AdKDD 2019, August 5, 2019

Feasible Bidding Strategies through Pure-Exploration Bandits

Julian Katz-Samuels, **Abraham Bagherjeiran (*)**
(*) Amazon Advertising – Sponsored Brands, Palo Alto, CA

amazonadvertising

Bidding in online advertising

A three-sided marketplace

Publisher

Requests ads, runs auctions, wants more ad spend

Advertiser

Bids for ad opportunities across multiple publishers to drive their business

User

Interacts with the ad (or not), wants meaningful ad experiences

Bidding strategies

A strategy is an algorithm that real-time bidders use to calculate the bid for an ad opportunities. Different strategies may be needed depending on the context

- First-price auctions
- Dynamic floor pricing
- Marketplace reserve pricing
- Changes to ad campaign setup

amazonadvertising

Selecting a bidding strategy

Subject of this paper

We are not proposing a new bidding strategy but a process for selecting a few among many strategies. Why can't we just test all combinations online or offline?

- Testing strategies offline introduces errors
- Test a few strategies online
- Too many strategies to test all online
- Too many parameters to test online

Many possible bidding strategies

Some example bidding strategies [21]

Fixed Bid (const)

Bid the same amount all the time

True Value (mCPC)

Bid p(Action) * V(Action)

Linear Scaling (lin)

Scale mCPC by a constant, such as historical p(Action)

Random (rand)

Bid a random value

[21] Weinan Zhang, Shuai Yuan, Jun Wang, and Xuehua Shen. 2014. Real-time bidding benchmarking with iPinYou dataset, arXiv.

Run it on a trickle of traffic

Typical A/B testing scenario in online systems

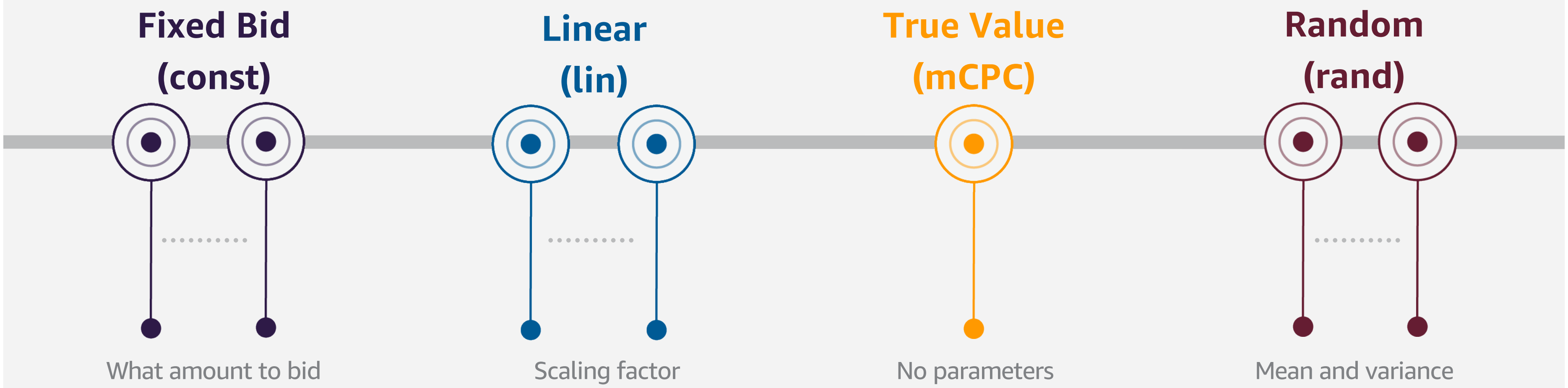
Modern ad servers support A/B test where a small trickle of traffic sees a new bidding behavior

- Requires ad server to implement all strategies
- Too many options means longer tests and larger errors
- Typical A/B test can run for weeks
- How do we know which is best? By which metric?

amazonadvertising

Should we A/B test all parameters of bidding strategies

Might be nice, but there are too many parameters to test online



[21] Weinan Zhang, Shuai Yuan, Jun Wang, and Xuehua Shen. 2014. Real-time bidding benchmarking with iPinYou dataset, arXiv.

Sounds like a job for multi-arm bandits

Bet you couldn't guess that was coming.

Traditional multi-arm bandits

Scalar-value setting

Reward distribution not known to the learner. It is observed only after each pull

- Seek to maximize total long-term value
- Explore-exploit
- Explore more at first
- Exploit more at the end

Notation

K arms $\{1, \ldots, K\}$

ν_i scalar-valued reward distribution

$$\mu_i = \mathbb{E}_{X \sim \nu_i} X$$

I_t arm pulled at each round, t

$X_t \sim \nu_{I_t}$ reward observed after each pull

amazonadvertising

Pure-exploration bandits

See [11, 15]

Explore until a fixed, predetermined stopping criteria

- Fixed confidence [1, 3, 10, 12, 14]
- Fixed budget [3, 13]
- Explore only
- No exploitation

Pure-exploration tasks

Find the one arm that maximizes μ_i after B tries

Find all arms with mean larger than a threshold, with fixed confidence

amazonadvertising

For references, see full paper.

Real A/B testing, revisited

Never heard of pure-exploration bandits? Maybe you have

Real-world A/B testing is similar to Uniform Allocation (UA) pure-exploration bandits

- Allocate a fixed portion of traffic to each treatment
- Do not try to auto-adjust allocation (No Exploit)
- Run until B days (Fixed Budget)
- Or, run until T power (Fixed Confidence)

Acceptance criteria for A/B Tests

Determine which treatment is feasible, by meeting all criteria

+CTR

Increase Click Through Rate (CTR)

Prefer bidding strategies that surfaces ads which will lead to clicks

-CPM

Reduce Cost per Impression (CPM)

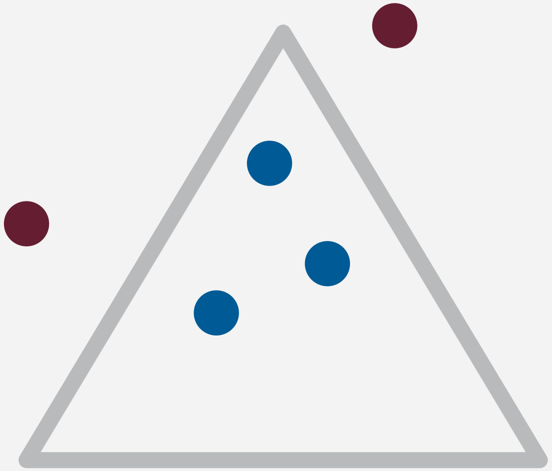
Prefer bidding strategies that are cheaper for advertisers

Feasible arm identification

Bandits which conform to multiple criteria

Constrained optimization for multi-armed bandits, find the best arm which is feasible.

- Multiple acceptable criteria == feasible polyhedron
- Are arms within the polyhedron or not?
- How to select arms to pull?
- Do we really need ALL the feasible arms?



Feasible polyhedron, which arm should we pull next?

amazonadvertising

Feasible arm identification

Pure exploration within fixed budget setting [13]

Now, each arm has a vector-valued reward distribution.

- Vector-valued rewards
- Feasibility == rewards within a polyhedron
- Still fixed budget
- Still explore only, no exploitation

Notation

K arms $\{1, \ldots, K\}$

ν_i D-dimensional **vector**-valued reward distribution

$$\mu_i = \mathbb{E}_{X \sim \nu_i} X$$

I_t arm pulled at each round, t

$X_t \sim \nu_{I_t}$ reward observed after each pull

$$P = \{\boldsymbol{x} \in \mathbb{R}^D : A\boldsymbol{x} \leqslant b\}$$

Feasible region, polyhedron

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

MD-APT

Given a fixed tolerance and budget, find all feasible arms.

Pull arms which are likely to be closest to boundary

- Still uncertain about arms on the boundary
- Tolerance defines fineness of boundary
- Bounds depend on distance of all arms to boundary

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

1: **Input:** tolerance ϵ
2: Initialize by pulling each arm once
3: **for** $t = K+1, \ldots, T$ **do**
4: Choose $I_t = \arg\min_i [\text{dist}(\hat{\boldsymbol{\mu}}_{i,t}, \partial P) + \epsilon]\sqrt{T_i(t)}$ and sample $\boldsymbol{X}_t \sim \nu_{I_t}$.
5: **return** $\hat{S} = \{i \in [K] : \hat{\boldsymbol{\mu}}_{i,T_i(t+1)} \in P\}$

$$\text{dist}(\hat{\boldsymbol{\mu}}_{i,t}, \partial P)$$

Distance to boundary of polyhedron, P

$$\text{Define } H_\epsilon = \sum_i [\text{dist}(\boldsymbol{\mu}_i, \partial P) + \epsilon]^{-2}$$

Let $K \geq 0$, $T \geq 2K$, and $\epsilon \geq 0$. Then, with probability at least

$$\Omega(1 - (\log(T) + 1)K5^D \exp(-\frac{T}{R^2 H_\epsilon})),$$

MD-APT(ϵ) returns $\hat{S}$ such that

- *if dist($\boldsymbol{\mu}_i, P^c$) $\geq \epsilon$, then $i \in \hat{S}$ and*
- *if dist($\boldsymbol{\mu}_i, P$) $\geq \epsilon$, then $i \notin \hat{S}$.*

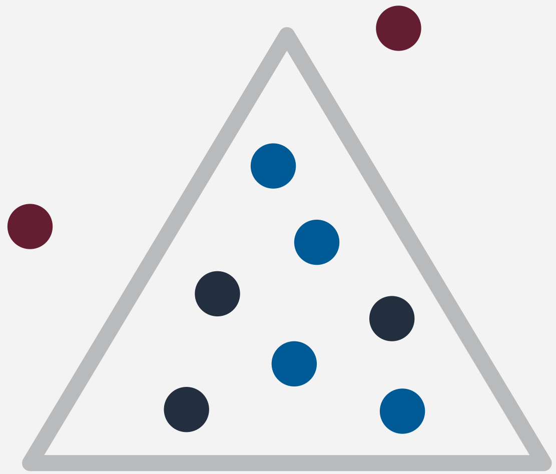
See [13] for details on algorithm and convergence bounds

Any-*m* feasible arm identification

Setting discussed in this paper

Could it be faster to find only some feasible arms, but there are several edge cases

- If there are fewer than *m* feasible arms, return all
- Could we just end the full search earlier?



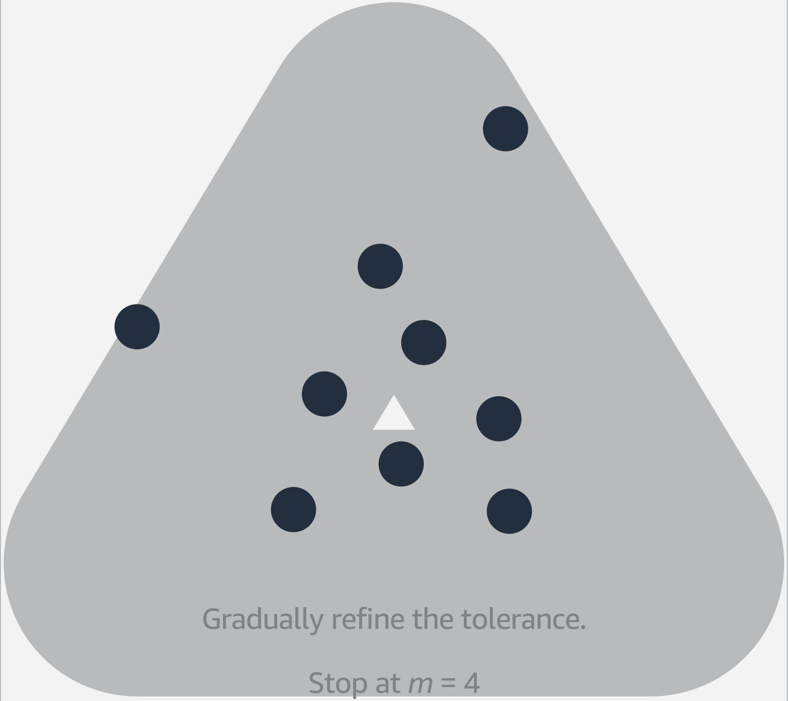
Do we really need to know if every arm is feasible? Maybe only $m = 4$ will do.

Any-*m* feasible arm identification

Example: Iteration 1

Allocate smaller budgets to MD-APT and gradually reduce tolerance

- Focus less on looking just around the boundary
- Quickly enumerates some feasible arms
- Can it guarantee returning at least *m*?



amazonadvertising

Any-*m* feasible arm identification

Example: Iteration 2

Allocate smaller budgets to MD-APT and gradually reduce tolerance

- Focus less on looking just around the boundary
- Quickly enumerates some feasible arms
- Can it guarantee returning at least *m*?

Stop at $m = 4$

amazonadvertising

Any-*m* feasible arm identification

Example: Iteration 3

Allocate smaller budgets to MD-APT and gradually reduce tolerance

- Focus less on looking just around the boundary
- Quickly enumerates some feasible arms
- Can it guarantee returning at least *m*?

Gradually refine the tolerance.

If $m = 3$, we would be done

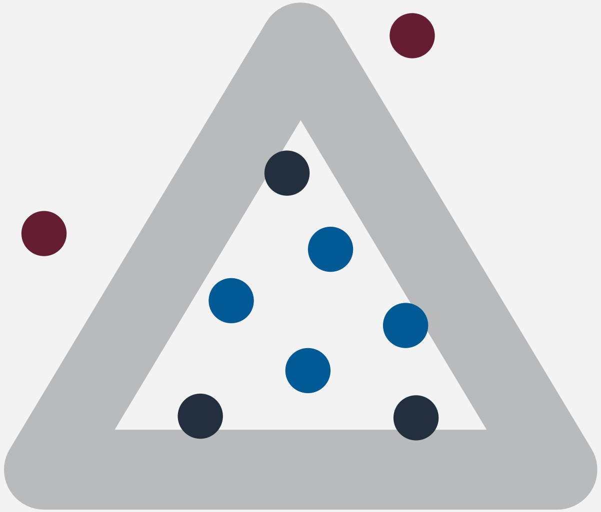
amazonadvertising

Any-*m* feasible arm identification

Example: Iteration 4

Allocate smaller budgets to MD-APT and gradually reduce tolerance

- Focus less on looking just around the boundary
- Quickly enumerates some feasible arms
- Can it guarantee returning at least *m*?



Since m = 4, we are done

MD-APT-ANY

Given a fixed tolerance and budget, find *m* feasible arms.

Sequentially calls MD-APT on fixed budget to trace out arms

- Call MD-APT with decreasing tolerance
- Checks whether MD-APT found at least *m*-feasible arms
- Bounds depend on worst-case infeasible arm
- Returns all feasible arms if there are less than *m*

Algorithm 2 MD-APT-ANY

```
Input: tolerance ϵ
Define ϵ_r := B/2^r and t_r = r T/⌈log_2(B/ϵ)⌉
for r = 1, ..., ⌈log_2(B/ϵ)⌉ do
    Run MD-APT(ϵ_r) for T/⌈log_2(B)⌉ iterations.
    Ŝ_r = {i ∈ [K] : μ̂_{i,T_i(t_r+1)} ∈ P and dist(μ̂_{i,T_i(t_r+1)}, ∂P) ≥ ϵ_{r−1}}
    if |Ŝ_r| ≥ m then
        Ŝ := argmax_{Z⊂Ŝ_r, |Z|=m} Σ_{i∈Z} dist(μ̂_{i,T_i(t_r+1)}, ∂P)
        Return Ŝ
Pick any Ŝ ⊂ {i ∈ [K] : dist(μ̂_{i,T_i(T+1)}, P) ≤ ϵ} such that |Ŝ| = min(m, |{i ∈ [K] : dist(μ̂_{i,T_i(T+1)}, P) ≤ ϵ}|).
Return Ŝ
```

Define

$$H_m = \sum_{i \in [K]} \max(\Gamma_m, \ \mathrm{dist}(\boldsymbol{\mu}_i, \partial P))^{-2}$$

$$\Gamma_m = \max\Big(\max_{i:\boldsymbol{\mu}_i \in P}^{(m)} \ \mathrm{dist}(\boldsymbol{\mu}_i, \partial P), 0\Big)$$

Let $\epsilon > 0$. With probability at least

$$\Omega\Big(1 - \log_2\Big(\frac{B}{\epsilon}\Big)\log(T) K 5^D \exp\Big(-\frac{T}{\log_2(\frac{B}{\epsilon}) H_m R^2}\Big)\Big),$$

MD-APT-ANY(ϵ) outputs $\hat{S}$ such that

- *if $\epsilon < \frac{\Gamma_m}{2}$, then $|\hat{S}| = m$ and $\hat{S} \subset \{i \in [K] : \boldsymbol{\mu}_i \in P\}$;*
- *otherwise,*

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

See full paper for proof of the theorems

MD-APT-ANY-F

Heuristic improvement

Sample arms within border a little more on odd rounds

- Focus more on quasi-feasible arms
- Double-check that they are still feasible

Algorithm 3

1. On even rounds, run

MD-APT-ANY

2. On odd rounds, sample the top m arms by:

$$\text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(t)}, P^c) - \text{dist}(\widehat{\boldsymbol{\mu}}_{i,T_i(t)}, P).$$

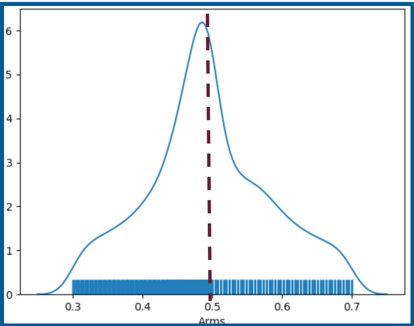
amazonadvertising

Inspired by hyperparameter optimization algorithms like Hyperband

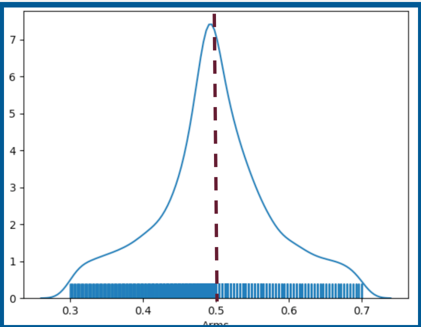
Comparing arm pull distribution

MD-APT-ANY pulls arms which are likely to be feasible more often than other algorithms

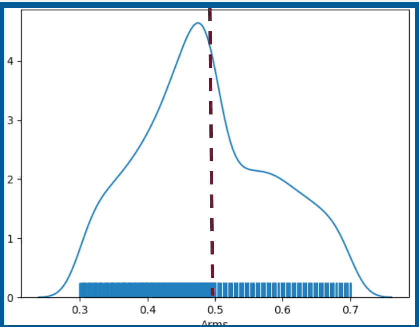
Arm pulls on synthetic experiment E2, feasible means are <= 0.5 (left of line below)



MD-SAR [13]



MD-APT [13]



MD-APT-ANY

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

amazonadvertising

Bidding strategies data

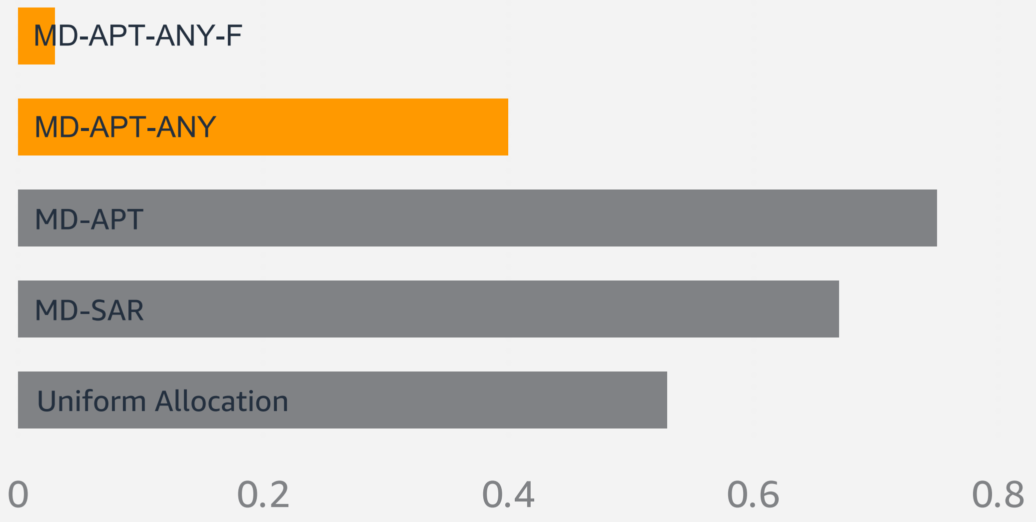
From iPinYou RTB bidding competition [21]

- K = 144 arms, bidding strategies and their parameters
- M = 1, want to find just one good strategy
- D = 2, two rewards:
  - CTR > default strategy
  - Advertiser CPM < default strategy

[21] Weinan Zhang, Shuai Yuan, Jun Wang, and Xuehua Shen. 2014. Real-time bidding benchmarking with iPinYou dataset, arXiv.

Results on bidding strategies

Compares probability of pulling an infeasible arm



amazonadvertising

Results on other public datasets

Comparing MD-APT-ANY against other state-of-the-art feasible arm identification methods

	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)

	E3 (m = 5)	E3 (m = 15)	E3 (m = 30)	Med 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)	0.18 (0.03)
MD-APT-ANY	0.29 (0.03)	0.47 (0.04)	0.36 (0.03)	0.13 (0.00)	0.32 (0.03)
MD-APT	0.36 (0.03)	0.33 (0.03)	0.40 (0.03)	0.21 (0.01)	0.62 (0.03)
MD-SAR	0.35 (0.03)	0.40 (0.03)	0.89 (0.02)	0.14 (0.00)	0.44 (0.04)
UA	0.84 (0.03)	0.82 (0.03)	0.92 (0.02)	0.13 (0.00)	0.42 (0.03)

E1, E2, E3: Synthetic **Med E: Drug dosage** **Crowd E: Effective of crowdsourcing workers**

See paper for details on these experiments

Hypothetical A/B testing timeline with pure-exploration bandits

Fixed-budget any-m feasible arm identification

MD-APT-ANY
Test ~100 parameters

MD-APT
Test ~10 parameters

A/B Test
Test ~3 parameters in full A/B test

Launch
Fine tune with lots of parameters and MD-APT-ANY

Traditional Bandits
Long-running explore / exploit methods on a small set of parameters

Conclusion

1 Real A/B tests fit well within pure-exploration feasible-arm identification bit learning

2 Any-*m* feasible arm identification returns a few good parameters within budget

3 Bidding strategies and their parameters fit within this scheme, as shown in results

4 Apply to other domains such as hyperparameter optimization

"Join us today!
We're Hiring in
Palo Alto, CA, USA"

Contact abagher@amazon.com

See our listings on Whova app

amazonadvertising