Learning to Bid in Repeated First-price Auctions



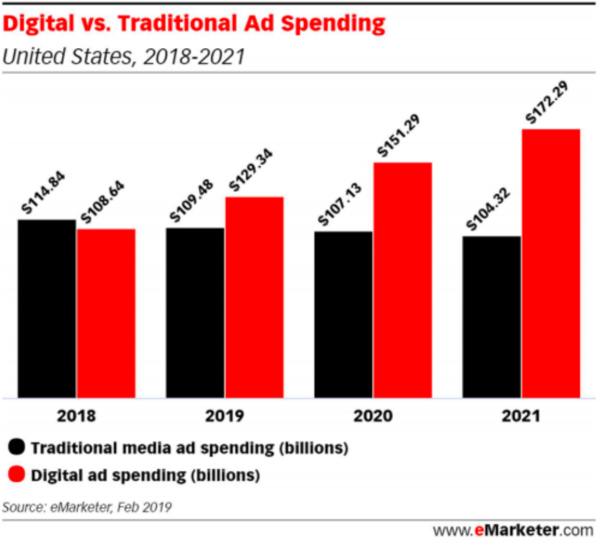


Yanjun Han
(UC Berkeley)

Tsachy Weissman (Stanford), Zhengyuan Zhou (NYU),
Aaron Flores & Erik Ordentlich (Yahoo! Research)

TOPS Seminar
Department of Technology, Operations, and Statistics
NYU Stern School of Business

Success of digital ads



Online auctions

Advertiser (bidder) Ad exchange Publisher (seller)

Online auctions

Advertiser (bidder) Ad exchange Publisher (seller)

Some popular auction designs:

- second-price auction: the bidder with the highest bid wins the auction, and pays the price equal to the second highest bid
- first-price auction: the bidder with the highest bid wins the auction, and pays the price equal to the highest bid

From second-price to first-price

There is a recent industrial shift to first-price auctions for display ads:

AppNexus
A Xandr Company

INDEX EXCHANGE

OpenX

Google

From second-price to first-price

There is a recent industrial shift to first-price auctions for display ads:

- greater transparency to bidders
- enhanced monetization for sellers
- preferable mechanism for header-bidding

More recent news

Google AdSense (contextual ads):

ADSENSE

Moving AdSense to a first-price auction

Oct 07, 2021 · 1 min read

Matt Wong
Product Manager

Share

Source: https://blog.google/products/adsense/our-move-to-a-first-price-auction/

Bidder's challenge

How to bid in first-price auctions where it is no longer optimal to bid truthfully?

Bidder's challenge

How to bid in first-price auctions where it is no longer optimal to bid truthfully?

Optimal bid in first-price auction:

$$b^\star = \arg\max_b \quad (\underbrace{v}_{\text{private value}} - b) \cdot \mathbb{P}(b \geq \underbrace{m}_{\text{others' maximum bid}})$$

Bidder's challenge

How to bid in first-price auctions where it is no longer optimal to bid truthfully?

Optimal bid in first-price auction:

$$b^\star = \arg\max_b \quad (\underbrace{v}_{\text{private value}} - b) \cdot \mathbb{P}(b \geq \underbrace{m}_{\text{others' maximum bid}})$$

- unknown bid distribution: need to learn $\mathbb{P}(b \geq m)$
- censored feedback: cannot directly observe m
- non-stationary environment: $\mathbb{P}_t(b \geq m)$ depends on t

An example strategy

AppNexus whitepaper 2018:

The available evidence suggests that many large buyers have yet to adjust their bidding behavior for first-price auctions.

Source: `https://www.appnexus.com/sites/default/files/whitepapers/49344-CM-Auction-Type-Whitepaper-V9.pdf`

An example strategy

AppNexus whitepaper 2018:

The available evidence suggests that many large buyers have yet to adjust their bidding behavior for first-price auctions.

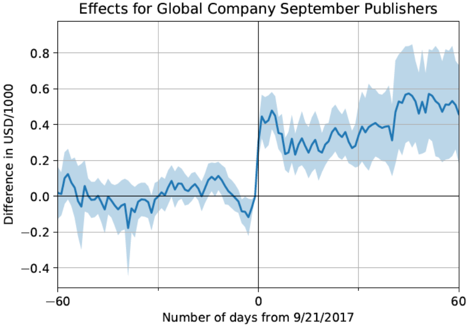
A suggested strategy in the whitepaper:

- The buyer starts by shading her bid by 20% of her valuation.
- If the buyer wins and has never lost, she reduces her bid by another 10% from her initial valuation.
- Once the buyer loses for the first time, she would increase her bid by 8% from her initial valuation.
- If the buyer wins a round but has also lost before, she reduces her bid by 4% from her initial valuation.
- If the buyer loses twice or more in a row, she increases her bid by 10%, up to 99% higher than her valuation.

Source: `https://www.appnexus.com/sites/default/files/whitepapers/49344-CM-Auction-Type-Whitepaper-V9.pdf`

Empirical study

[Goke et al. 2021]: "at least a subset of bidders responded suboptimally to the format change"



Our target

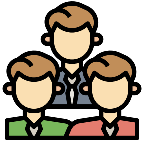
Provide sound theoretical guidelines and timely practical solutions to bidders

Model and Main Results

Bidder's sequential decision model

private source

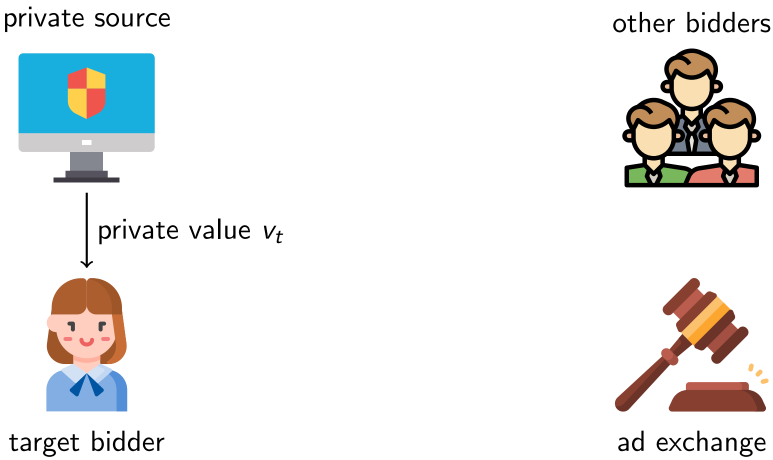
other bidders



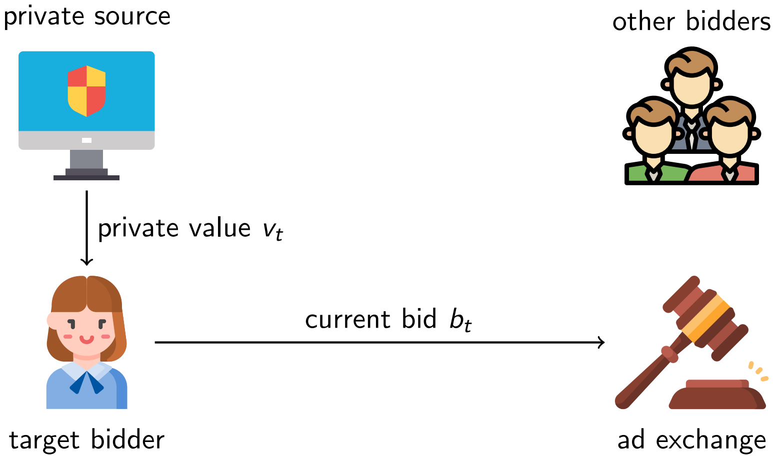
target bidder

ad exchange

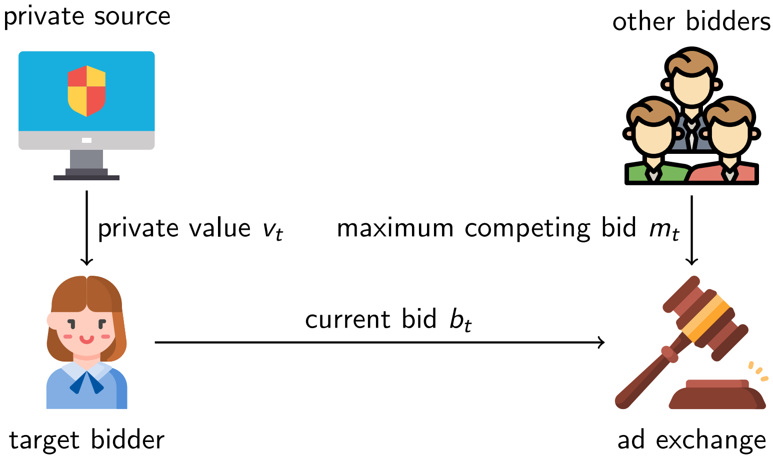
Bidder's sequential decision model



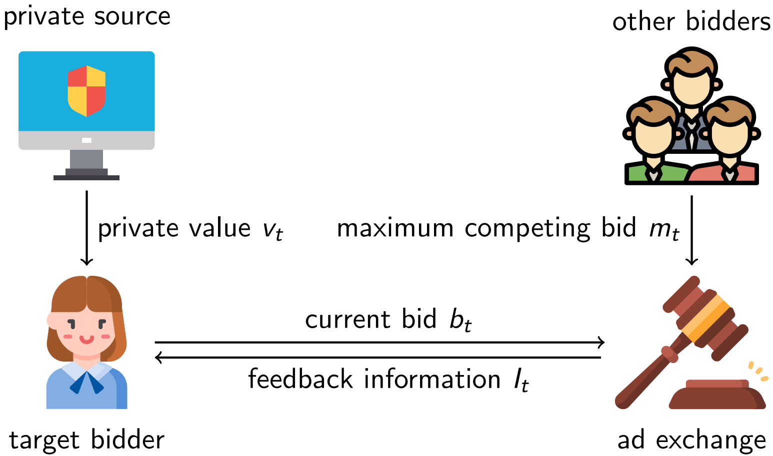
Bidder's sequential decision model



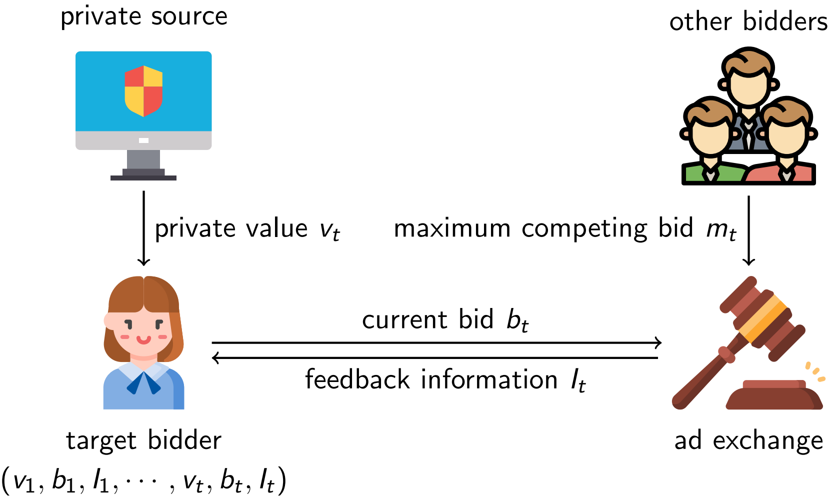
Bidder's sequential decision model



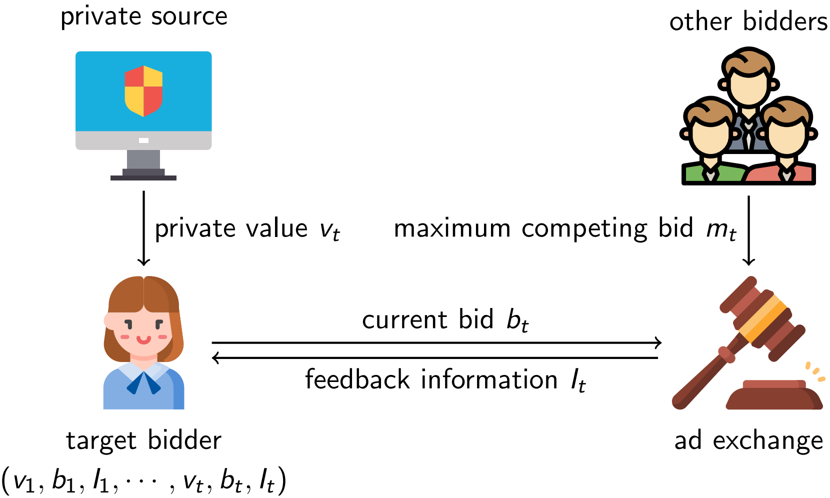
Bidder's sequential decision model



Bidder's sequential decision model



Bidder's sequential decision model



$$v_t, b_t, m_t \in [0, 1]$$

Instantaneous reward: $r(b_t; v_t, m_t) = (v_t - b_t) \cdot \mathbb{1}(b_t \geq m_t)$

Model assumption: feedback

- Unobservable bids: the bidder only knows whether he/she wins or not, i.e.

$$I_t = \mathbb{1}(b_t \geq m_t)$$

Model assumption: feedback

- Unobservable bids: the bidder only knows whether he/she wins or not, i.e.

$$I_t = \mathbb{1}(b_t \geq m_t)$$

- Censored bids: others' bids are left- or right-censored:

$$I_t = \max\{b_t, m_t\} \qquad \text{(winning price is announced)}$$
$$I_t = \min\{b_t, m_t\} \qquad \text{(feedback inherited from SPA)}$$

Model assumption: feedback

- Unobservable bids: the bidder only knows whether he/she wins or not, i.e.

$$I_t = \mathbb{1}(b_t \geq m_t)$$

- Censored bids: others' bids are left- or right-censored:

$$I_t = \max\{b_t, m_t\} \qquad \text{(winning price is announced)}$$
$$I_t = \min\{b_t, m_t\} \qquad \text{(feedback inherited from SPA)}$$

- Observable bids: the bidder always knows the minimum bid to win, i.e.

$$I_t = m_t$$

Model assumption: values and bids

- Stochastic setting: $m_t \overset{\text{i.i.d.}}{\sim} G$ with unknown CDF $G(\cdot)$
  - falls into standard learning framework
  - no additional assumption on G
  - reasonable in a short time window, or with irrelevant competitors

Model assumption: values and bids

- Stochastic setting: $m_t \overset{\text{i.i.d.}}{\sim} G$ with unknown CDF $G(\cdot)$
  - falls into standard learning framework
  - no additional assumption on G
  - reasonable in a short time window, or with irrelevant competitors
- Adversarial setting: m_t is an adversarial sequence
  - no distributional assumption
  - allows for others' strategic or even adversarial moves

Model assumption: values and bids

- Stochastic setting: $m_t \overset{\text{i.i.d.}}{\sim} G$ with unknown CDF $G(\cdot)$
  - falls into standard learning framework
  - no additional assumption on G
  - reasonable in a short time window, or with irrelevant competitors
- Adversarial setting: m_t is an adversarial sequence
  - no distributional assumption
  - allows for others' strategic or even adversarial moves
- Private value v_t always assumed to be known and adversarial

Bidder's target: regret

Regret of a bidding policy $\pi = (b_t)_{t=1}^{T}$:

$$R_T(\pi) \triangleq \underbrace{\max_{f \in \mathcal{F}} \mathbb{E}\left[\sum_{t=1}^{T} r(f(v_t); v_t, m_t)\right]}_{\text{oracle's reward}} - \underbrace{\mathbb{E}\left[\sum_{t=1}^{T} r(b_t; v_t, m_t)\right]}_{\text{bidder's reward}}$$

Bidder's target: regret

Regret of a bidding policy $\pi = (b_t)_{t=1}^T$:

$$R_T(\pi) \triangleq \underbrace{\max_{f \in \mathcal{F}} \mathbb{E}\left[\sum_{t=1}^{T} r(f(v_t); v_t, m_t)\right]}_{\text{oracle's reward}} - \underbrace{\mathbb{E}\left[\sum_{t=1}^{T} r(b_t; v_t, m_t)\right]}_{\text{bidder's reward}}$$

- stochastic setting: $\mathcal{F} = \{\text{all functions}\}$

$$R_T(\pi) \triangleq \sum_{t=1}^{T} \left(\underbrace{\max_{b_t^\star} (v_t - b_t^\star) G(b_t^\star)}_{\text{oracle's reward}} - \underbrace{\mathbb{E}[(v_t - b_t) G(b_t)]}_{\text{bidder's reward}} \right).$$

Bidder's target: regret

Regret of a bidding policy $\pi = (b_t)_{t=1}^T$:

$$R_T(\pi) \triangleq \underbrace{\max_{f \in \mathcal{F}} \mathbb{E}\left[\sum_{t=1}^T r(f(v_t); v_t, m_t)\right]}_{\text{oracle's reward}} - \underbrace{\mathbb{E}\left[\sum_{t=1}^T r(b_t; v_t, m_t)\right]}_{\text{bidder's reward}}$$

- stochastic setting: $\mathcal{F} = \{\text{all functions}\}$

$$R_T(\pi) \triangleq \sum_{t=1}^T \left(\underbrace{\max_{b_t^\star} (v_t - b_t^\star) G(b_t^\star)}_{\text{oracle's reward}} - \underbrace{\mathbb{E}[(v_t - b_t) G(b_t)]}_{\text{bidder's reward}} \right).$$

- adversarial setting: $\mathcal{F} = \mathcal{F}_{\text{Lip}} = \{\text{all 1-Lipschitz functions}\}$

$$R_T(\pi) \triangleq \underbrace{\max_{f \in \mathcal{F}_{\text{Lip}}} \sum_{t=1}^T r(f(v_t); v_t, m_t)}_{\text{oracle's reward}} - \underbrace{\mathbb{E}\left[\sum_{t=1}^T r(b_t; v_t, m_t)\right]}_{\text{bidder's reward}}.$$

Some key features

- non-linear reward with continuous action
  - $r(b; v, m) = (v - b) \cdot 1(b \geq m)$ not linear nor concave in b
  - a challenging problem in bandits, where UCB or Thompson sampling does not directly work

Some key features

- non-linear reward with continuous action
  - $r(b; v, m) = (v - b) \cdot 1(b \geq m)$ not linear nor concave in b
  - a challenging problem in bandits, where UCB or Thompson sampling does not directly work
- minimal assumptions on v_t and m_t
  - no structural assumptions such as smoothness or log-concavity

Some key features

- non-linear reward with continuous action
  - $r(b; v, m) = (v - b) \cdot 1(b \geq m)$ not linear nor concave in b
  - a challenging problem in bandits, where UCB or Thompson sampling does not directly work
- minimal assumptions on v_t and m_t
  - no structural assumptions such as smoothness or log-concavity
- censored feedback
  - interesting interplay between feedback structure and reward function

Some key features

- non-linear reward with continuous action
  - $r(b; v, m) = (v - b) \cdot 1(b \geq m)$ not linear nor concave in b
  - a challenging problem in bandits, where UCB or Thompson sampling does not directly work
- minimal assumptions on v_t and m_t
  - no structural assumptions such as smoothness or log-concavity
- censored feedback
  - interesting interplay between feedback structure and reward function
- strong time-variant oracle
  - competing with a meaningful and powerful benchmark

Table of optimal regrets

Setting / Feedback	stochastic	adversarial
Unobservable		
Censored		
Observable		

Table of optimal regrets

Feedback \ Setting	stochastic	adversarial
Unobservable	$T^{2/3}$	$T^{3/4}$
Censored		
Observable		

- unobservable case implied by [Balseiro et al. 2019]

Table of optimal regrets

Feedback \ Setting	stochastic	adversarial
Unobservable	$T^{2/3}$	$T^{3/4}$
Censored		
Observable	$\sqrt{T}$	

- unobservable case implied by [Balseiro et al. 2019]

Table of optimal regrets

Feedback \ Setting	stochastic	adversarial
Unobservable	$T^{2/3}$	$T^{3/4}$
Censored	$\sqrt{T}$	
Observable	$\sqrt{T}$	$\sqrt{T}$

- unobservable case implied by [Balseiro et al. 2019]

Table of optimal regrets

Feedback \ Setting	stochastic	adversarial
Unobservable	$T^{2/3}$	$T^{3/4}$
Censored	$\sqrt{T}$	
Observable	$\sqrt{T}$	$\sqrt{T}$

- unobservable case implied by [Balseiro et al. 2019]
- all terms within polylog(T) factors
- real-data experiments for the adversarial observable setting

Table of optimal regrets

Feedback \ Setting	stochastic	adversarial
Unobservable	$T^{2/3}$	$T^{3/4}$
Censored	$\sqrt{T}$	open
Observable	$\sqrt{T}$	$\sqrt{T}$

- unobservable case implied by [Balseiro et al. 2019]
- all terms within polylog(T) factors
- real-data experiments for the adversarial observable setting

Part I: Stochastic Auctions with Censored Feedback

Zhengyuan Zhou
NYU Stern

Tsachy Weissman
Stanford EE

"Optimal No-regret Learning in Repeated First-price Auctions"
arXiv: 2003.09795

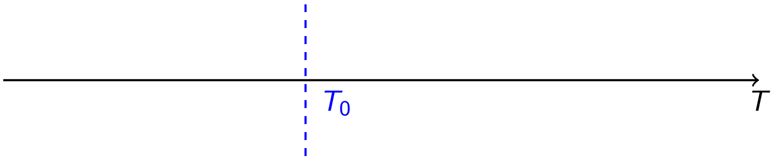
A first trial

left censoring: whenever the bidder wins the auction (exploitation), he/she loses the information for learning (exploration)

A first trial

left censoring: whenever the bidder wins the auction (exploitation), he/she loses the information for learning (exploration)

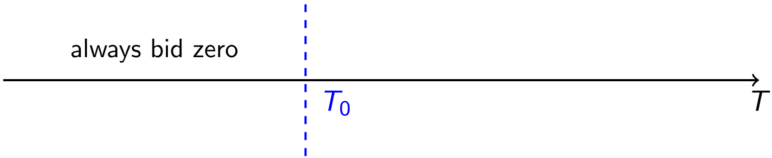
Explore-then-commit (ETC):



A first trial

left censoring: whenever the bidder wins the auction (exploitation), he/she loses the information for learning (exploration)

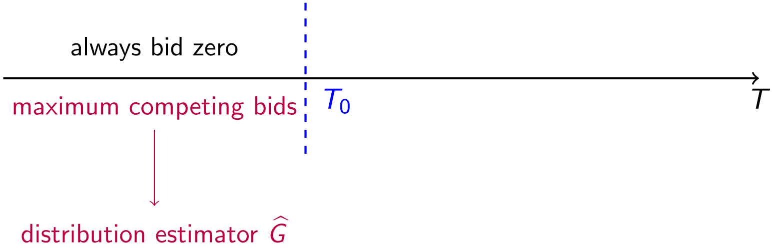
Explore-then-commit (ETC):



A first trial

left censoring: whenever the bidder wins the auction (exploitation), he/she loses the information for learning (exploration)

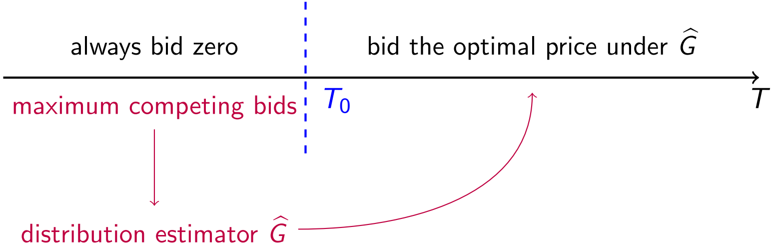
Explore-then-commit (ETC):



A first trial

left censoring: whenever the bidder wins the auction (exploitation), he/she loses the information for learning (exploration)

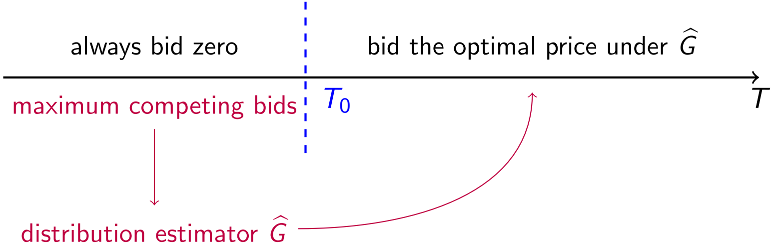
Explore-then-commit (ETC):



A first trial

left censoring: whenever the bidder wins the auction (exploitation), he/she loses the information for learning (exploration)

Explore-then-commit (ETC):



Regret analysis:

$$\text{regret of } \pi^{\text{ETC}} = O\left(T_0 + \frac{T}{\sqrt{T_0}}\right) \overset{T_0 \sim T^{2/3}}{=} O(T^{2/3})$$

Monotone Feedback and Monotone Successive Elimination

Contextual multi-armed bandit

- context (state): private value
- arm (action): bidder's bid
- reward: the bidder receives a random reward depending on both the bidding price (action) and the private value (context)

Contextual multi-armed bandit

- context (state): private value
- arm (action): bidder's bid
- reward: the bidder receives a random reward depending on both the bidding price (action) and the private value (context)

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3	✓						
Price 4							
Price 5				✓			
...							
Price K			✓				

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							✓
Price 2		✓					
Price 3							
Price 4							
Price 5					✓		
...							
Price K							

environment under private value #2

Contextual multi-armed bandit

- context (state): private value
- arm (action): bidder's bid
- reward: the bidder receives a random reward depending on both the bidding price (action) and the private value (context)

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3	✓						
Price 4							
Price 5				✓			
...							
Price K			✓				

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							✓
Price 2		✓					
Price 3							
Price 4							
Price 5					✓		
...							
Price K							

environment under private value #2

Under bandit feedback, the optimal regret is $\Theta(\sqrt{\#\text{context} \cdot \#\text{action} \cdot T})$.

Monotone feedback

Monotone feedback: each bid provides full information for all larger bids and all private values

- if bidder wins, then any larger bid wins too
- if bidder loses, then others' maximum bid is perfectly observed

Monotone feedback

Monotone feedback: each bid provides full information for all larger bids and all private values

- if bidder wins, then any larger bid wins too
- if bidder loses, then others' maximum bid is perfectly observed

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3	✓						
Price 4							
Price 5							
...							
Price K							

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3							
Price 4							
Price 5							
...							
Price K							

environment under private value #2

Monotone feedback

Monotone feedback: each bid provides full information for all larger bids and all private values

- if bidder wins, then any larger bid wins too
- if bidder loses, then others' maximum bid is perfectly observed

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #2

Monotone feedback

Monotone feedback: each bid provides full information for all larger bids and all private values

- if bidder wins, then any larger bid wins too
- if bidder loses, then others' maximum bid is perfectly observed

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2		✓					
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #2

Monotone feedback

Monotone feedback: each bid provides full information for all larger bids and all private values

- if bidder wins, then any larger bid wins too
- if bidder loses, then others' maximum bid is perfectly observed

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2		✓					
Price 3	✓	✓					
Price 4	✓	✓					
Price 5	✓	✓					
...	✓	✓					
Price K	✓	✓					

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2		✓					
Price 3	✓	✓					
Price 4	✓	✓					
Price 5	✓	✓					
...	✓	✓					
Price K	✓	✓					

environment under private value #2

Monotone feedback

Monotone feedback: each bid provides full information for all larger bids and all private values

- if bidder wins, then any larger bid wins too
- if bidder loses, then others' maximum bid is perfectly observed

Bid \ Time	1	2	3	4	5	...	T
Price 1							✓
Price 2		✓					✓
Price 3	✓	✓					✓
Price 4	✓	✓					✓
Price 5	✓	✓		✓	✓		✓
...	✓	✓		✓	✓		✓
Price K	✓	✓	✓	✓	✓	...	✓

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							✓
Price 2		✓					✓
Price 3	✓	✓					✓
Price 4	✓	✓					✓
Price 5	✓	✓		✓	✓		✓
...	✓	✓		✓	✓		✓
Price K	✓	✓	✓	✓	✓	...	✓

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3							
Price 4							
Price 5							
...							
Price K							

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3							
Price 4							
Price 5							
...							
Price K							

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
Price 1	✓						
Price 2							
Price 3							
Price 4							
Price 5							
...							
Price K							

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1							
Price 2							
Price 3							
Price 4							
Price 5							
...							
Price K							

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
Price 1	✓						
Price 2	✓						
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
Price 1	✓						
Price 2	✓						
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
Price 1	✓						
Price 2	✓						
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
Price 2	✓						
Price 3	✓						
Price 4	✓						
~~Price 5~~	✓						
...	✓						
Price K	✓						

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
Price 1	✓						
Price 2	✓						
Price 3	✓						
Price 4	✓						
Price 5	✓						
...	✓						
Price K	✓						

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
Price 2	✓	✓					
Price 3	✓						
Price 4	✓						
~~Price 5~~	✓						
...	✓						
Price K	✓						

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
Price 1	✓						
Price 2	✓	✓					
Price 3	✓	✓					
Price 4	✓	✓					
Price 5	✓	✓					
...	✓	✓					
Price K	✓	✓					

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
Price 2	✓	✓					
Price 3	✓	✓					
Price 4	✓	✓					
~~Price 5~~	✓						
...	✓	✓					
Price K	✓	✓					

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
~~Price 2~~	✓	✓					
Price 3	✓	✓					
~~Price 4~~	✓	✓					
~~Price 5~~	✓	✓					
...	✓	✓					
Price K	✓	✓					

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
Price 2	✓	✓					
Price 3	✓	✓					
Price 4	✓	✓					
~~Price 5~~	✓						
...	✓	✓					
Price K	✓	✓					

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
~~Price 2~~	✓	✓					
Price 3	✓	✓	✓				
~~Price 4~~	✓	✓					
~~Price 5~~	✓	✓					
...	✓	✓					
Price K	✓	✓					

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
Price 2	✓	✓					
Price 3	✓	✓					
Price 4	✓	✓					
~~Price 5~~	✓						
...	✓	✓					
Price K	✓	✓					

environment under private value #2

Algorithm: monotone successive elimination

The monotone successive elimination (MSE) policy: at each time,

- bidder observes the current private value (context)
- successively eliminate probably bad bids (actions) under this context
- choose the smallest non-eliminated bid (action) under this context

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
~~Price 2~~	✓	✓					
Price 3	✓	✓	✓				
~~Price 4~~	✓	✓					
~~Price 5~~	✓	✓					
...	✓	✓	✓				
Price K	✓	✓	✓				

environment under private value #1

Bid \ Time	1	2	3	4	5	...	T
~~Price 1~~	✓						
Price 2	✓	✓					
Price 3	✓	✓	✓				
Price 4	✓	✓	✓				
~~Price 5~~	✓						
...	✓	✓	✓				
Price K	✓	✓	✓				

environment under private value #2

Performance of MSE

Theorem (Upper Bound with Exchangeable Contexts)

For contextual bandits with monotone feedback, if the contexts have an exchangeable distribution, then the MSE policy satisfies

$$\mathbb{E}[\text{regret of } \pi^{\mathsf{MSE}}] \lesssim \sqrt{T}\log(T)\log(\#\text{context} \cdot \#\text{action} \cdot T).$$

Performance of MSE

Theorem (Upper Bound with Exchangeable Contexts)

For contextual bandits with monotone feedback, if the contexts have an exchangeable distribution, then the MSE policy satisfies

$$\mathbb{E}[\text{regret of } \pi^{\mathsf{MSE}}] \lesssim \sqrt{T} \log(T) \log(\#\text{context} \cdot \#\text{action} \cdot T).$$

Corollary

When the private values are exchangeable, for stochastic auctions with (left or right) censored feedback, the MSE bidding policy achieves an $O(\sqrt{T} \log^2 T)$ expected regret.

Limitation of MSE

Theorem (Lower Bound)

There exists an instance of contextual bandit with monotone feedback and an adversarially chosen sequence of contexts such that, any policy incurs a worst-case regret at least $\Omega(T^{2/3})$.

Limitation of MSE

Theorem (Lower Bound)

There exists an instance of contextual bandit with monotone feedback and an adversarially chosen sequence of contexts such that, any policy incurs a worst-case regret at least $\Omega(T^{2/3})$.

- $\widetilde{O}(\sqrt{T})$ regret on average, but $\Omega(T^{2/3})$ for worst-case contexts
- monotone feedback is insufficient to achieve a small regret

An Interval-Splitting Scheme

Help from the reward function

stochastic private value: $\sqrt{T}$ regret

adversarial private value: $T^{2/3}$ regret

$\uparrow$

monotone feedback

full information

small bid

large bid

Help from the reward function

stochastic private value: $\sqrt{T}$ regret

adversarial private value: $T^{2/3}$ regret

monotone feedback

full information

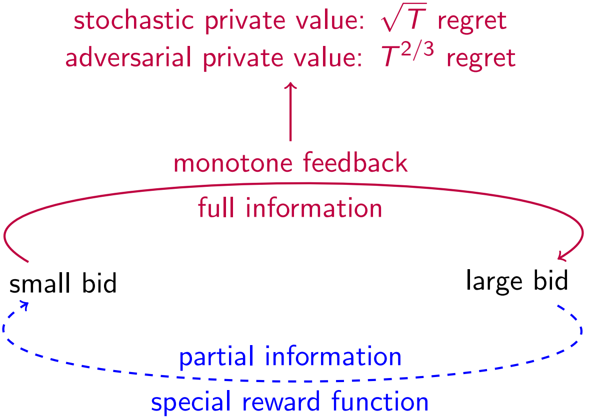
small bid

large bid

partial information

special reward function

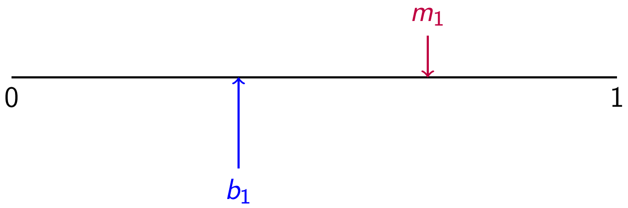
Help from the reward function



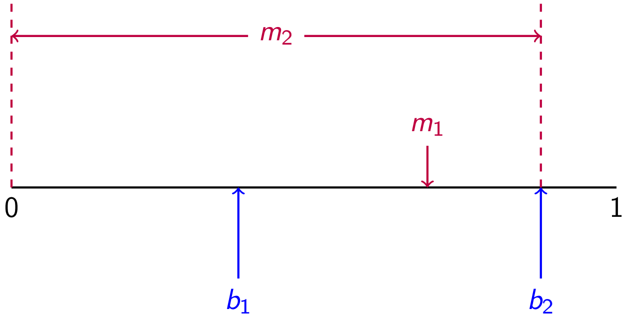
For prices $b < b'$:

$$\mathbb{P}(m_t > b) = \underbrace{\mathbb{P}(m_t > b')}_{\text{one more observation}} + \underbrace{\mathbb{P}(b < m_t \leq b')}_{\text{smaller target quantity}}$$

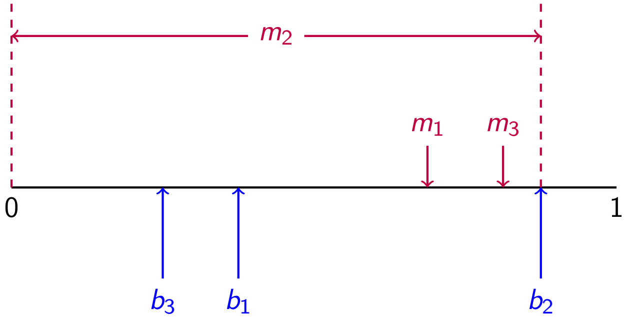
Interval-splitted estimation



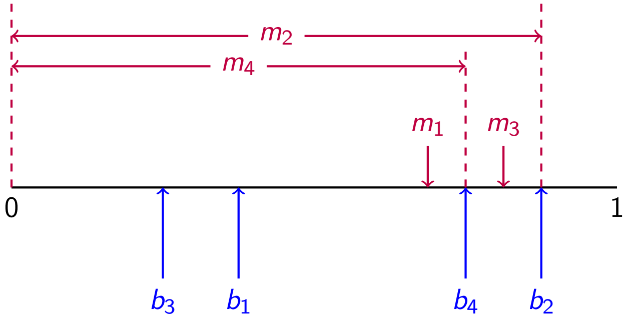
Interval-splitted estimation



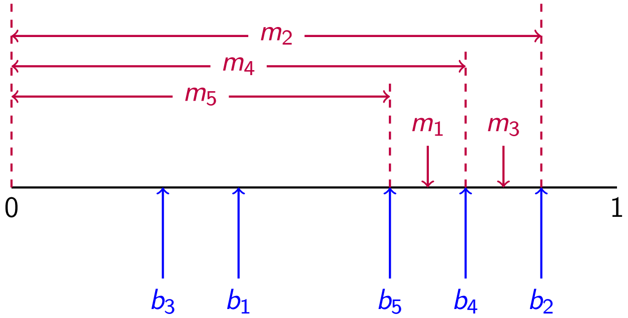
Interval-splitted estimation



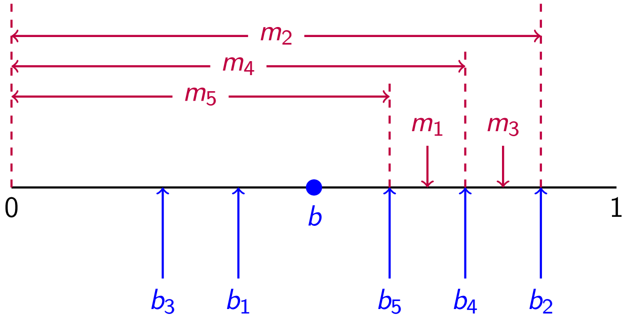
Interval-splitted estimation



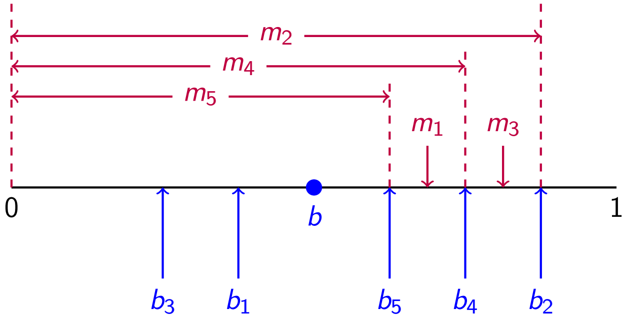
Interval-splitted estimation



Interval-splitted estimation

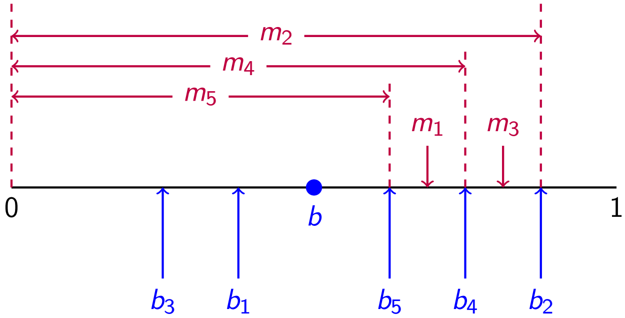


Interval-splitted estimation



$$\widehat{\mathbb{P}}(m_t > b) = \widehat{\mathbb{P}}(b < m_t \leq b_5) + \widehat{\mathbb{P}}(b_5 < m_t \leq b_4) + \widehat{\mathbb{P}}(b_4 < m_t \leq b_2) + \widehat{\mathbb{P}}(m_t > b_2)$$
$$= \frac{0}{2} + \frac{1}{3} + \frac{1}{4} + \frac{0}{5}$$

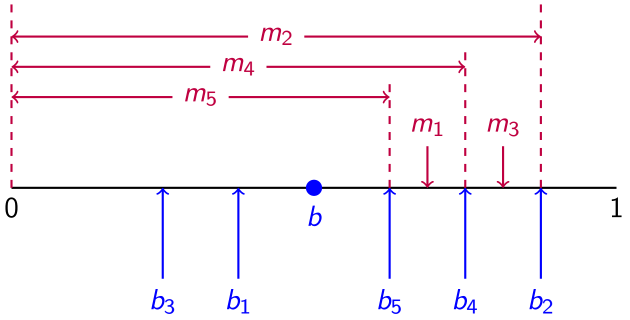
Interval-splitted estimation



$$
\begin{aligned}
\widehat{\mathbb{P}}(m_t > b) &= \widehat{\mathbb{P}}(b < m_t \leq b_5) + \widehat{\mathbb{P}}(b_5 < m_t \leq b_4) + \widehat{\mathbb{P}}(b_4 < m_t \leq b_2) + \widehat{\mathbb{P}}(m_t > b_2) \\
&= \frac{0}{2} + \frac{1}{3} + \frac{1}{4} + \frac{0}{5}
\end{aligned}
$$

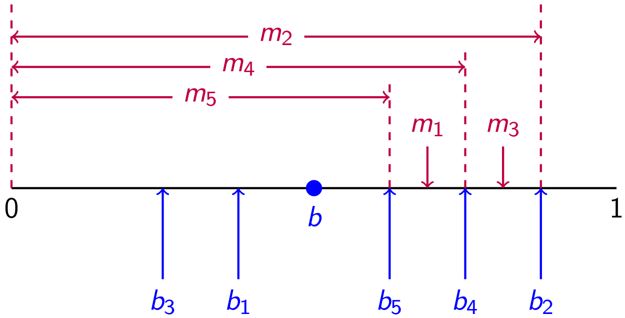
(an additive version of Kaplan-Meier estimator)

Interval-splitted estimation



$$\mathrm{sd}(b) \approx \sqrt{\frac{\mathbb{P}(b < m_t \leq b_5)}{2} + \frac{\mathbb{P}(b_5 < m_t \leq b_4)}{3} + \frac{\mathbb{P}(b_4 < m_t \leq b_2)}{4} + \frac{\mathbb{P}(m_t > b_2)}{5}}$$

Interval-splitted estimation



$$\mathrm{sd}(b) \approx \sqrt{\frac{\mathbb{P}(b < m_t \leq b_5)}{2} + \frac{\mathbb{P}(b_5 < m_t \leq b_4)}{3} + \frac{\mathbb{P}(b_4 < m_t \leq b_2)}{4} + \frac{\mathbb{P}(m_t > b_2)}{5}}$$

$$\widehat{\mathrm{sd}(b)} \approx \sqrt{\frac{\widehat{\mathbb{P}}(b < m_t \leq b_5)}{2} + \frac{\widehat{\mathbb{P}}(b_5 < m_t \leq b_4)}{3} + \frac{\widehat{\mathbb{P}}(b_4 < m_t \leq b_2)}{4} + \frac{\widehat{\mathbb{P}}(m_t > b_2)}{5}}$$

UCB policy

The upper confidence bound policy:

$$b_t = \arg\max_{b \in [0,1]} \quad (v_t - b) \cdot \left(\widehat{\mathbb{P}}_t(m_t \leq b) + \widehat{\mathrm{sd}_t(b)} \right).$$

UCB policy

The upper confidence bound policy:

$$b_t = \arg\max_{b \in [0,1]} \quad (v_t - b) \cdot \left(\widehat{\mathbb{P}}_t(m_t \leq b) + \widehat{\mathrm{sd}_t(b)} \right).$$

- some technical issues:
  - dependence across different intervals
  - dependence across time
  - estimation error in standard deviation

UCB policy

The upper confidence bound policy:

$$b_t = \arg\max_{b \in [0,1]} \quad (v_t - b) \cdot \left(\widehat{\mathbb{P}}_t(m_t \leq b) + \widehat{\mathrm{sd}_t(b)} \right).$$

- some technical issues:
  - dependence across different intervals
  - dependence across time
  - estimation error in standard deviation
- solution: a multi-stage algorithm

UCB policy

The upper confidence bound policy:

$$b_t = \arg\max_{b \in [0,1]} \quad (v_t - b) \cdot \left(\widehat{\mathbb{P}}_t(m_t \leq b) + \widehat{\mathrm{sd}_t(b)} \right).$$

- some technical issues:
  - dependence across different intervals
  - dependence across time
  - estimation error in standard deviation
- solution: a multi-stage algorithm

Theorem (Upper Bound with Adversarial Private Values)

For adversarially chosen private values, the (multi-stage version of) UCB algorithm achieves

$$\text{regret of } \pi^{\mathrm{UCB}} \lesssim \sqrt{T} \log^3 T.$$

Summary of Part I

monotone feedback

full information

small bid

large bid

Summary of Part I

stochastic private value: $\sqrt{T}$ regret

adversarial private value: $T^{2/3}$ regret

$\uparrow$

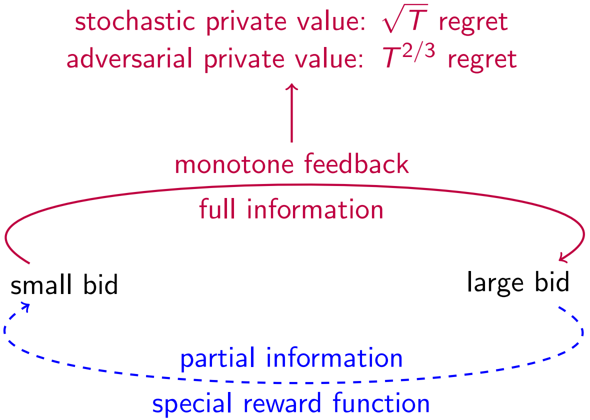
monotone feedback

full information

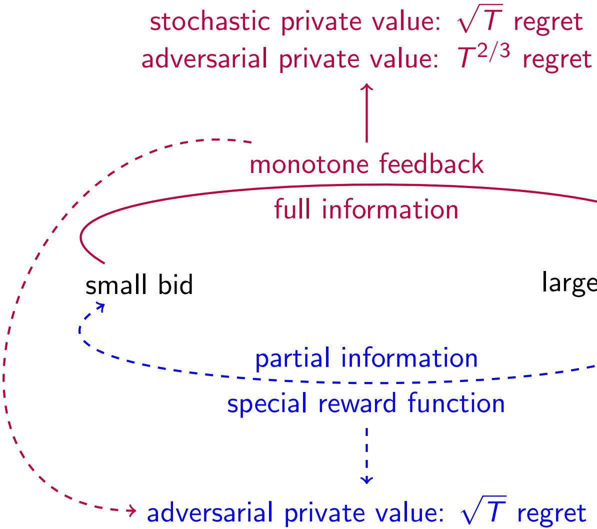
small bid

large bid

Summary of Part I



Summary of Part I



Part II: Adversarial Auctions with Full Information

Zhengyuan Zhou
NYU Stern

Aaron Flores
Yahoo! Research

Erik Ordentlich
Yahoo! Research

Tsachy Weissman
Stanford EE

"Learning to Bid Optimally and Efficiently in Adversarial First-price Auctions"
arXiv: 2007.04568

Adversarial setting revisited

Assumptions:

- modeling of private value: v_t adversarial
- modeling of others' bids: m_t adversarial
- feedback structure: m_t is always revealed

Regret in adversarial auctions

$$R_T(\pi) \triangleq \underbrace{\max_{f \in \mathcal{F}_{\mathrm{Lip}}} \sum_{t=1}^{T} r(f(v_t); v_t, m_t)}_{\text{oracle's reward}} - \underbrace{\mathbb{E}\left[\sum_{t=1}^{T} r(b_t; v_t, m_t)\right]}_{\text{bidder's reward}},$$

where $\mathcal{F}_{\mathrm{Lip}}$ is the set of all 1-Lipschitz functions $f : [0,1] \to [0,1]$.

An Optimal and Efficient Policy

Prediction with expert advice

- oracle $f \in \mathcal{F}_{\mathsf{Lip}} \longleftrightarrow$ expert
- expert f bids price $b_t = f(v_t)$ at each time
- full-information feedback: rewards of all experts are revealed

Prediction with expert advice

- oracle $f \in \mathcal{F}_{\text{Lip}} \longleftrightarrow$ expert
- expert f bids price $b_t = f(v_t)$ at each time
- full-information feedback: rewards of all experts are revealed

Expert \ Time	1	2	3	4	$\cdots$	T
policy f_1						
policy f_2						
policy f_3	✓					
policy f_4						
$\cdots$						
policy f_K						

Prediction with expert advice

- oracle $f \in \mathcal{F}_{\text{Lip}} \longleftrightarrow$ expert
- expert f bids price $b_t = f(v_t)$ at each time
- full-information feedback: rewards of all experts are revealed

Expert \ Time	1	2	3	4	$\cdots$	T
policy f_1	$\checkmark$					
policy f_2	$\checkmark$					
policy f_3	$\checkmark$					
policy f_4	$\checkmark$					
$\cdots$	$\cdots$					
policy f_K	$\checkmark$					

Prediction with expert advice

- oracle $f \in \mathcal{F}_{\text{Lip}} \longleftrightarrow$ expert
- expert f bids price $b_t = f(v_t)$ at each time
- full-information feedback: rewards of all experts are revealed

Expert \ Time	1	2	3	4	$\cdots$	T
policy f_1	✓					
policy f_2	✓	✓				
policy f_3	✓					
policy f_4	✓					
$\cdots$	$\cdots$					
policy f_K	✓					

Prediction with expert advice

- oracle $f \in \mathcal{F}_{\text{Lip}} \longleftrightarrow$ expert
- expert f bids price $b_t = f(v_t)$ at each time
- full-information feedback: rewards of all experts are revealed

Expert \ Time	1	2	3	4	$\cdots$	T
policy f_1	✓	✓				
policy f_2	✓	✓				
policy f_3	✓	✓				
policy f_4	✓	✓				
$\cdots$	$\cdots$	$\cdots$				
policy f_K	✓	✓				

Prediction with expert advice

- oracle $f \in \mathcal{F}_{\text{Lip}} \longleftrightarrow$ expert
- expert f bids price $b_t = f(v_t)$ at each time
- full-information feedback: rewards of all experts are revealed

Expert \ Time	1	2	3	4	$\cdots$	T
policy f_1	✓	✓	✓	✓	$\cdots$	✓
policy f_2	✓	✓	✓	✓	$\cdots$	✓
policy f_3	✓	✓	✓	✓	$\cdots$	✓
policy f_4	✓	✓	✓	✓	$\cdots$	✓
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
policy f_K	✓	✓	✓	✓	$\cdots$	✓

Prediction with expert advice

- oracle $f \in \mathcal{F}_{\text{Lip}} \longleftrightarrow$ expert
- expert f bids price $b_t = f(v_t)$ at each time
- full-information feedback: rewards of all experts are revealed

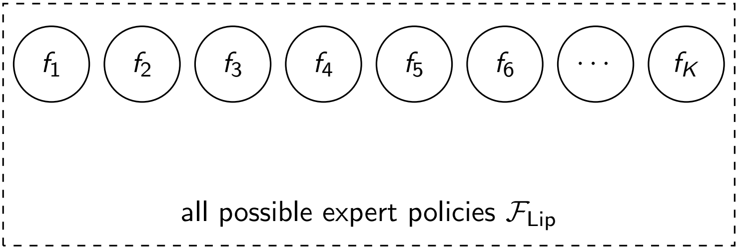
Expert \ Time	1	2	3	4	$\cdots$	T
policy f_1	✓	✓	✓	✓	$\cdots$	✓
policy f_2	✓	✓	✓	✓	$\cdots$	✓
policy f_3	✓	✓	✓	✓	$\cdots$	✓
policy f_4	✓	✓	✓	✓	$\cdots$	✓
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
policy f_K	✓	✓	✓	✓	$\cdots$	✓

Optimal regret relative to the best fixed expert is $\Theta(\sqrt{T \log K})$.

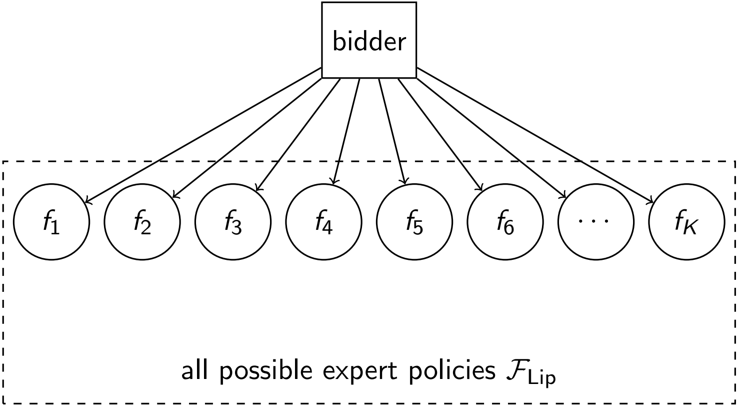
An independent set of experts

all possible expert policies $\mathcal{F}_{\mathsf{Lip}}$

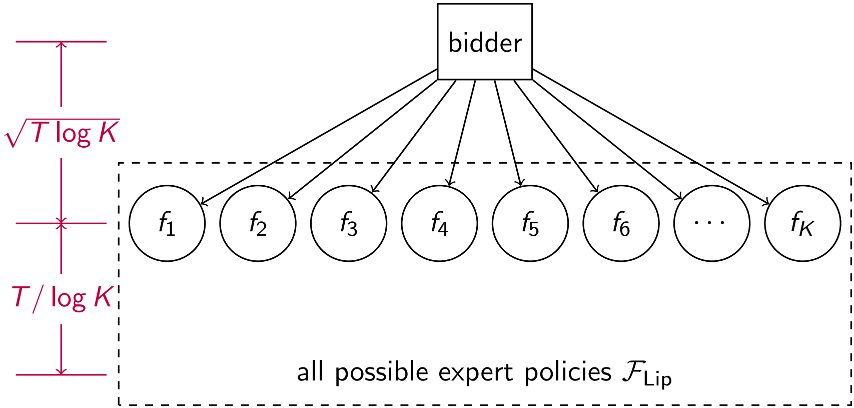
An independent set of experts



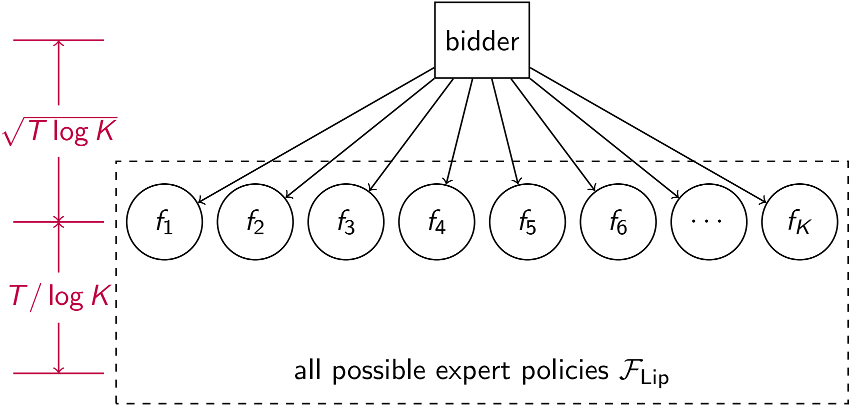
An independent set of experts



An independent set of experts

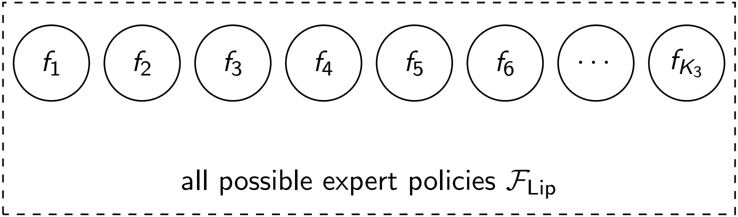


An independent set of experts

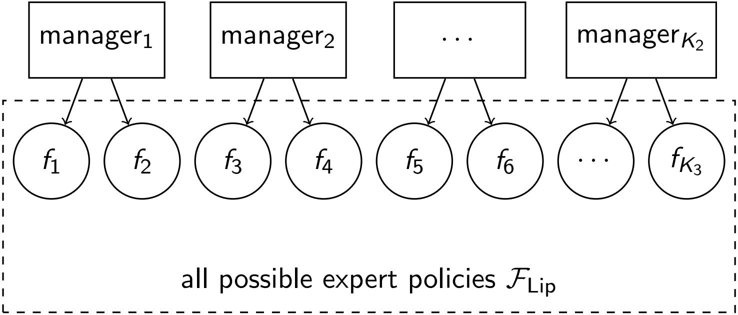


Optimal expert size $K = \exp(T^{1/3})$, achieving regret $T^{2/3}$

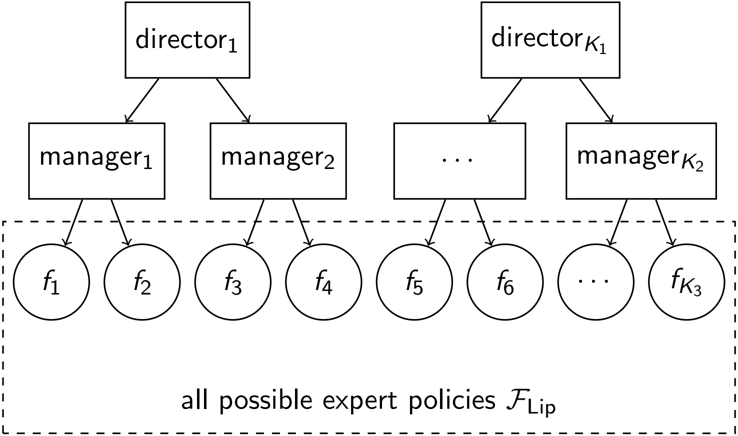
A hierarchical chaining of experts



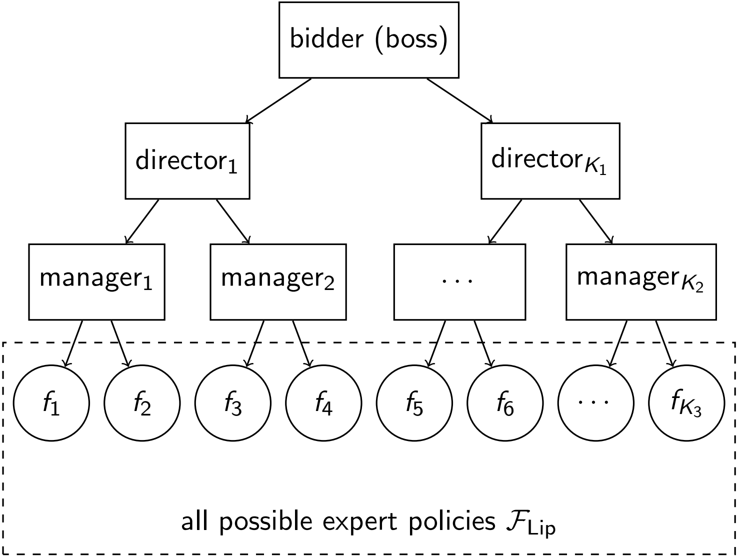
A hierarchical chaining of experts



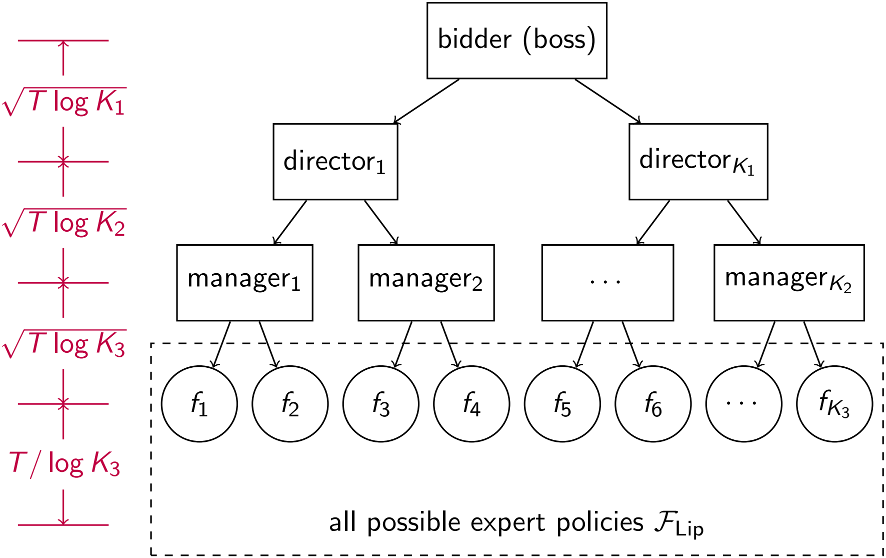
A hierarchical chaining of experts



A hierarchical chaining of experts



A hierarchical chaining of experts



Help from a good expert

- note that the reward $b \mapsto (v - b)\mathbb{1}(b \geq m)$ is discontinuous
- need a good notion of similarity

Help from a good expert

- note that the reward $b \mapsto (v - b)\mathbb{1}(b \geq m)$ is discontinuous
- need a good notion of similarity

Definition (Good Expert)

In prediction with expert advice, an expert is Δ-good if at each time, the reward of that expert is Δ-close to the reward of the best expert.

Help from a good expert

- note that the reward $b \mapsto (v - b)\mathbb{1}(b \geq m)$ is discontinuous
- need a good notion of similarity

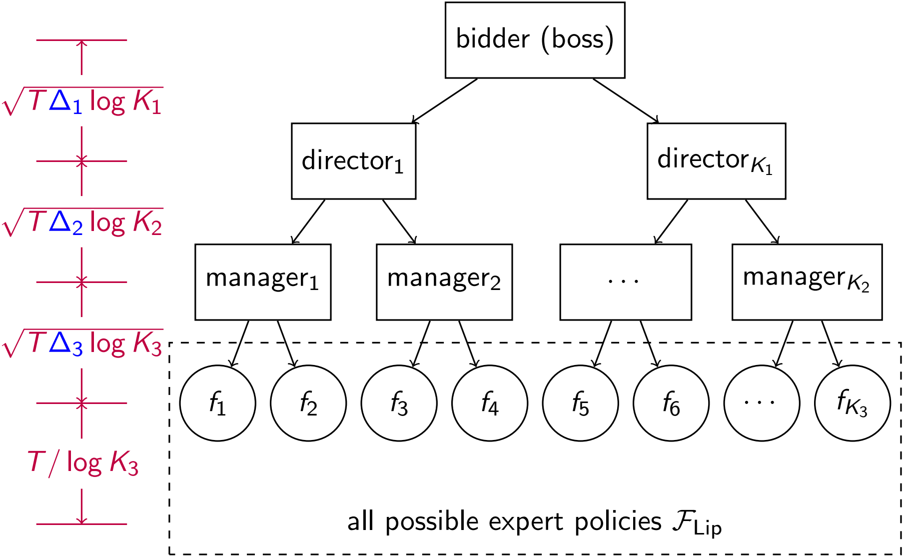
Definition (Good Expert)

In prediction with expert advice, an expert is Δ-good if at each time, the reward of that expert is Δ-close to the reward of the best expert.

Theorem (Optimal Regret with Good Expert)

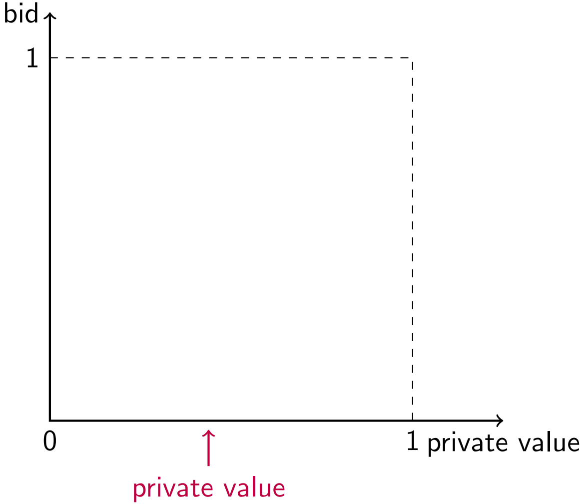
For $\Delta \in [T^{-1} \log K, 1]$, the optimal regret in prediction with expert advice and a Δ-good expert is $\Theta(\sqrt{T\Delta \log K})$.

Improve regrets in the chain



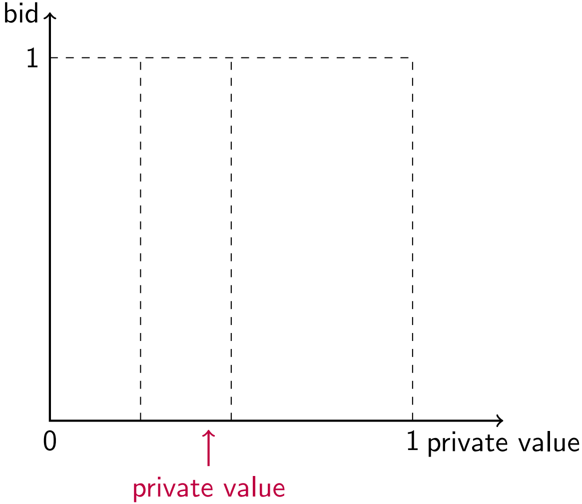
Computational efficiency

A modified policy: successive exponential weighting (SEW)



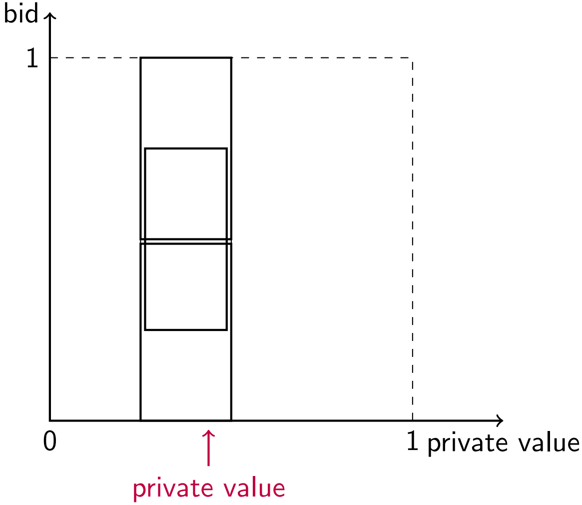
Computational efficiency

A modified policy: successive exponential weighting (SEW)



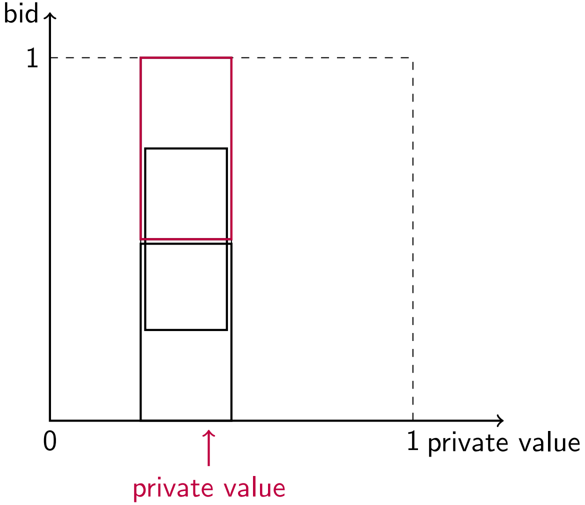
Computational efficiency

A modified policy: successive exponential weighting (SEW)



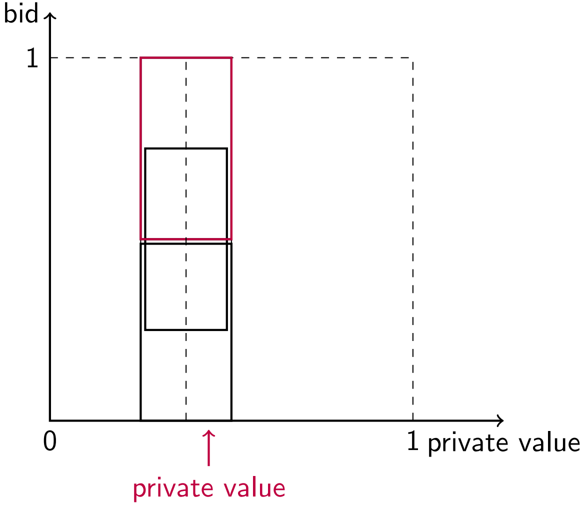
Computational efficiency

A modified policy: successive exponential weighting (SEW)



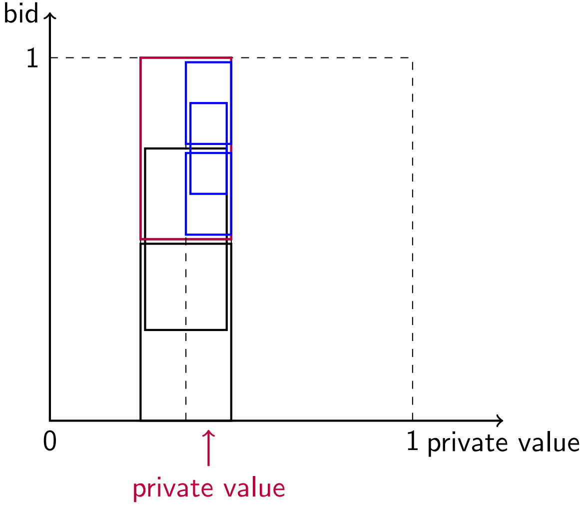
Computational efficiency

A modified policy: successive exponential weighting (SEW)



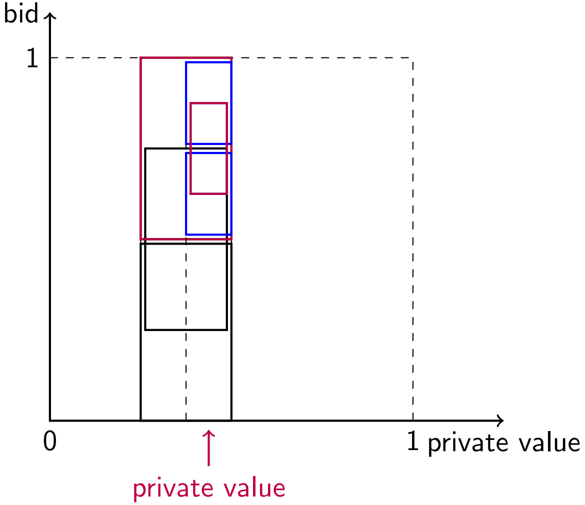
Computational efficiency

A modified policy: successive exponential weighting (SEW)



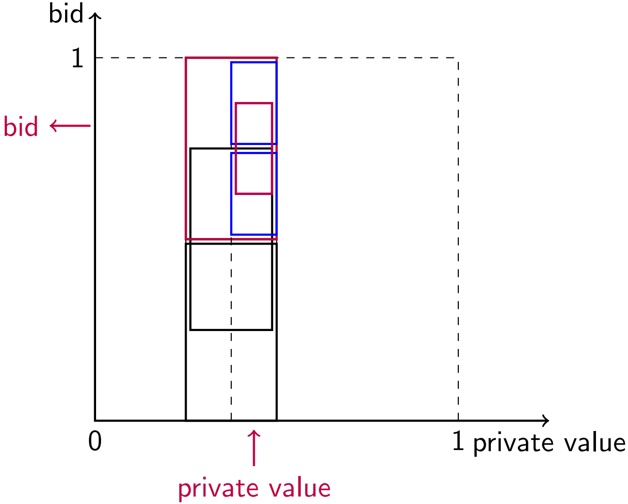
Computational efficiency

A modified policy: successive exponential weighting (SEW)



Computational efficiency

A modified policy: successive exponential weighting (SEW)



Different layers of experts correspond to different resolutions.

Regret guarantee

Theorem (Adversarial Auction with Full Information)

The SEW policy takes $O(T)$ space and $O(T^{1.5})$ time, and satisfies

$$\text{regret of } \pi^{\mathsf{SEW}} \lesssim \sqrt{T} \log T.$$

Real-data Experiments

Real data experiments

Datasets:

- three real datasets from Verizon Media
- each consists of two sequences $\{v_t\}$ and $\{m_t\}$
- duration: from June 8, 2020 to July 6, 2020
- sample size: 0.70M, 1.34M, and 1.53M

Real data experiments

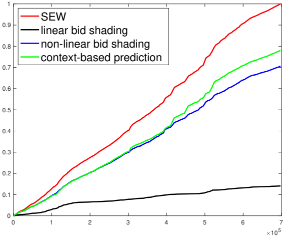
Datasets:

- three real datasets from Verizon Media
- each consists of two sequences $\{v_t\}$ and $\{m_t\}$
- duration: from June 8, 2020 to July 6, 2020
- sample size: 0.70M, 1.34M, and 1.53M

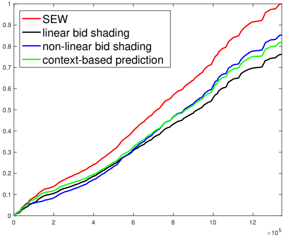
Competing policies:

- linear bid-shading: $b_t = \theta \cdot v_t$
- non-linear bid-shading: $b_t = f(v_t; \theta)$ with non-linear f
- context-based prediction: estimate m_t based on side information

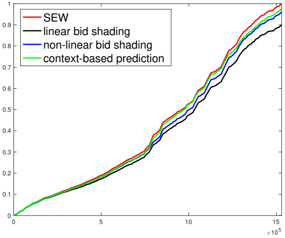
Experimental results



Dataset A



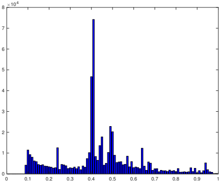
Dataset B



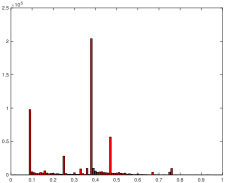
Dataset C

Adaptation to different data nature

Visualization of Dataset A:

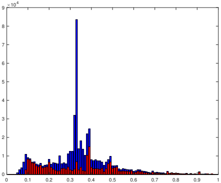


Private values

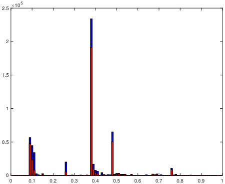


Competing bids

Bidder's bids:

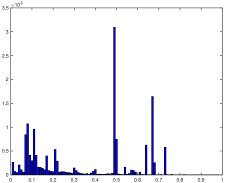


Non-linear bid shading

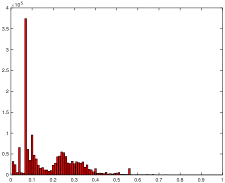


SEW

Visualization of Dataset C:

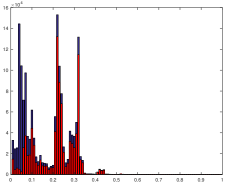


Private values

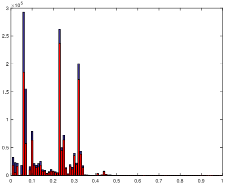


Competing bids

Bidder's bids:

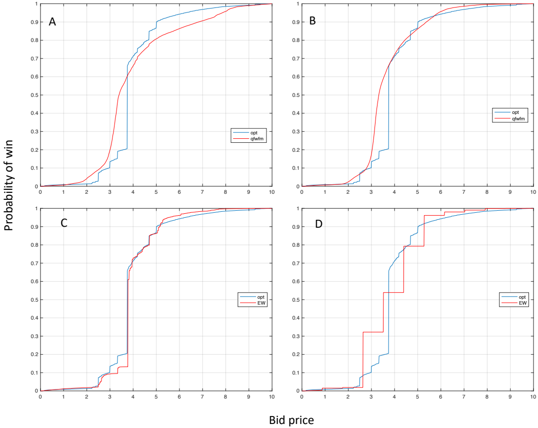


Non-linear bid shading



SEW

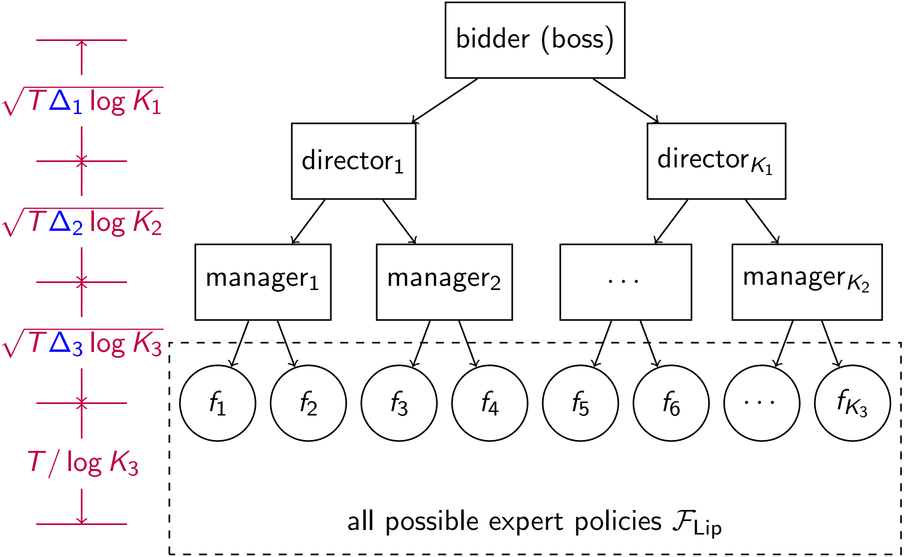
Online experiments



Comparisons of distributions of b_t and m_t

Reference: Zhang et al. "MEOW: A Space-Efficient Non-Parametric Bid Shading Algorithm." KDD 2021.

Summary of Part II



Concluding remarks

Optimal regret efficiently achievable for a single bidder in various scenarios with different assumptions on:

- characteristics of the other bidders' bids
- characteristics of the bidder's private valuation
- feedback structure of the auction
- reference policies with which our bidder competes

Future directions:

- additional contexts (hints, semiparametric model, etc.)
- budget constraints (model return instead of revenue)
- joint value estimation and bidding
- equilibrium theory for multiple bidders/sellers

Thank You!