shown in Algorithm 2, for any-m feasible arm identification. MD-APT-ANY takes as an input a tolerance parameter $\epsilon > 0$. It divides the budget T into $\left\lceil \log_2(\frac{B}{\epsilon}) \right\rceil$ rounds. Each round r consists of two main steps. First, it runs MD-APT(ϵ_r) for $\frac{T}{\lceil \log_2(B) \rceil}$ iterations with a decreasing tolerance for feasibility, $\epsilon_r := \frac{B}{2^r}$. Next, it performs a test to determine whether there are at least m arms close enough to the boundary. Specifically, if there are m arms such that $\widehat{\boldsymbol{\mu}}_{i,T_i(t_r+1)} \in P$ and $\text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(t_r+1)}, \partial P) \geqslant \epsilon_{r-1}$, then it concludes that these arms belong to P and returns m of them. We note that in practice once the condition in line 6 is met, one could run MD-APT($\frac{B}{2^r}$) for the remaining number of iterations. Finally, if at the end of round $\left\lceil \log_2(\frac{B}{\epsilon}) \right\rceil$, the condition in line 6 is still not satisfied, the algorithm returns a set of arms $\widehat{S}$ such that each $i \in \widehat{S}$ satisfies $\text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(T+1)}, P) \leqslant \epsilon$ and $|\widehat{S}|$ has size as follows:

$$\min\left(m, |\{i \in [K] : \text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(T+1)}, P) \leqslant \epsilon\}|\right)$$

Algorithm 1 MD-APT: Multi-dimensional Anytime Parameter-Free Thresholding algorithm [13]

1: **Input:** K arms, polyhedron P, tolerance ϵ, budget T
2: **for** $t = 1, \ldots, T$ **do**
3: **if** $t \leqslant K$ **then**
4: Sample $\boldsymbol{X}_t \sim \nu_t$.
5: **else**
6: Choose $I_t = \arg\min_i \widehat{\Delta}^{(\epsilon)}_{i,T_i(t)}\sqrt{T_i(t)}$. Sample $\boldsymbol{X}_t \sim \nu_{I_t}$.

Next, we give an upper bound for the performance of the algorithm. Define Γ_m as the distance of the mth arm furthest from the boundary, as:

$$\Gamma_m = \max\left(0, \max^{(m)}_{i:\boldsymbol{\mu}_i\in P} \text{dist}(\boldsymbol{\mu}_i, \partial P)\right),$$
$$H^{(\epsilon)}_m = \sum_{i\in[K]} [\,\text{dist}(\boldsymbol{\mu}_i, \partial P) + \max(\Gamma_m, \epsilon)]^{-2}.$$

where $H^{(\epsilon)}_m$ is a worst-case proximity to the feasibility boundary for all arms. The following theorem bounds the number of exploration rounds required to either the output the m arms in the polyhedron or the m arms within a distance of ϵ.

Algorithm 2 MD-APT-ANY

1: **Input:** K arms, polyhedron P, tolerance ϵ, budget T, m number of desired arms
2: Define $\epsilon_r := \frac{B}{2^r}$ and $t_r = r\frac{T}{\lceil \log_2(\frac{B}{\epsilon}) \rceil}$
3: **for** $r = 1, \ldots, \left\lceil \log_2(\frac{B}{\epsilon}) \right\rceil$ **do**
4: Run MD-APT(ϵ_r) for $\frac{T}{\lceil \log_2(B) \rceil}$ iterations.
5: $\widehat{S}_r = \{i \in [K] : \widehat{\boldsymbol{\mu}}_{i,T_i(t_r+1)} \in P \text{ and } \text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(t_r+1)}, \partial P) \geqslant \epsilon_{r-1}\}$
6: **if** $|\widehat{S}_r| \geqslant m$ **then**
7: $\widehat{S} := \arg\max_{Z\subset\widehat{S}_r, |Z|=m} \sum_{i\in Z} \text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(t_r+1)}, \partial P)$
8: **Return** $\widehat{S}$
9: Pick any $\widehat{S} \subset \{i \in [K] : \text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(T+1)}, P) \leqslant \epsilon\}$ such that $|\widehat{S}| = \min\left(m, |\{i \in [K] : \text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(T+1)}, P) \leqslant \epsilon\}|\right)$.
10: **Return** $\widehat{S}$

Theorem 1. *Let $\epsilon > 0$. Suppose that $T \geqslant 2K\left\lceil \log_2(\frac{B}{\epsilon}) \right\rceil$. Then for some universal constant $c > 0$, with probability at least*

$$1 - c\log_2\left(\frac{B}{\epsilon}\right)\log(T)K5^D\exp\left(\frac{-T}{2592\log_2(\frac{B}{\epsilon})H_mR^2}\right),$$

(1) *if $\epsilon < \frac{\Gamma_m}{2}$, MD-APT-ANY($\epsilon$) outputs $\widehat{S}$ such that $|\widehat{S}| = m$ and $\widehat{S} \subset \{i \in [K] : \boldsymbol{\mu}_i \in P\}$;*
(2) *otherwise, MD-APT-ANY(ϵ) outputs $\widehat{S}$ such that*

$$\min\left(|\{i \in [K] : \boldsymbol{\mu}_i \in P\}|, m\right) \leqslant |\widehat{S}| \leqslant m$$
$$\forall i \in \widehat{S} : \text{dist}(\boldsymbol{\mu}_i, P) \leqslant 2\epsilon$$

A few remarks are in order. First, it is possible to change MD-APT-ANY so that if the test in line 6 is never satisfied, then it simply concludes that there are not m arms with $\text{dist}(\boldsymbol{\mu}_i, P^c) \geqslant 2\epsilon$. Then, the proof of Theorem 1 (omitted due to space restrictions) implies that this variant of MD-APT-ANY would be correct with the probability given in Theorem 1. Second, there is a tradeoff in how to select ϵ. On the one hand, the smaller ϵ is, the larger the probability of error via the term $\log_2(\frac{B}{\epsilon})$. But, on the other hand, the smaller ϵ is, the more reasonable it is to believe that $\epsilon < \frac{\Gamma_m}{2}$ and if $\frac{\Gamma_m}{2} > \epsilon$, MD-APT-ANY finds arms closer to P.

Finally, whereas the probability of error of MD-APT decays as $\exp\left(-c\frac{T}{H}\right)$, the probability of error of MD-APT-ANY decays as $\exp\left(\frac{-T}{\log_2(\frac{B}{\epsilon})H_m}\right)$. Although there appears to be some looseness in our algorithm with the term $\log_2\left(\frac{B}{\epsilon}\right)$, it is usually dominated by the difference between H and H_m. Indeed, H can be arbitrarily larger than H_m. Intuitively, we can think of H_m as limiting how far past the boundary we need to go to find the m feasible arms, rather than going all the way across the boundary H.

Intuition for MD-APT-ANY. MD-APT-ANY is based on two main observations. First, if Γ_m were known, then one could run MD-APT($\frac{\Gamma_m}{2}$) for T iterations, after which with probability at least (ignoring lower order terms) $1 - \exp(-\frac{T}{H_m})$, the following two

events occur: 1) any feasible arm whose distance from the boundary of P is at least as large as Γ_m (i.e., $\text{dist}(\boldsymbol{\mu}_i, P^c) \geqslant \Gamma_m$) would satisfy $A\widehat{\boldsymbol{\mu}}_{i,T_i(T+1)} \leqslant \boldsymbol{b} - \gamma_m \mathbf{1}$ where γ_m is a period-arm distance to the boundary; and 2) any infeasible arm is not too far away from the complement of the boundary P^c or, more specifically, $A\widehat{\boldsymbol{\mu}}_{i,T_i(T+1)} \not\leqslant \boldsymbol{b} - \gamma_m \mathbf{1}$. Thus, MD-APT($\frac{\Gamma_m}{2}$) avoids sampling arms close to the boundary too much and still allows for distinguishing m arms in P.

Yet, Γ_m is not known in practice, so it is not possible to run MD-APT as is. The second observation is that we can overcome our lack of knowledge of Γ_m by applying the principle of optimism in the face of uncertainty with respect to Γ_m. Specifically, since $\Gamma_m \in \left[\frac{B}{2^{r+1}}, \frac{B}{2^r}\right)$ for some $r \in \mathbb{N}$, if we then run MD-APT($\frac{B}{2^r}$) for T iterations for increasing r, then for $r = \left\lceil \log_2\left(\frac{2B}{\Gamma_m}\right)\right\rceil$, we have that

$$\frac{B}{2^r} \in \left[\frac{\Gamma_m}{2}, \frac{\Gamma_m}{4}\right]$$

so that the observation concerning MD-APT($\frac{\Gamma_m}{2}$) in the previous paragraph applies only $T\log(\frac{2B}{\Gamma_m})$ total iterations have passed. Of course, since we are concerned with the fixed budget setting here where only T rounds are permitted, we apply this idea by incorporating a tolerance ϵ into MD-APT-ANY.

Comparison with MD-APT. Consider the example $m = 1$, $\mu_1 = .5 + c\epsilon, \mu_{2:100} = .5 - \epsilon$ and $P = \{x \in \mathbb{R} : x \geqslant .5\}$, where c is a positive constant. Running MD-APT alone simulates a static allocation algorithm that knows the gaps $\text{dist}(\boldsymbol{\mu}_i, \partial P)$. This algorithm would sample the ith arm $\frac{T}{\text{dist}(\boldsymbol{\mu}_i, \partial P)^2 H}$ times. Thus, this algorithm would pull arm 1 $\frac{T}{c^2(K-1)+1}$ times. Thus, by letting ϵ go to 0 and c to ∞, we have that arm 1 would be pulled $\frac{T}{K^d}$ times for any positive constant d. More formally, it can be shown that the probability of error is on the order of $\exp(-\frac{T}{H})$.

5 EXPERIMENTS

We conduct experiments on a public real-time bidding dataset and several synthetic and real-world datasets. We compare our proposed MD-APT-ANY against three state of the art algorithms: MD-APT, MD-SAR, and uniform allocation (UA).

Each experiment measures the probability of an error after fixed budget T rounds of exploration. The budget is assumed to be a fixed input, but for the experiments we choose T to be on the same order of H_m. The error probability measures how often the algorithm pulls an infeasible arm. It does not depend on the number of feasible arms required, m.

5.1 Bidding Strategies

We cast bidding strategy selection as a feasible arm identification in which the arms are not the bids themselves but rather policies that can be used to set the bids. Thus the pure-exploration bandit is running a large-scale A/B/n test, trying many strategies and returning at least m feasible strategies. We use a public advertising benchmark dataset collected by [21] from the RTB algorithm competition of iPinYou in 2013.

The arms are the existing bidding algorithms described in [21], each falling into the categories: constant (const), random (rand), Mcpc, and a linear-form bidding (lin). Const, rand, and lin each have a hyperparameter, which results in a total of 144 bidding strategies (arms). Again, our method is not a novel bidding strategy but rather selects from a set of strategies.

Feasibility is defined here as improving over all criteria relative to a baseline model. We consider two business criteria. Average cost per 1000 impressions (CPM), $\mu_{i,1}$, and average click through rate (CTR), $\mu_{i,2}$. The baseline method is a fixed set of parameters for the linear bidding algorithm, whose CPM and CTR are $\mu_{l,1}$ and $\mu_{l,2}$, respectively. The feasible arm identification task is to determine if $\boldsymbol{\mu}_i \in P$ or equivalently, whether $\mu_{i,1} \geqslant \mu_{l,1}$ and $\mu_{i,2} \geqslant \mu_{l,2}$.

In Table 1, column 5 (A/B E), we see that MD-APT-ANY shows a 46% relative improvement over MD-APT. By focusing sampling time on determining whether at least a few arms are feasible rather than returning all feasible arms, MD-APT-ANY can spend exploration budget within the interior of the polyhedron.

5.1.1 MD-APT-ANY-F. During our implementation we noticed that MD-APT-ANY finds quasi-feasible arms quickly and then spends budget validating feasibility. By focusing more on the quasi-feasible arms, we might see an improvement. We call this heuristic method MD-APT-ANY-F. During even iterations, it samples only the m quasi-feasible arms that maximize $\text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(t)}, P^c) - \text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(t)}, P)$ where P^c is the complement of the polyhedron P. This simple heuristic improvement turned out to improve over our main algorithm. This is likely because it is sampling the feasible arms more often than would be warranted based on the criteria in MD-APT-ANY.

5.2 Real-world Dataset Experiments

Although conceived for bidding strategy identification, we evaluated the algorithm on several real-world datasets. Each of the following real-world datasets poses a multi-dimensional any-m-feasible arm identification problem. In each task, there is more than one desirable criteria that must be satisfied for an arm to be acceptable. Moreover, each task seeks only m feasible solutions.

Medical Experiment. We consider the problem in clinical trials of identifying the drug that: 1) has a sufficiently high probability of being effective; and 2) meets some safety standard. We use data from [7] (see ARCR20 in week 16 in Table 2 and Table 3), which studies the drug secukinumab for treating rheumatoid arthritis. They test four dosage levels (25mg, 75mg, 150mg, 300mg) and a placebo. Each arm corresponds to a drug and has two attributes: let $\mu_{i,1}$ denote the probability of being effective and $\mu_{i,2}$ the probability of causing an infection or infestation. The dosage levels 25mg, 75mg, 150mg, and 300mg have means $\boldsymbol{\mu}_1 = (.34, .259)^\top$, $\boldsymbol{\mu}_2 = (.469, .184)^\top$, $\boldsymbol{\mu}_3 = (.465, .209)^\top$, $\boldsymbol{\mu}_4 = (.537, .293)^\top$, respectively, and the placebo has mean $\boldsymbol{\mu}_5 = (.36, .36)^\top$. We deem a drug acceptable if the probability of being effective is at least .4 and the probability of causing an infection is at most .25. We set $m = 1$, i.e., the goal is to find one acceptable drug. In our experiment, whenever arm i is chosen two Bernoulli random variables with means given by $\boldsymbol{\mu}_i$ are drawn.

Crowdsourcing Experiment. We examine the task of using a limited budget of queries to find crowdsourcing workers that

	E3 (m = 5)	E3 (m = 15)	E3 (m = 30)	Med E (m = 1)	A/B E (m = 1)	Crowd E (m = 5)
MD-APT-ANY-F	0.20 (0.03)	0.23 (0.03)	0.35 (0.03)	0.11 (0.00)	0.03 (0.01)	0.18 (0.03)
MD-APT-ANY	0.29 (0.03)	0.47 (0.04)	0.36 (0.03)	0.13 (0.00)	0.40 (0.03)	0.32 (0.03)
MD-APT	0.36 (0.03)	0.33 (0.03)	0.40 (0.03)	0.21 (0.01)	0.75 (0.03)	0.62 (0.03)
MD-SAR	0.35 (0.03)	0.40 (0.03)	0.89 (0.02)	0.14 (0.00)	0.67 (0.03)	0.44 (0.04)
UA	0.84 (0.03)	0.82 (0.03)	0.92 (0.02)	0.13 (0.00)	0.53 (0.04)	0.42 (0.03)

Table 1: Experiments on synthetic dataset (E3) and real-world datasets: Estimated probability of error with standard errors.

are sufficiently likely to: 1) give the correct answer; and 2) respond on average at a suitable speed. We use a crowdsourcing dataset collected by [17] in which Amazon Mechanical Turk workers evaluate the content of tweets. We only consider workers that have answered at least 50 questions, leaving a total of 44 workers. Here, $\mu_{i,1}$ is the probability of being correct and $\mu_{i,2}$ is the average amount of time required. We set $P = \{\boldsymbol{x} : x_1 \geqslant .75, x_2 \leqslant 15\}$, i.e., we deem a worker satisfactory if he answers correctly with probability at least .75 and on average within 15 seconds. We set $m = 5$, i.e., the goal is to identify 5 workers that satisfy the criteria. Because the data for each worker is limited, whenever an algorithm pulls an arm corresponding to a worker, it samples a data point associated with that worker uniformly at random with replacement.

5.2.1 Results. The results on real-world datasets illustrate the obvious advantage of solving the any-m feasible compared to the fully feasible arm identification. In all experiments, shown in Table 1, either MD-APT-ANY or its modification MD-APT-ANY-F showed significant improvement over the current state of the algorithms for feasible arm identification, achieving the lowest probability of error in each of the experiments.

In the any-m feasible arm identification problem, when there are few arms, uniform allocation (UA) may have an advantage of simply being lucky. This appears to be the case in our medical experiment (Med E) in which uniform allocation has a the best error rate. Although, as indicated in [13], UA does not return all the feasible arms, it does appear to find the best one of four. For the other experiments, UA simply has too many options to be effective.

5.3 Synthetic Experiments

To better understand MD-APT-ANY and the intuition behind MD-APT-ANY-F, we considered several synthetic datasets, defined below.

Experiment 1. The polyhedron is $P = \{\boldsymbol{x} \in \mathbb{R}^5 : x_i \geqslant .5\}$. The distributions are 5-dimensional multivariate normal distributions with covariance matrix $\frac{1}{4}I$. 50 of the arms have mean $(0.49999)^{\otimes 5}$ and 50 of the arms have mean $(0.6)^{\otimes 5}$. We vary $m = 5, 10, 15$.

Experiment 2. The polyhedron is $P = \{\boldsymbol{x} \in \mathbb{R}^5 : x_i \geqslant .5\}$. Arm $i \in [50]$ has mean $(0.5 + .2\frac{i}{50})^{\otimes 2}$. Arm $i \in [200] \backslash 50$ has mean $(0.5 - .2\frac{i-50}{150})^{\otimes 2}$.

Experiment 3. The polyhedron is $P = \{\boldsymbol{x} \in \mathbb{R}^5 : \boldsymbol{x}_i \leqslant \boldsymbol{x}_{i+1} \forall i \in [4]\}$. We use 5-dimensional Bernoulli distributions. 50 of the arms have mean $(.1, .3, .5, .3, .1)^\top$ and 50 of the arms have mean $(0.1, .3, .15, .7, .9)^\top$.

5.3.1 Results. The results on synthetic datasets show that MD-APT-ANY clearly outperforms the other feasible arm identification algorithms. Figure 1 shows a kernel density estimate of the distribution of arm pulls for Experiment 2, where each arm is represented by its mean value. In Experiment 2, arms with mean greater than 0.5 are feasible. In Figure 1a, MD-SAR pulls arms close to the boundary and is slightly more likely to pull arms within the feasible set. With an accept and reject sampling technique, MD-SAR, spends budget on determining whether an arm is feasible within the margin before deciding to accept or reject it. Although MD-APT, shown in Figure 1b, improves on sampling outside the feasible set, it focuses even more on arms that are nearly feasible. In contrast, the proposed algorithm MD-APT-ANY, shown in Figure 1c, spends the exploration budget on exploring arms clearly outside the feasible region–which is illustrated in the figure as a skewed distribution with peak and most of the mass on the left (infeasible) side. This suggests that MD-APT-ANY quickly identifies some feasible arms and then continues exploring the infeasible arms to find better candidates. This confirms our intuition for the heuristic in MD-APT-ANY-F, which places additional emphasis on feasible arms–essentially double-checking that the quickly identified arms really are feasible.

Discussion. When confronted with problems having many arms, such as the A/B test and crowdsourcing experiments, MD-APT-ANY and its extension MD-APT-ANY-F exhibit dramatic reductions in probability of error compared to state-of-the-art algorithms, as much as 10X for one dataset. Drawing upon the intuitive behavior illustrated in Figure 1, MD-APT-ANY quickly identifies feasible arms and then continues to explore other possibilities. In addition, the heuristic modification MD-APT-ANY-F shows that simply restricting the sampling space to feasible arms is a good strategy.

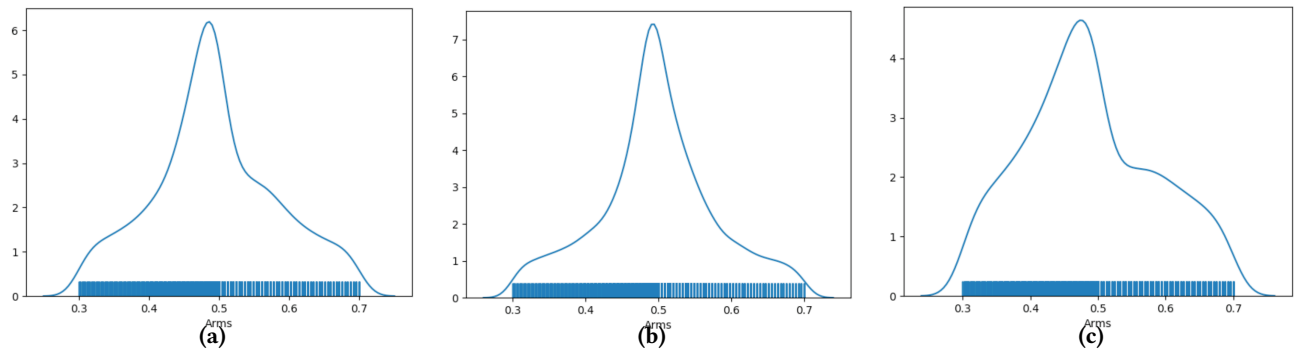
6 CONCLUSION

We introduce and propose a solution for the any-m feasible arm identification problem. In this setting, an agent plays a sequential game where at each round, it pulls one of the arms and observes an independent realization from a distribution associated with the arm. In contrast to the classical multi-armed bandit setting, the agent in a pure-exploration multi-armed bandit seeks by the end of the game only m arms meeting several criteria rather than the single best arm in just a single criteria.

Identifying a few known good solutions can suffice for a variety of real-world environments. For example, in A/B testing, we may have many parameters of the same algorithm to test. It is infeasible to test all the parameters thoroughly, but we also do not to hastily select just one. Instead, we would like to find a few known good solutions and run these in a longer test. Identifying just a few rather than all feasible arms dramatically reduces the error. Moreover, it

	E1 (m = 5)	E1 (m = 10)	E1 (m = 15)	E2 (m = 5)	E2 (m = 15)	E2 (m = 20)
MD-APT-ANY-F	0.03 (0.01)	0.05 (0.01)	0.15 (0.03)	0.24 (0.03)	0.11 (0.02)	0.12 (0.02)
MD-APT-ANY	0.08 (0.02)	0.07 (0.02)	0.17 (0.03)	0.54 (0.04)	0.65 (0.03)	0.71 (0.03)
MD-APT	0.14 (0.02)	0.11 (0.02)	0.43 (0.03)	0.54 (0.04)	0.95 (0.02)	0.98 (0.01)
MD-SAR	0.14 (0.02)	0.15 (0.02)	0.34 (0.03)	0.47 (0.04)	0.80 (0.03)	0.85 (0.03)
UA	0.33 (0.03)	0.44 (0.04)	0.69 (0.03)	0.32 (0.03)	0.59 (0.03)	0.77 (0.03)

Table 2: Experiments on synthetic datasets: Estimated probability of error with standard errors, comparing the proposed algorithm MD-APT-ANY against state of the art algorithms MD-APT and MD-SAR.



(a) (b) (c)

Figure 1: Kernel density estimate (KDE) of the distribution of arm pulls for real-valued arms in Experiment 2 (m=20) for three algorithms (a): MD-SAR , (b): MD-APT, (c): Proposed algorithm: MD-APT-ANY. MD-APT-ANY focuses its exploration budget on eliminating infeasible arms–note the peak of the distribution just less than the feasible boundary of 0.5.

unlocks practical value by increasing the exploration budget that can be used to consider even more alternatives.

The results suggest that, not only is MD-APT-ANY a good solution to the any-m feasible arm identification problem, the heuristic modification MD-APT-ANY-F may be able to improve other algorithms. In future work, we may apply this heuristic to the other benchmark algorithms. We would also like to further develop its theory, which we believe will lead to further improvements.

References.

[1] Peter Auer, Chao-Kai Chiang, Ronald Ortner, and Madalina Drugan. 2016. Pareto front identification from stochastic bandit feedback. *Artificial Intelligence and Statistics* (2016), 939–947.

[2] Sébastian Bubeck, Tengyao Wang, and Nitin Viswanathan. 2013. Multiple identifications in multi-armed bandits. *International Conference on Machine Learning* (2013), 258–265.

[3] Lijie Chen, Anupam Gupta, Jian Li, Mingda Qiao, and Ruosong Wang. 2017. Nearly Optimal Sampling Algorithms for Combinatorial Pure Exploration. *Proceedings of Machine Learning Research* 65 (2017), 1–53.

[4] Shouyuan Chen, Tian Lin, Irwin King, Michael Lyu, and Wei Chen. 2014. Combinatorial pure exploration of multi-armed bandits. *Advances in Neural Information Processing Systems* (2014), 379–387.

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

[6] Nicola Gatti, Alessandro Lazaric, and Francesco Trovò. 2012. A truthful learning mechanism for contextual multi-slot sponsored search auctions with externalities. In *Proceedings of the 13th ACM Conference on Electronic Commerce*. ACM, 605–622.

[7] Mark Genovese, Patrick Durez, Hanno Richards, Jerzy Supronik, Eva Dokoupilova, Vadim Mazurov, and Jacob Aelion. 2013. Efficacy and safety of secukinumab in patients with rheumatoid arthritis: a phase II, dose-finding, double-blind, randomised, placebo controlled study. *Annals of the rheumatic diseases* 72 (2013), 863–869.

[8] Rica Gonen and Elan Pavlov. 2007. An incentive-compatible multi-armed bandit mechanism. In *Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing*. ACM, 362–363.

[9] Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. 2014. lil'ucb: An optimal exploration algorithm for multi-armed bandits. *Conference on Learning Theory* (2014), 424–439.

[10] Kevin G Jamieson and Lalit Jain. 2018. A Bandit Approach to Sequential Experimental Design with False Discovery Control. In *Advances in Neural Information Processing Systems*. 3664–3674.

[11] Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. 2012. PAC Subset Selection in Stochastic Multi-armed Bandits.. In *ICML*, Vol. 12. 655–662.

[12] Hideaki Kano, Junya Honda, Kentaro Sakamaki, Kentaro Matsuura, Atsuyoshi Nakamura, and Masashi Sugiyama. 2017. Good Arm Identification via Bandit Feedback. *arXiv preprint arXiv:1710.06360* (2017).

[13] Julian Katz-Samuels and Clayton Scott. 2018. Feasible Arm Identification. (2018), 2540–2548.

[14] Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. 2016. An optimal algorithm for the thresholding bandit problem. *Proceedings of The 33rd International Conference on Machine Learning* (2016), 1690–1698.

[15] Shie Mannor and John Tistisklis. 2004. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research* (2004), 623–648.

[16] Herbert Robbins. 1952. Some Aspects of the Sequential Design of Experiments. *Bull. Amer. Math. Soc.* 58 (1952), 527–535.

[17] Matteo Venanzi, John Guiver, Pushmeet Kohli, and Nicholas R Jennings. 2016. Time-sensitive bayesian information aggregation for crowdsourcing systems. *Journal of Artificial Intelligence Research* 56 (2016), 517–545.

[18] Roman Vershynin. 2012. Introduction to the non-asymptotic analysis of random matrices. In *Compressed Sensing: Theory and Applications*, Yonina Eldar and Gitta Kutyniok (Eds.). Cambridge University Press, 210–268.

[19] Roman Vershynin. 2018. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Vol. 47. Cambridge University Press.

[20] Marcin Waniek, Long Tran-Tranh, and Tomasz Michalak. 2016. Repeated dollar auctions: A multi-armed bandit approach. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 579–587.

[21] Weinan Zhang, Shuai Yuan, Jun Wang, and Xuehua Shen. 2014. Real-time bidding benchmarking with ipinyou dataset. *arXiv preprint arXiv:1407.7073* (2014).