Feasible Bidding Strategies through Pure-Exploration Bandits

Julian Katz-Samuels, Abraham Bagherjeirian (*)
(*) Amazon Advertising – Sponsored Brands, Palo Alto, CA
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
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 $p(\text{Action}) \times V(\text{Action})$.
- Linear Scaling (lin): Scale mCPC by a constant, such as historical $p(\text{Action})$.
- Random (rand): Bid a random value.

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?
Should we A/B test all parameters of bidding strategies

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

- **Fixed Bid (const)**
  - What amount to bid

- **Linear (lin)**
  - Scaling factor

- **True Value (mCPC)**
  - No parameters

- **Random (rand)**
  - Mean and variance

---

Sounds like a job for multi-arm bandits
**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 $\{1, \ldots, K\}$
- $\nu_i$ scalar-valued reward distribution
  \[ \mu_i = \mathbb{E}_{X \sim \nu_i} X \]
- $l_t$ arm pulled at each round, $t$
- $X_t \sim \nu_{l_t}$ reward observed after each pull
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 \( \mu_i \) after \( B \) tries

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

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

Increase Click Through Rate (CTR)

Prefer bidding strategies that surfaces ads which will lead to clicks

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?
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 \(\{1, \ldots, K\}\)

$\nu_i$ \(D\)-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

$$P = \{ \mathbf{x} \in \mathbb{R}^D : A\mathbf{x} \leq b \}$$

Feasible region, polyhedron
**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 $\epsilon$
2. Initialize by pulling each arm once
3. **for** $t = K + 1, \ldots, T$ **do**
4. Choose $l_t = \arg \min_i [\text{dist}(\hat{\mu}_i, \partial P) + \epsilon] \sqrt{T_i(t)}$ and sample $X_t \sim \nu_h$.
5. **return** $\hat{S} = \{i \in [K] : \hat{\mu}_i, T_i(t+1) \in P\}$

### Distance to boundary of polyhedron, $P$

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

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

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

MD-APT($\epsilon$) returns $\hat{S}$ such that

- if $\text{dist}(\mu_i, P^c) \geq \epsilon$, then $i \in \hat{S}$ and
- if $\text{dist}(\mu_i, P) \geq \epsilon$, 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 \(m\)?

Gradually refine the tolerance.
Stop at \(m = 4\)
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 \( m \)?
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.
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**

1. **Input:** tolerance $\epsilon$
2. Define $\epsilon_r := \frac{B}{2^r}$ and $t_r = r \left\lceil \frac{T}{\log_2(B)} \right\rceil$
3. for $r = 1, \ldots, \lceil \log_2(B) \rceil$ do
4. Run MD-APT($\epsilon_r$) for $\frac{T}{\log_2(B)}$ iterations.
5. $\hat{S}_r = \{ i \in [K] : \hat{\mu}_i, t_i(t_i+1) \in P \text{ and } \text{dist}(\hat{\mu}_i, t_i(t_i+1), \partial P) \geq \epsilon_{r-1} \}$
6. if $|\hat{S}_r| > m$ then
7. $\hat{S} = \arg \max_{Z \subseteq S_r, |Z| = m} \sum_{i \in Z} \text{dist}(\hat{\mu}_i, t_i(t_i+1), \partial P)$
8. Return $\hat{S}$
9. Pick any $\hat{S} \subset \{ i \in [K] : \text{dist}(\hat{\mu}_i, t_i(t_i+1), P) \leq \epsilon \}$ such that $|\hat{S}| = \min(m, |\{ i \in [K] : \text{dist}(\hat{\mu}_i, t_i(t_i+1), P) \leq \epsilon \}|)$.
10. Return $\hat{S}$

Define $H_m = \sum_{i \in [K]} \max(\Gamma_m, \text{dist}(\mu_i, \partial P))^{-2}$

$$\Gamma_m = \max_{i: \mu_i \in P} \max_{i: \mu_i \in P} \text{dist}(\mu_i, \partial P), 0$$

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

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

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

- if $\epsilon < \frac{B}{2^r}$, then $|\hat{S}| = m$ and $\hat{S} \subset \{ i \in [K] : \mu_i \in P \}$;
- otherwise,

$$\min(|\{ i \in [K] : \mu_i \in P \}|, m) \leq |\hat{S}| \text{ and } \forall i \in \hat{S} : \text{dist}(\mu_i, P) \leq 2\epsilon$$

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:

   $$\text{dist}(\hat{\mu}_i, T_i(t), P^c) - \text{dist}(\hat{\mu}_i, T_i(t), P).$$
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)

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:
  - $\text{CTR} > \text{default strategy}$
  - Advertiser CPM $< \text{default strategy}$

Results on bidding strategies

Compares probability of pulling an infeasible arm

**MD-APT-ANY-F**

**MD-APT-ANY**

**MD-APT**

**MD-SAR**

**Uniform Allocation**

0  0.2  0.4  0.6  0.8
Results on other public datasets

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

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

<table>
<thead>
<tr>
<th></th>
<th>E3 (m = 5)</th>
<th>E3 (m = 15)</th>
<th>E3 (m = 30)</th>
<th>Med E (m = 1)</th>
<th>Crowd E (m = 5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MD-APT-ANY-F</td>
<td>0.20 (0.03)</td>
<td>0.23 (0.03)</td>
<td>0.35 (0.03)</td>
<td>0.11 (0.00)</td>
<td>0.18 (0.03)</td>
</tr>
<tr>
<td>MD-APT-ANY</td>
<td>0.29 (0.03)</td>
<td>0.47 (0.04)</td>
<td>0.36 (0.03)</td>
<td>0.13 (0.00)</td>
<td>0.32 (0.03)</td>
</tr>
<tr>
<td>MD-APT</td>
<td>0.36 (0.03)</td>
<td>0.33 (0.03)</td>
<td>0.40 (0.03)</td>
<td>0.21 (0.01)</td>
<td>0.62 (0.03)</td>
</tr>
<tr>
<td>MD-SAR</td>
<td>0.35 (0.03)</td>
<td>0.40 (0.03)</td>
<td>0.89 (0.02)</td>
<td>0.14 (0.00)</td>
<td>0.44 (0.04)</td>
</tr>
<tr>
<td>UA</td>
<td>0.84 (0.03)</td>
<td>0.82 (0.03)</td>
<td>0.92 (0.02)</td>
<td>0.13 (0.00)</td>
<td>0.42 (0.03)</td>
</tr>
</tbody>
</table>

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
Test ~10 parameters

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

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

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

MD-APT-ANY
Test ~100 parameters
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