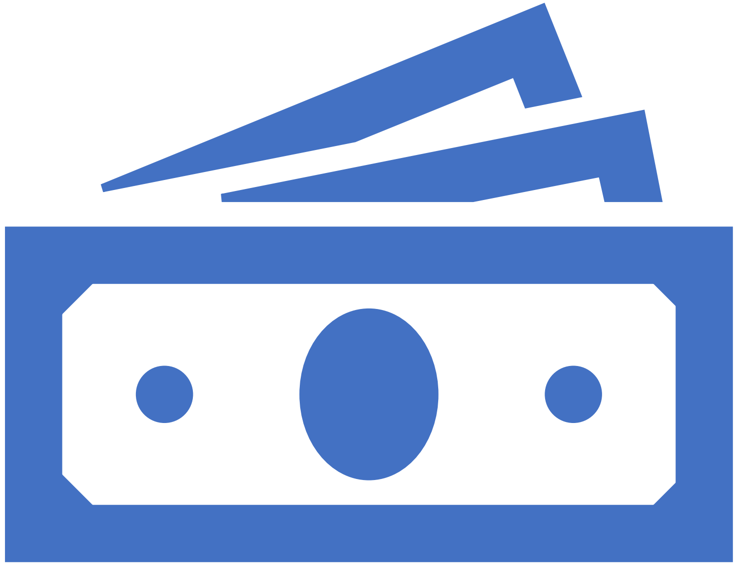


Selling to Multiple No Regret Buyers

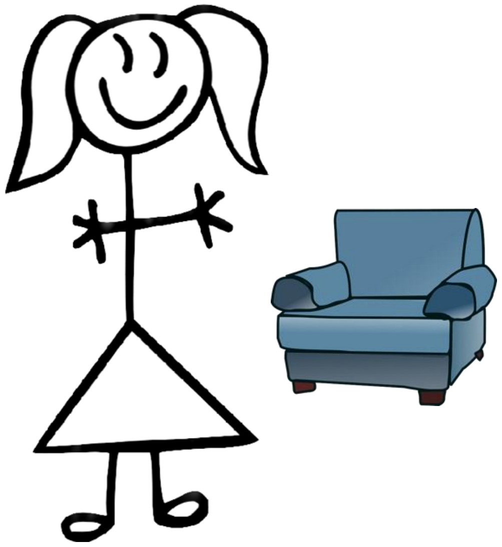
Linda Cai, S. Matthew Weinberg, Evan Wildenhain, Shirley Zhang

Princeton University

Auction: Selling an Item to n Buyers

Auction: Selling an Item to n Buyers

One seller with an item



Auction: Selling an Item to n Buyers

n independent and identical buyers

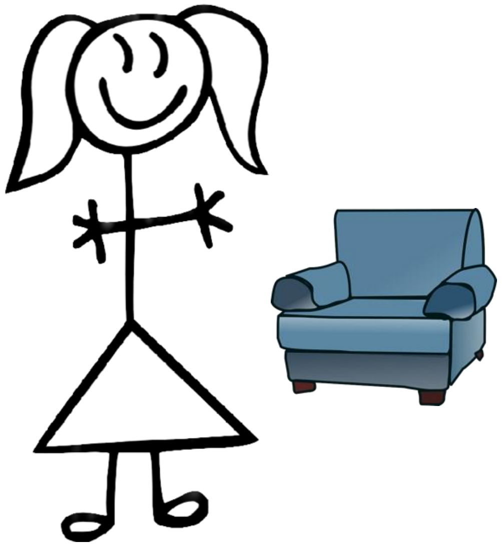
$v_1 \sim D$

$v_2 \sim D$

…

$v_n \sim D$

One seller with an item



Auction: Selling an Item to n Buyers

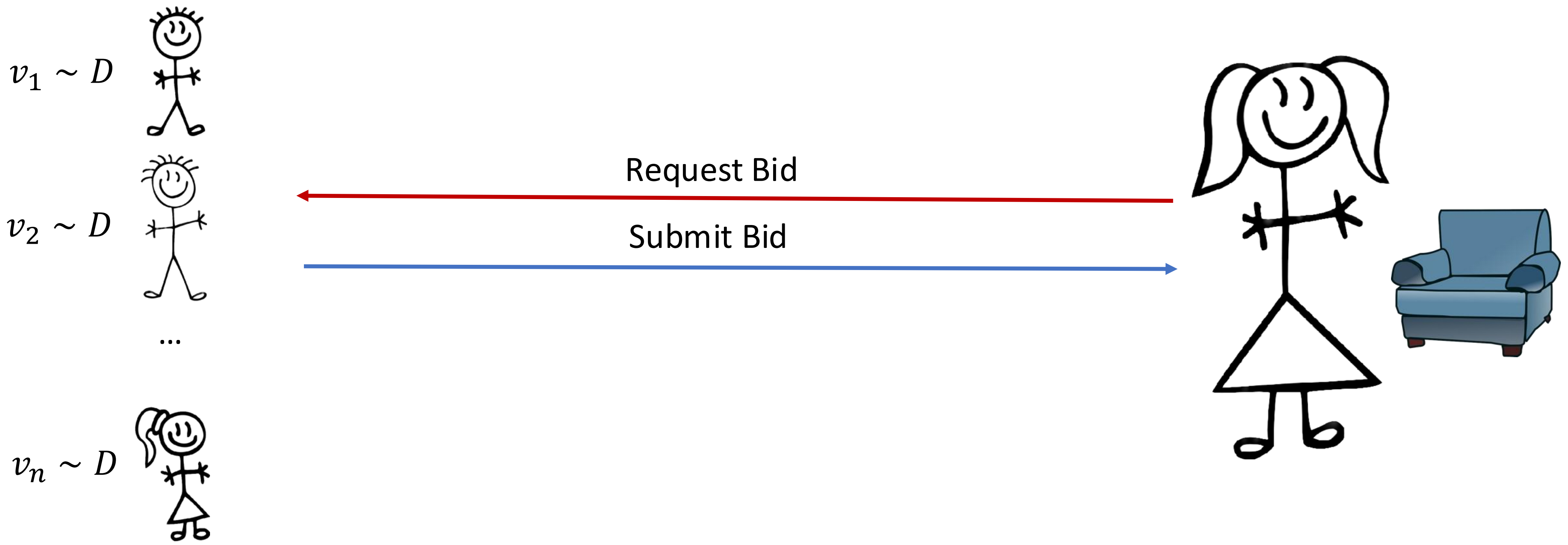
n independent and identical buyers

One seller with an item

Auction: Selling an Item to n Buyers

n independent and identical buyers

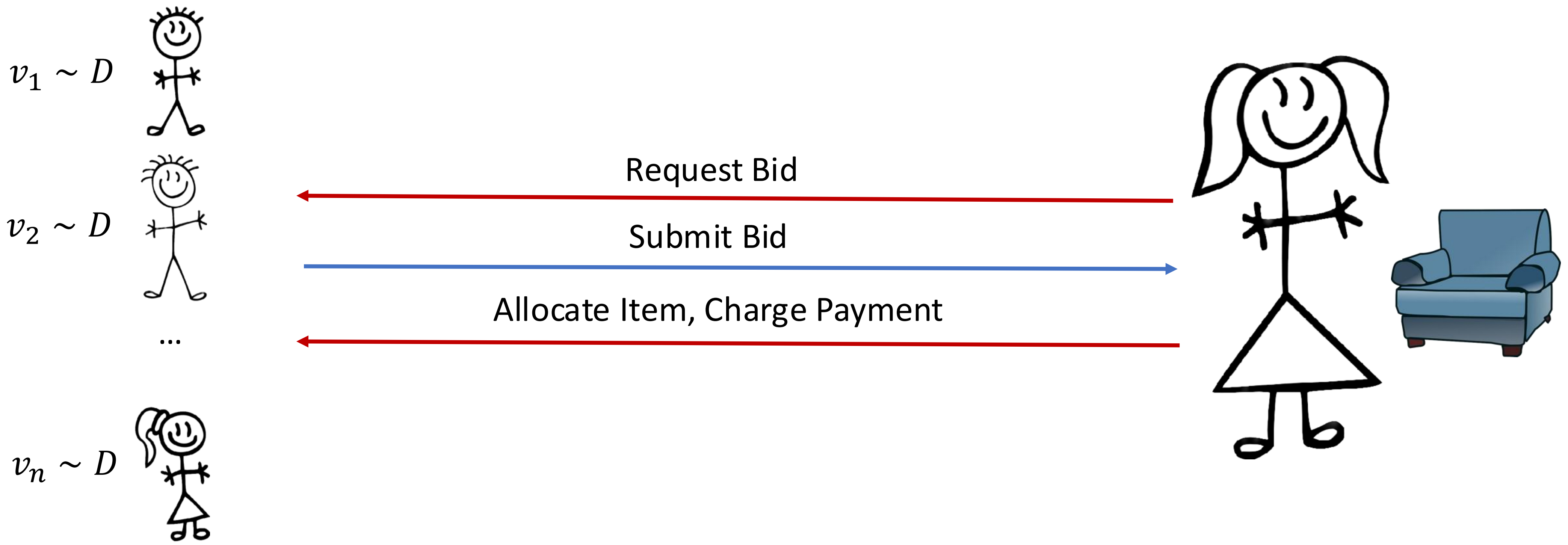
One seller with an item



Auction: Selling an Item to n Buyers

n independent and identical buyers

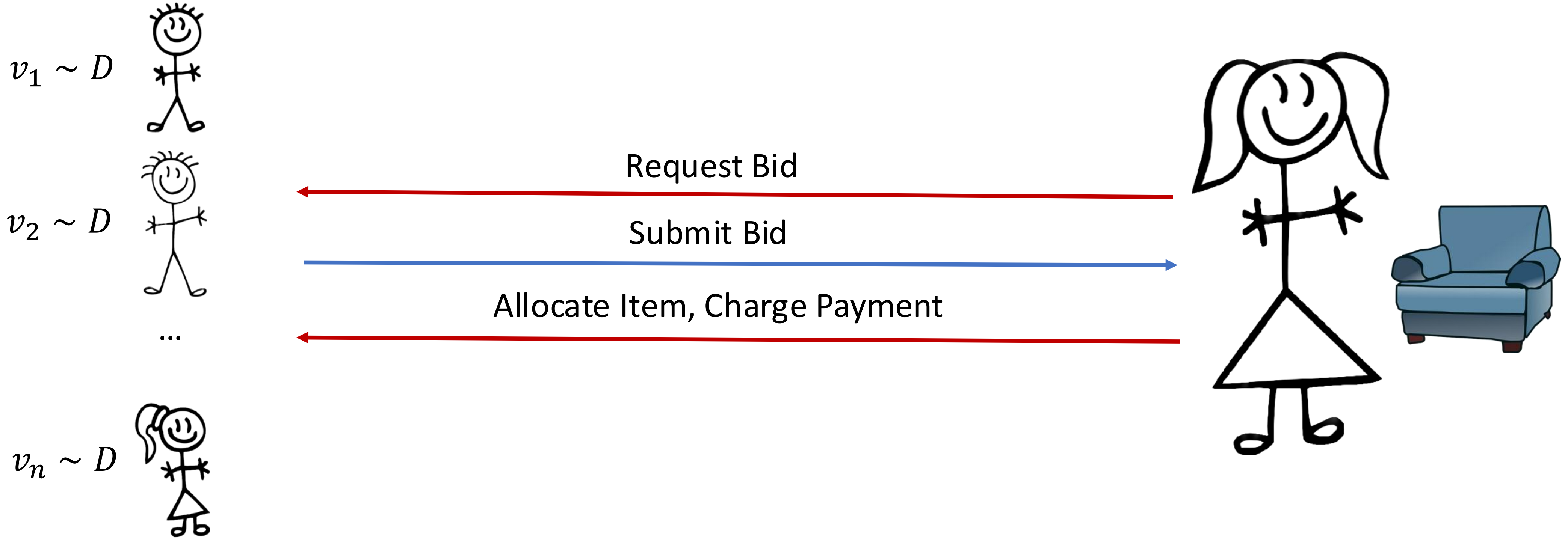
One seller with an item



Auction: Selling an Item to n Buyers

n independent and identical buyers

One seller with an item

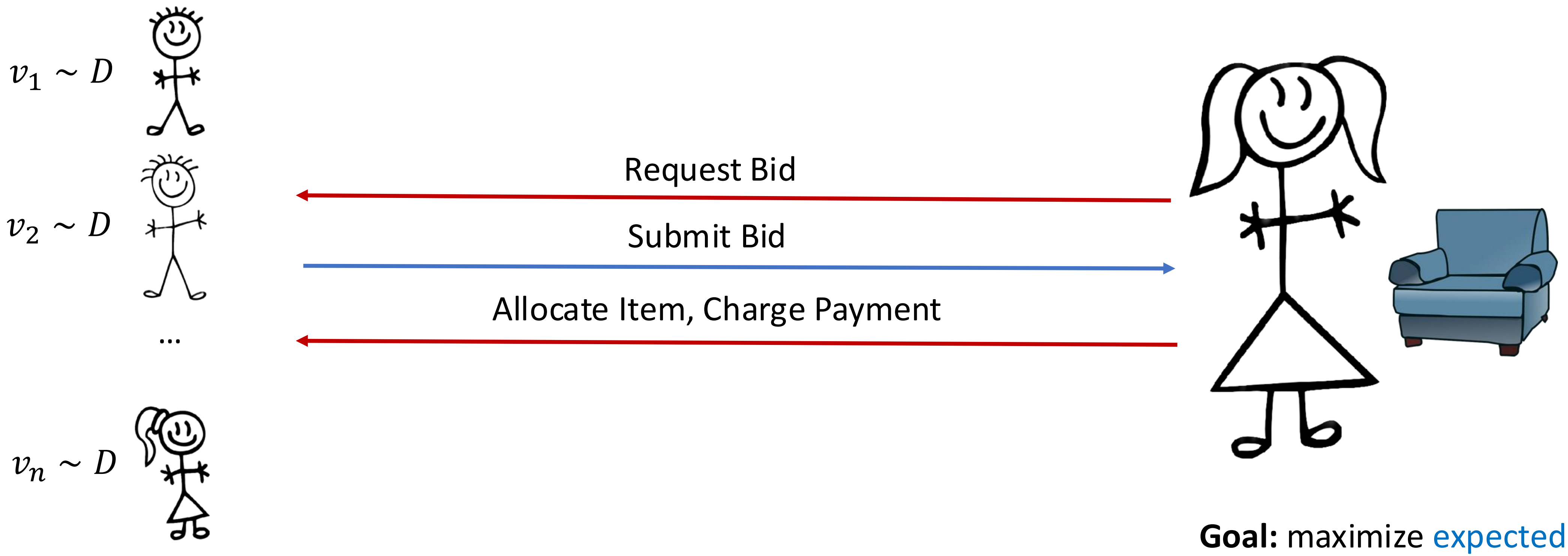


Goal: maximize their expected utility
= value of item * Pr[receiving the item] – expected payment

Auction: Selling an Item to n Buyers

n independent and identical buyers

One seller with an item



Goal: maximize their expected utility
= value of item * Pr[receiving the item] – expected payment

Goal: maximize expected revenue (payment received)

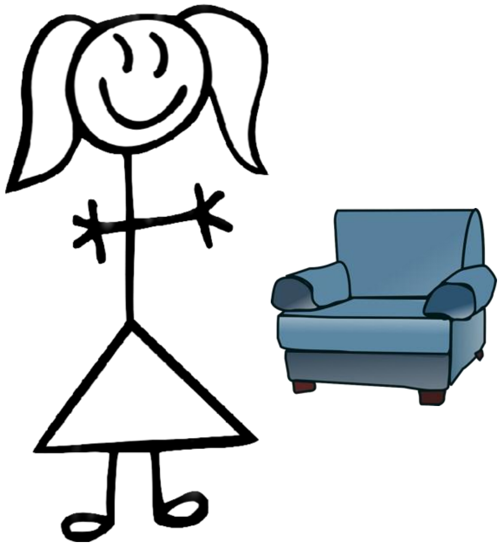
Auction: Selling an Item to n Buyers

$v_1 \sim D$

$v_2 \sim D$

…

$v_n \sim D$



Auction: Selling an Item to n Buyers

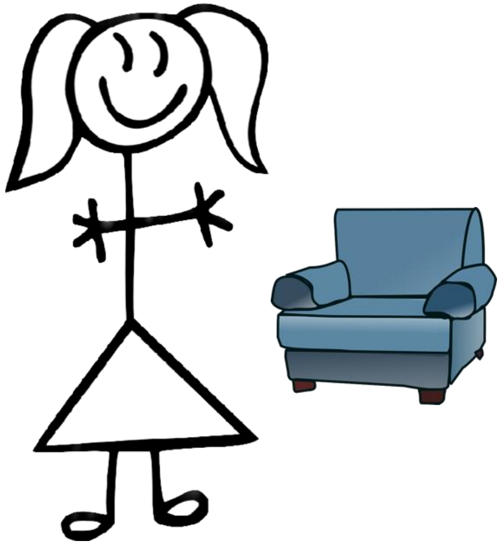
One Shot Auction: classical Bayesian auction design study *truthful* auctions: all buyers can maximize their expected utility by reporting their value truthfully

$v_1 \sim D$ 

$v_2 \sim D$ 

…

$v_n \sim D$ 



Auction: Selling an Item to n Buyers

One Shot Auction: classical Bayesian auction design study *truthful* auctions: all buyers can maximize their expected utility by reporting their value truthfully

$v_1 \sim D$

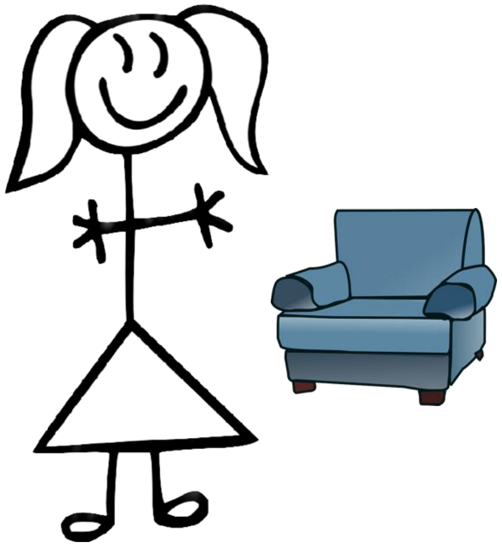
30

$v_2 \sim D$

…

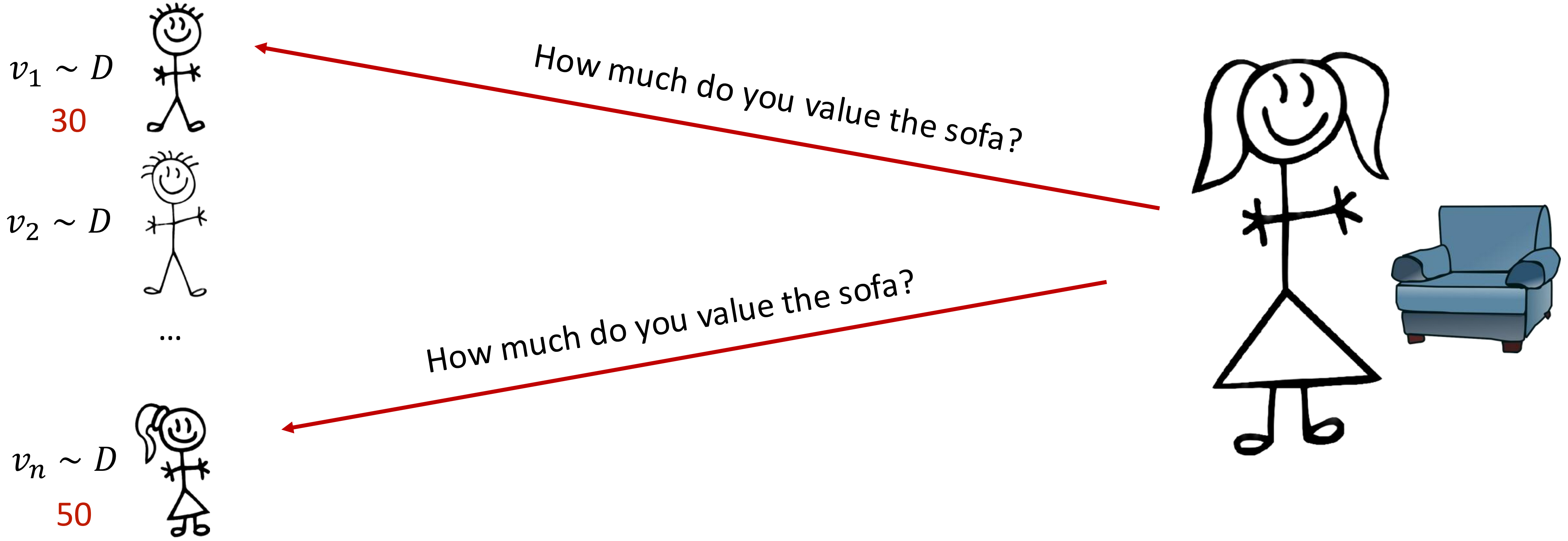
$v_n \sim D$

50



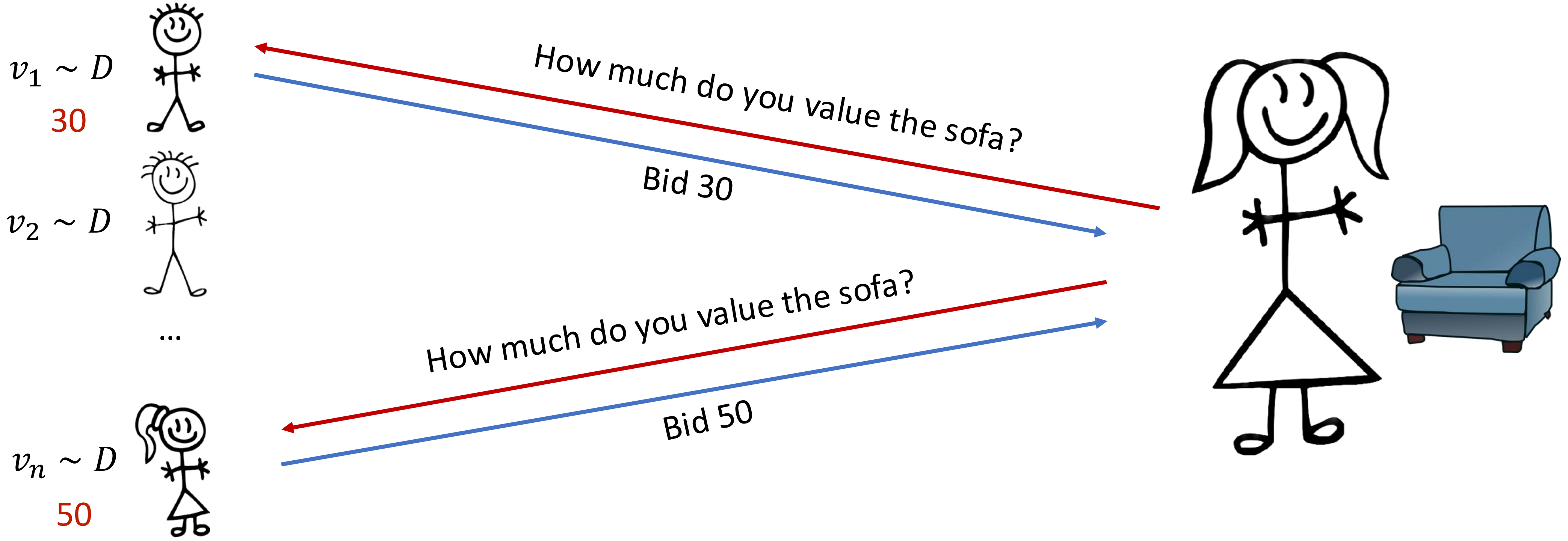
Auction: Selling an Item to n Buyers

One Shot Auction: classical Bayesian auction design study *truthful* auctions: all buyers can maximize their expected utility by reporting their value truthfully



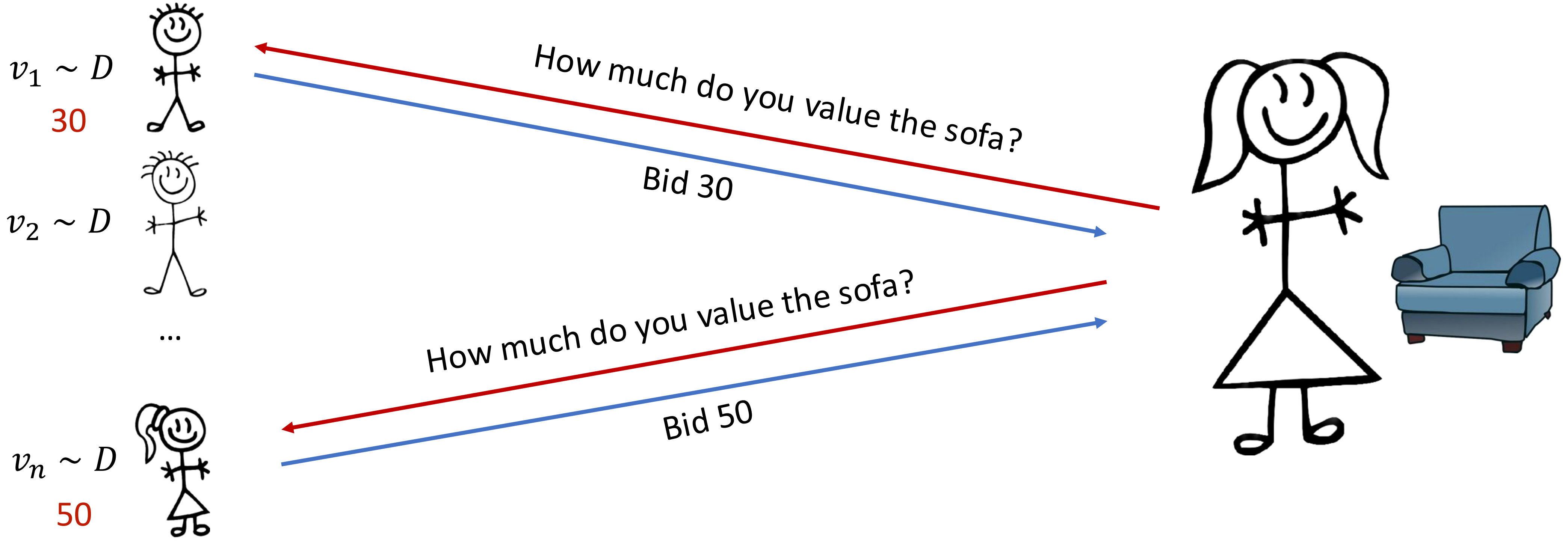
Auction: Selling an Item to n Buyers

One Shot Auction: classical Bayesian auction design study *truthful* auctions: all buyers can maximize their expected utility by reporting their value truthfully



Auction: Selling an Item to n Buyers

One Shot Auction: classical Bayesian auction design study *truthful* auctions: all buyers can maximize their expected utility by reporting their value truthfully



Myerson gives a complete characterization of *revenue optimal* truthful auction

Auction: Selling an Item to n Buyers

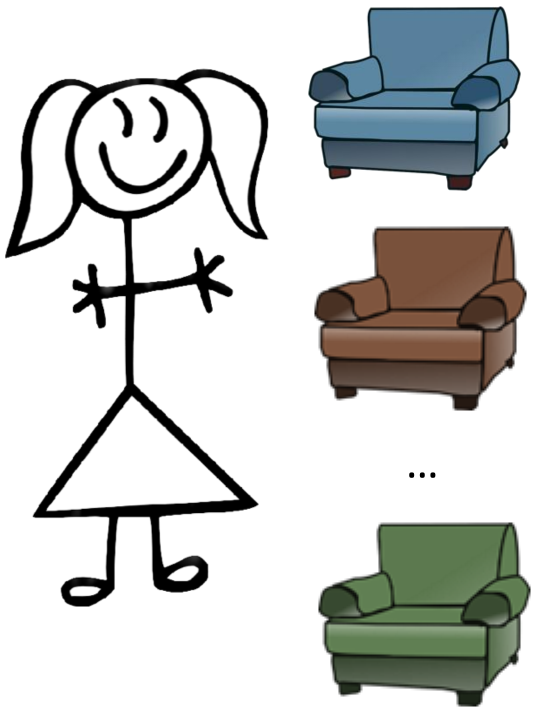
Repeated Auction: there are T rounds, in each round the buyers' values are drawn independently from the prior distribution

$v_{1,i} \sim D$

$v_{2,i} \sim D$

…

$v_{n,i} \sim D$



Day 1

Day 2

…

Day T

Auction: Selling an Item to n Buyers

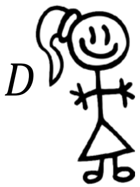
Repeated Auction: there are T rounds, in each round the buyers' values are drawn independently from the prior distribution

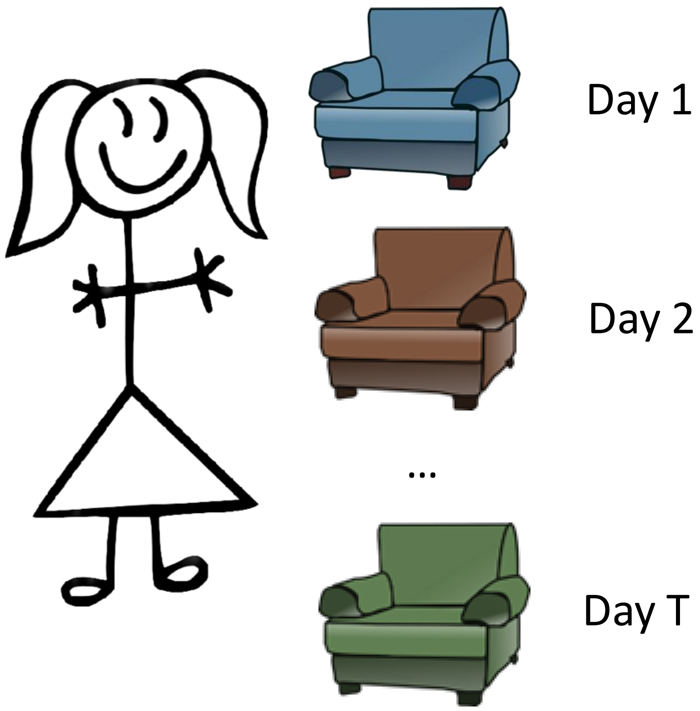
$v_{1,i} \sim D$

$v_{2,i} \sim D$

…

$v_{n,i} \sim D$





Goal: maximize their expected **total** utility

= $\sum_{i=1}^{T}$ (value of item in round i)$*$ Pr[receiving the item in round i] – expected payment in round i

Auction: Selling an Item to n Buyers

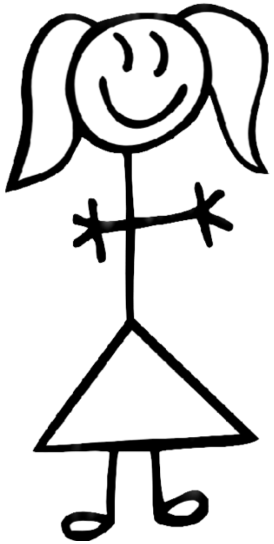
Repeated Auction: there are T rounds, in each round the buyers' values are drawn independently from the prior distribution

$v_{1,i} \sim D$

$v_{2,i} \sim D$

…

$v_{n,i} \sim D$



Day 1

Day 2

…

Day T

Goal: maximize their expected **total** utility
$= \sum_{i=1}^{T}$ (value of item in round i)$*$ Pr[receiving the item in round i] – expected payment in round i

Goal: maximize expected **total** revenue (payment received)

Auction: Selling an Item to n Buyers

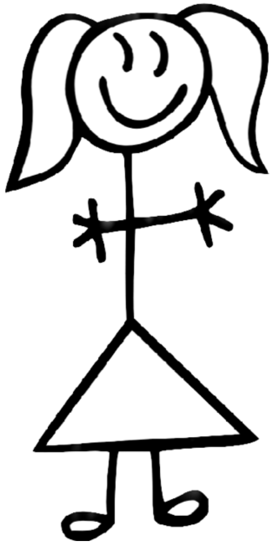
Repeated Auction: there are T rounds, in each round the buyers' values are drawn independently from the prior distribution

$v_{1,i} \sim D$ 

$v_{2,i} \sim D$ 

…

$v_{n,i} \sim D$ 



 Day 1

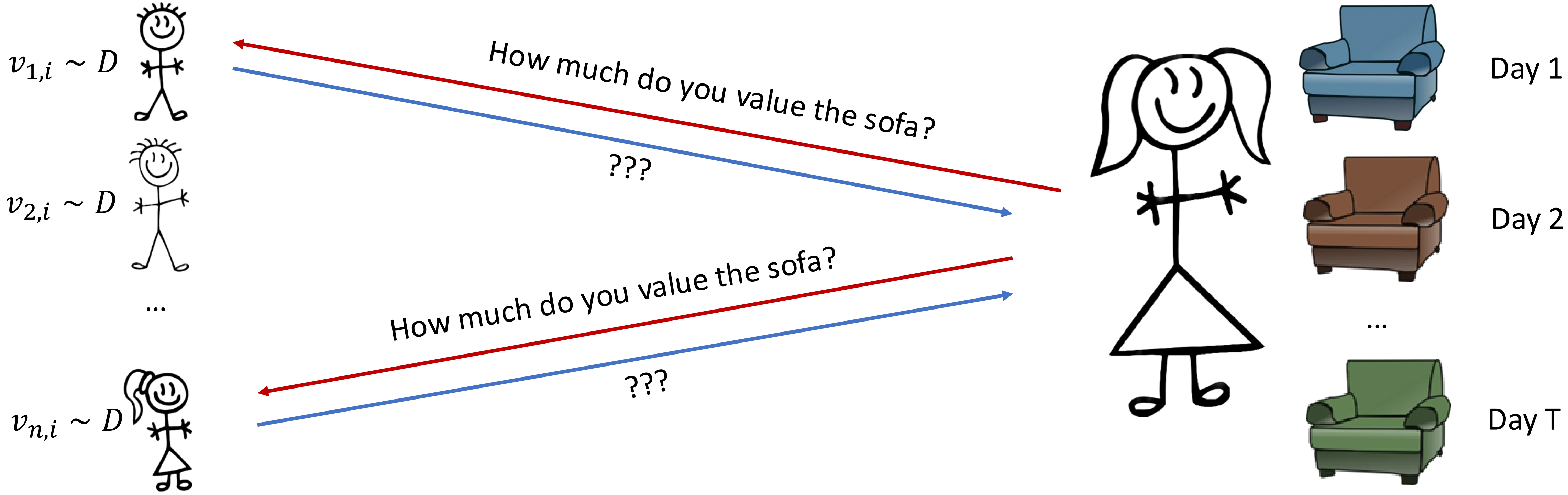
 Day 2

…

 Day T

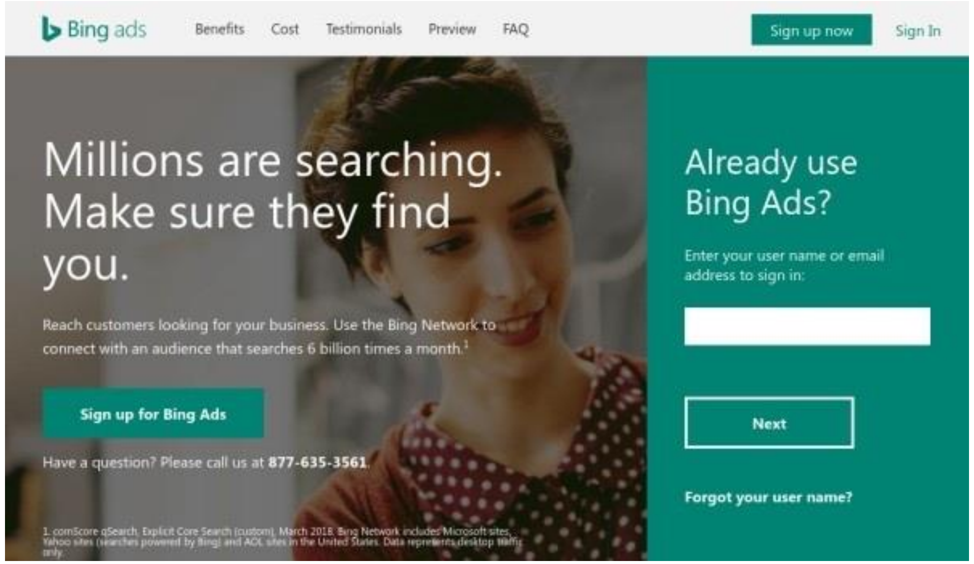
Auction: Selling an Item to n Buyers

Repeated Auction: there are T rounds, in each round the buyers' values are drawn independently from the prior distribution



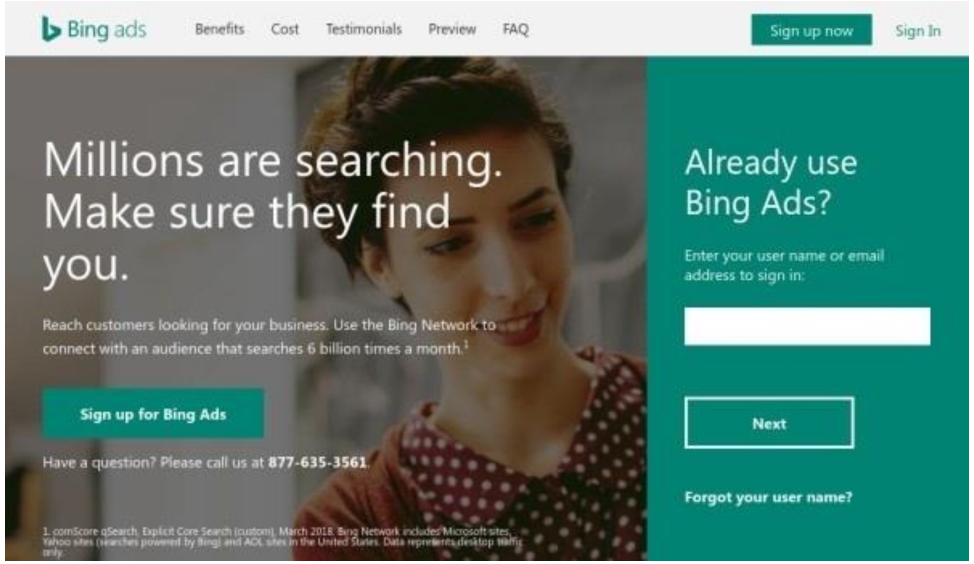
Question: not clear how to incentivize truthfulness, how do we model buyer behavior?

Learning in Repeated Auction



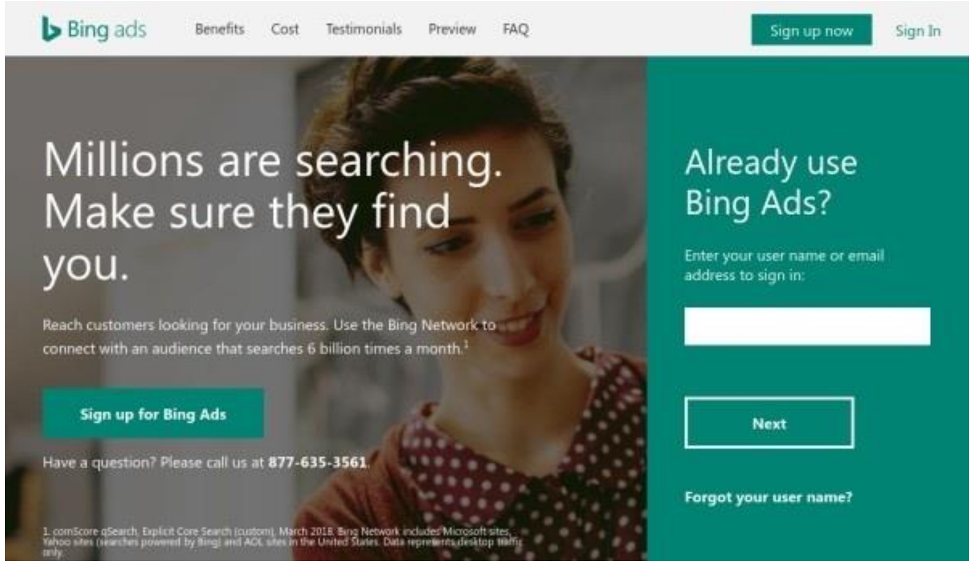
Learning in Repeated Auction

- In classical auction theory, agents are typically assumed to be rational/best responding



Learning in Repeated Auction

- In classical auction theory, agents are typically assumed to be rational/best responding
- **[Nekipelov, Syrgkanis, and Tardos 2015]** Bidder behavior is more consistent with no-regret learning in repeated auctions (e.g. in Bing ad auction)



No Regret Learning with Context

Action 1

Action 2

Action 3

…

Action m

No Regret Learning with Context

Action 1

Action 2

Action 3

…

Action m

In a repeated game with T rounds with context space V and action space A:

No Regret Learning with Context

Action 1

Action 2

Action 3

…

Action m

In a repeated game with T rounds with context space V and action space A:

- In each round t, player i observes a context $v_{i,t} \in V$, then takes an action $a_{i,t} \in A$ and observe a reward $r_t(a_{i,t})$ (the reward function may be correlated with the context)

No Regret Learning with Context

Action 1

Action 2

Action 3

…

Action m

In a repeated game with T rounds with context space V and action space A:

- In each round t, player i observes a context $v_{i,t} \in V$, then takes an action $a_{i,t} \in A$ and observe a reward $r_t(a_{i,t})$ (the reward function may be correlated with the context)
- A *fixed-action strategy* is a strategy that always takes the same action given the same context

No Regret Learning with Context

Action 1

Action 2

Action 3

…

Action m

In a repeated game with T rounds with context space V and action space A:

- In each round t, player i observes a context $v_{i,t} \in V$, then takes an action $a_{i,t} \in A$ and observe a reward $r_t(a_{i,t})$ (the reward function may be correlated with the context)
- A *fixed-action strategy* is a strategy that always takes the same action given the same context
- The player's strategy is *no regret* if their average reward from the process is (approximately) at least that of the best fixed-action strategy

No Regret Learning with Context

Action 1 Action 2 Action 3 … Action m

In a repeated game with T rounds with context space V and action space A:

- In each round t, player i observes a context $v_{i,t} \in V$, then takes an action $a_{i,t} \in A$ and observe a reward $r_t(a_{i,t})$ (the reward function may be correlated with the context)
- A *fixed-action strategy* is a strategy that always takes the same action given the same context
- The player's strategy is *no regret* if their average reward from the process is (approximately) at least that of the best fixed-action strategy

$$\sum_{t=1}^{T} r_t(a_{i,t}) \geq \max_{fixed\, action\, strategy\, s} \sum_{t=1}^{T} r_t(\text{action } s \text{ takes given context } v_{i,t}) - o(T)$$

Learning in Repeated Auctions

Learning in Repeated Auctions

Given a repeated auction designed by the seller:

Learning in Repeated Auctions

Given a repeated auction designed by the seller:

- **Players**: n i.i.d buyers

Learning in Repeated Auctions

Given a repeated auction designed by the seller:

- **Players**: n i.i.d buyers
- **Action space A:** bidding values that are allowed by the seller (e.g multiples of 0.01)

Learning in Repeated Auctions

Given a repeated auction designed by the seller:

- **Players**: n i.i.d buyers
- **Action space A:** bidding values that are allowed by the seller (e.g multiples of 0.01)
- **Buyer i's context** in each round t: equal to their value of the item sold in round t

Learning in Repeated Auctions

Given a repeated auction designed by the seller:

- **Players**: n i.i.d buyers
- **Action space A:** bidding values that are allowed by the seller (e.g multiples of 0.01)
- **Buyer i's context** in each round t: equal to their value of the item sold in round t
- **Fixed action strategies**: strategies that given a fixed value, always bid the same number

Learning in Repeated Auctions

Given a repeated auction designed by the seller:

- **Players**: n i.i.d buyers
- **Action space** A**:** bidding values that are allowed by the seller (e.g multiples of 0.01)
- **Buyer** i**'s context** in each round t: equal to their value of the item sold in round t
- **Fixed action strategies**: strategies that given a fixed value, always bid the same number
- The buyer's strategy is *no regret* if their average reward from the process is (approximately) at least that of the best fixed-action strategy

Learning in Repeated Auctions

Given a repeated auction designed by the seller:

- **Players**: n i.i.d buyers
- **Action space A:** bidding values that are allowed by the seller (e.g multiples of 0.01)
- **Buyer i's context** in each round t: equal to their value of the item sold in round t
- **Fixed action strategies**: strategies that given a fixed value, always bid the same number
- The buyer's strategy is *no regret* if their average reward from the process is (approximately) at least that of the best fixed-action strategy

- Buyers may be using common off-the-shelf no regret learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc

Learning in Repeated Auction

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?

Definition (Mean-Based No Regret): An algorithm is *mean-based no regret* if:
cumulative utility for action i is less than that of another action j by at least $f(T) = o(T)$
$\Rightarrow$ the probability that the algorithm plays action i on round t is at most $o(1)$.

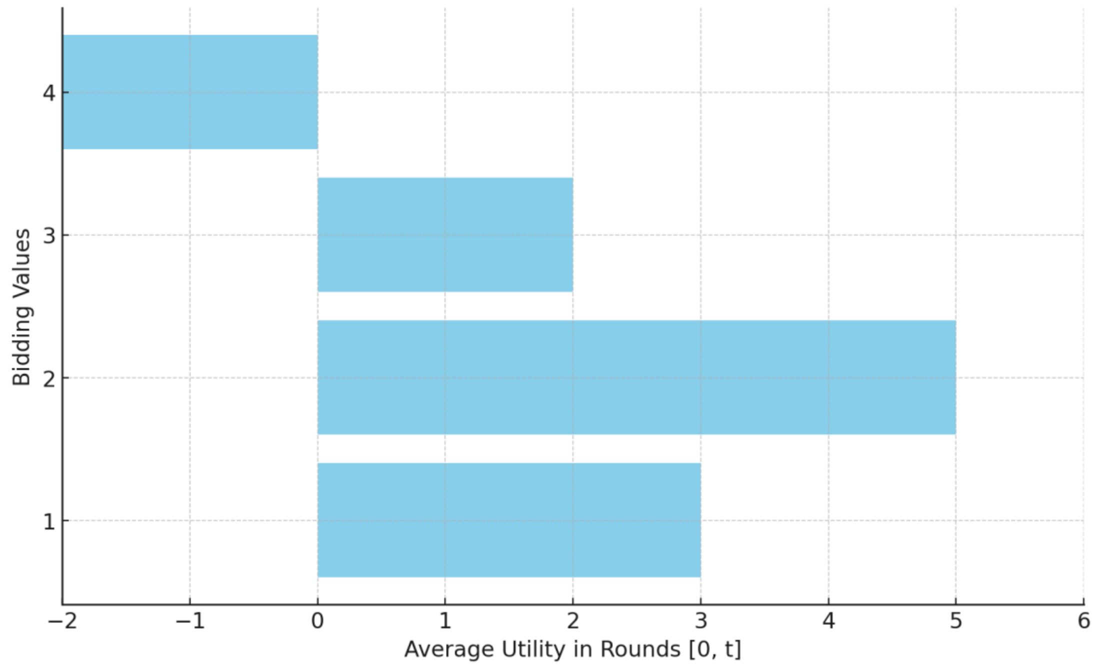
- Intuition: if an action seems much better historically it would be played with much higher probability

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?

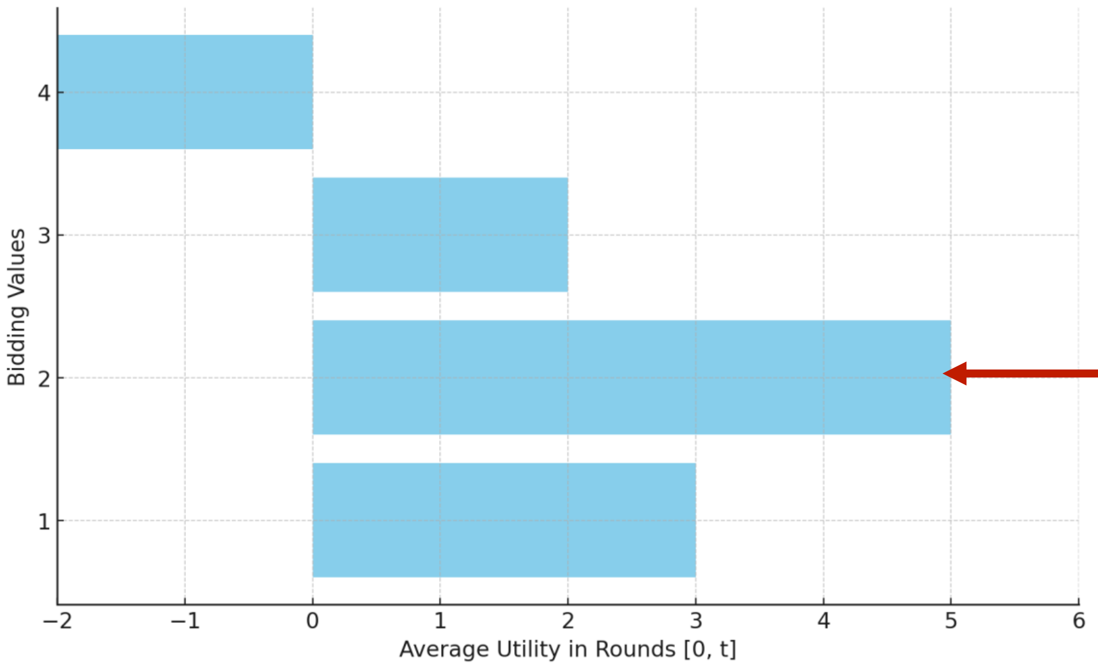
Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?



Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?



A mean-based no regret algorithm will choose to bid 2 with high probability

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?

Definition (Mean-Based No Regret): An algorithm is *mean-based no regret* if:
cumulative utility for action i is less than that of another action j by at least $f(T) = o(T)$
$\Rightarrow$ the probability that the algorithm plays action i on round t is at most $o(1)$.

Example: Follow the Perturbed Leader algorithm

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?

Definition (Mean-Based No Regret): An algorithm is *mean-based no regret* if:
cumulative utility for action i is less than that of another action j by at least $f(T) = o(T)$
$\Rightarrow$ the probability that the algorithm plays action i on round t is at most $o(1)$.

Example: Follow the Perturbed Leader algorithm

1) Allocates random costs to each expert at the initial time;

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?

Definition (Mean-Based No Regret): An algorithm is *mean-based no regret* if:
cumulative utility for action i is less than that of another action j by at least $f(T) = o(T)$
$\Rightarrow$ the probability that the algorithm plays action i on round t is at most $o(1)$.

Example: Follow the Perturbed Leader algorithm

1) Allocates random costs to each expert at the initial time;
2) Subsequently, at each time step, it selects the expert with the lowest cumulative cost, inclusive of the initial step.

Learning in Repeated Auction

- Buyers may be using common off-the-shelf learning algorithms such as Exp3, Multiplicative Weights Update, Follow-the-Perturbed-Leader, etc
- What do these algorithms have in common?

Definition (Mean-Based No Regret): An algorithm is *mean-based no regret* if:
cumulative utility for action i is less than that of another action j by at least $f(T) = o(T)$
$\Rightarrow$ the probability that the algorithm plays action i on round t is at most $o(1)$.

Example: Follow the Perturbed Leader algorithm

1) Allocates random costs to each expert at the initial time;
2) Subsequently, at each time step, it selects the expert with the lowest cumulative cost, inclusive of the initial step.

Once the difference between cumulative utility of two actions exceed the initial cost, the inferior action will never be chosen.

Exploitability of Learning Agents

- **Question:** How much revenue can the seller extract *knowing* that the buyers are using mean-based no regret learning algorithms (but not the specific algorithm they are using)?

- [Braverman, Mao, Schneider, and Weinberg'18] For the one seller one buyer setting:
  - **For naïve mean-based no regret buyer**: Full surplus extraction (seller gets revenue ≈ optimal social welfare and buyer gets ≈ zero utility)
  - **For non overbidding mean-based no regret buyer**: Optimal revenue can be unboundedly worse than for naïve learners, yet still unboundedly better than a seller who runs the same auction in each round.

Exploitability of Learning Agents

- **Question:** How much revenue can the seller extract *knowing* that the buyers are using mean-based no regret learning algorithms (but not the specific algorithm they are using)?

- **[Our Result]** For the one seller ~~one~~ *multiple* buyer setting:
  - **For naïve mean-based no regret buyers**: Full surplus extraction (seller gets revenue ≈ optimal social welfare and buyer gets ≈ zero utility)
  - **For non-overbidding mean-based no regret buyers**: *much harder to understand seller's optimal revenue than the single buyer setting (formal barriers)*

Exploitability of a Naïve learner [BMSW]

Exploitability of a Naïve learner [BMSW]

Idea: lure the buyer into bidding high by offering the item for cheap at first, then start charging the item with a high price. It will take a while for the mean-based no regret buyer to realize they should not bid high anymore.

Exploitability of a Naïve learner [BMSW]

Idea: lure the buyer into bidding high by offering the item for cheap at first, then start charging the item with a high price. It will take a while for the mean-based no regret buyer to realize they should not bid high anymore.

$v_{1,t} = 1/2$

Exploitability of a Naïve learner [BMSW]

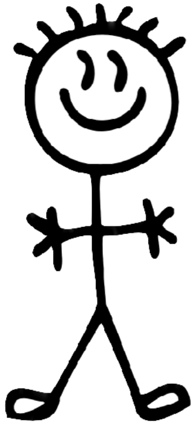
Idea: lure the buyer into bidding high by offering the item for cheap at first, then start charging the item with a high price. It will take a while for the mean-based no regret buyer to realize they should not bid high anymore.

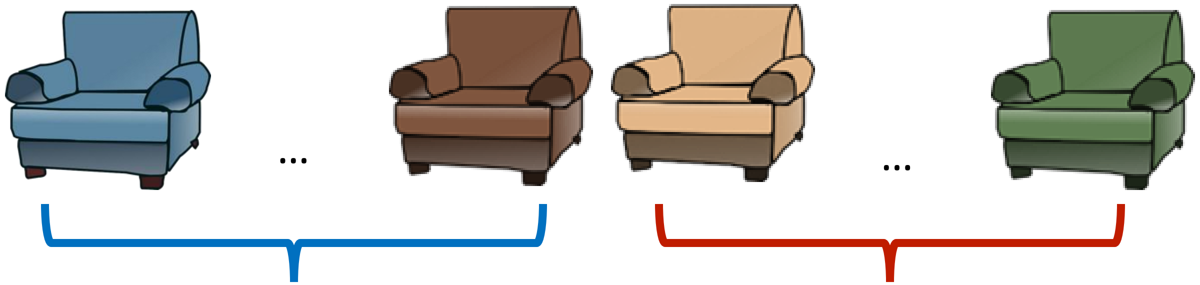
$v_{1,t} = 1/2$

Action space: bid 1 →get the item
bid <1 →get nothing

Exploitability of a Naïve learner [BMSW]

Idea: lure the buyer into bidding high by offering the item for cheap at first, then start charging the item with a high price. It will take a while for the mean-based no regret buyer to realize they should not bid high anymore.





First $T/2$ rounds charge 0 Last $T/2$ rounds charge 1

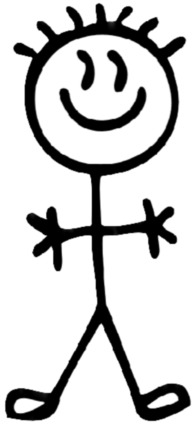
$v_{1,t} = 1/2$

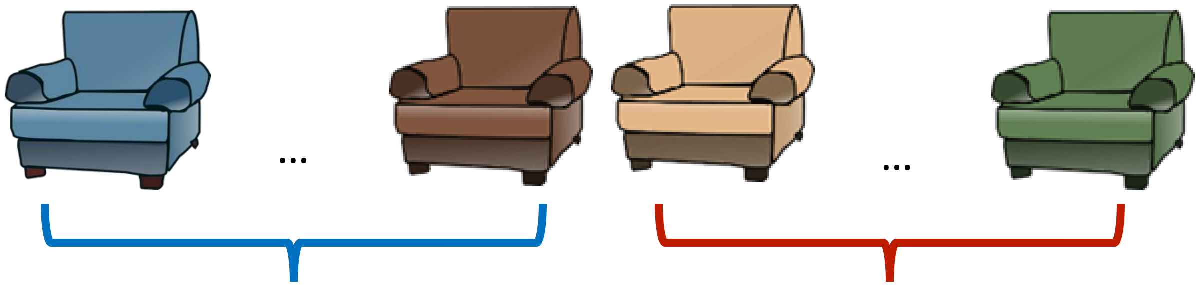
Action space: bid 1 →get the item

bid <1 →get nothing

Exploitability of a Naïve learner [BMSW]

Idea: lure the buyer into bidding high by offering the item for cheap at first, then start charging the item with a high price. It will take a while for the mean-based no regret buyer to realize they should not bid high anymore.





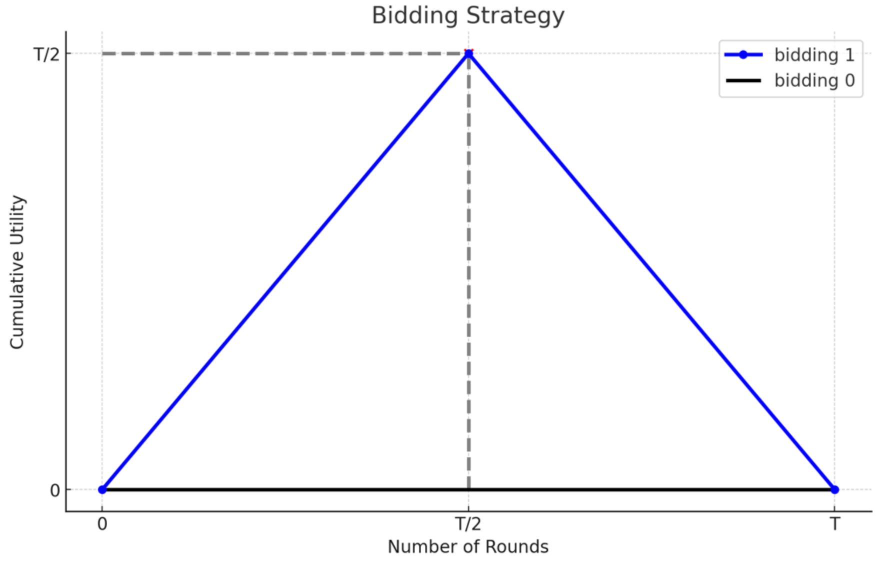
First $T/2$ rounds charge 0

Last $T/2$ rounds charge 1

$v_{1,t} = 1/2$

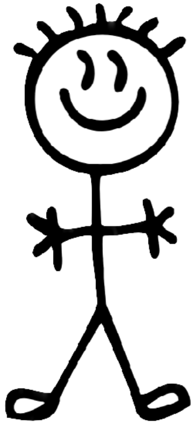
Action space: bid 1 →get the item

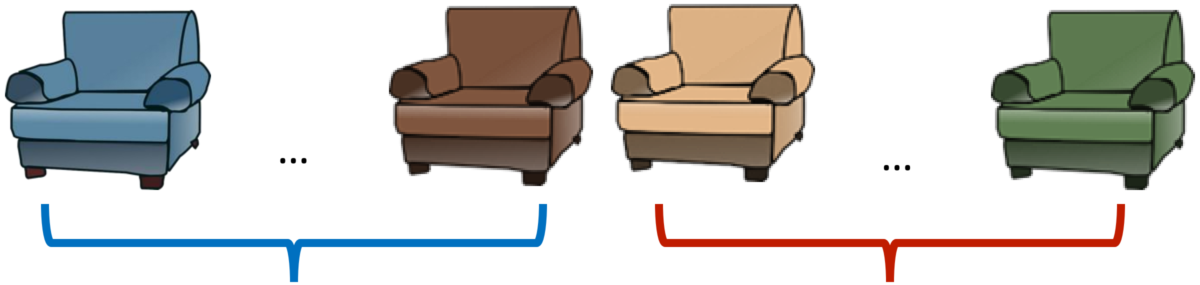
bid <1 →get nothing



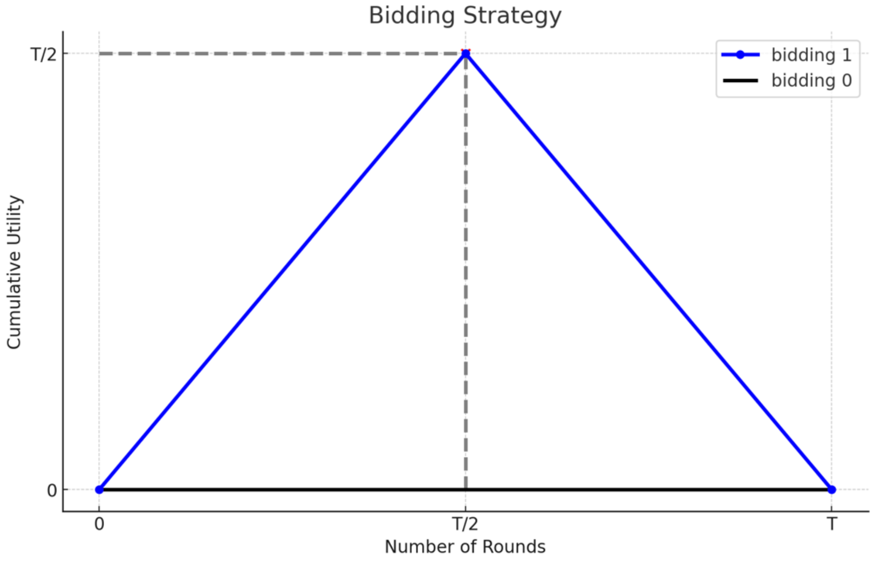
Exploitability of a Naïve learner [BMSW]

Idea: lure the buyer into bidding high by offering the item for cheap at first, then start charging the item with a high price. It will take a while for the mean-based no regret buyer to realize they should not bid high anymore.





First $T/2$ rounds charge 0   Last $T/2$ rounds charge 1



$v_{1,t} = 1/2$

Action space: bid 1 →get the item
bid <1 →get nothing

- Naïve mean-based no regret buyer: bid 1 for all but $o(T)$ rounds, average utility ≈ 0
- Seller revenue : $1 - o(1)$

Exploitability of Multiple Naïve learners

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue ≈ optimal social welfare and buyer gets ≈ zero utility)?

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue $\approx$ optimal social welfare and buyer gets $\approx$ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue $\approx$ optimal social welfare and buyer gets $\approx$ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

Our Idea:

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue $\approx$ optimal social welfare and buyer gets $\approx$ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

Our Idea:

- In the first few rounds, seller set up the utilities so that buyers of different values have maximum utility for different actions.

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue $\approx$ optimal social welfare and buyer gets $\approx$ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

Our Idea:

- In the first few rounds, seller set up the utilities so that buyers of different values have maximum utility for different actions.
- Then, the auction is conducted in several phases.

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue $\approx$ optimal social welfare and buyer gets $\approx$ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

Our Idea:

- In the first few rounds, seller set up the utilities so that buyers of different values have maximum utility for different actions.
- Then, the auction is conducted in several phases.
- In each phase, the seller lures the buyer with the highest value into bidding ~~high~~ an intended value by offering the item for a fair price during first half of the phase, then start charging surcharge in the second half of the phase.

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue ≈ optimal social welfare and buyer gets ≈ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

Our Idea:

- In the first few rounds, seller set up the utilities so that buyers of different values have maximum utility for different actions.
- Then, the auction is conducted in several phases.
- In each phase, the seller lures the buyer with the highest value into bidding ~~high~~ an intended value by offering the item for a fair price during first half of the phase, then start charging surcharge in the second half of the phase.
- Once the no-regret buyers learn to switch to other actions than their intended value at the end of the phase, repeat above process for a different set of intended values.

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue $\approx$ optimal social welfare and buyer gets $\approx$ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

In a Phase with R rounds:

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue ≈ optimal social welfare and buyer gets ≈ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

In a Phase with R rounds:

Buyer value to bid mapping:

$v_1 \rightarrow b_1$

$v_2 \rightarrow b_2$

…

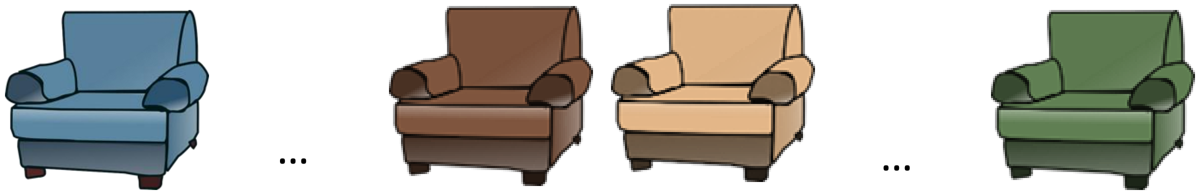
$v_m \rightarrow b_m$

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue ≈ optimal social welfare and buyer gets ≈ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

In a Phase with R rounds:



First $R/2$ rounds, run second price auction on underlying value based on bids

Last $R/2$ rounds, run second price auction with surcharge

Buyer value to bid mapping:

$v_1 \rightarrow b_1$

$v_2 \rightarrow b_2$

…

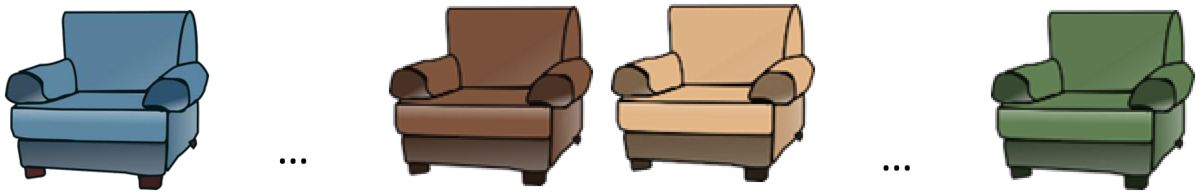
$v_m \rightarrow b_m$

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue $\approx$ optimal social welfare and buyer gets $\approx$ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

In a Phase with R rounds:



First $R/2$ rounds, run second price auction on underlying value based on bids

bids $(b_1, b_2) \Rightarrow$ charge v_1 buyer with v_2

Last $R/2$ rounds, run second price auction with surcharge

Buyer value to bid mapping:

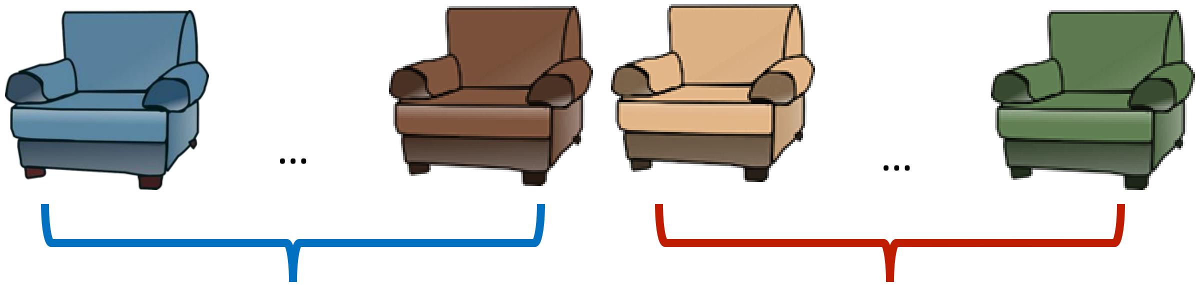
$v_1 \to b_1$
$v_2 \to b_2$
…
$v_m \to b_m$

Exploitability of Multiple Naïve learners

Question: Is full surplus extraction still possible (seller gets revenue ≈ optimal social welfare and buyer gets ≈ zero utility)?

Challenge for the seller: when buyers with different value for the item choose the same bid, the seller cannot treat them differently

In a Phase with R rounds:



First $R/2$ rounds, run second price auction on underlying value based on bids

bids $(b_1, b_2) \Rightarrow$ charge v_1 buyer with v_2

Last $R/2$ rounds, run second price auction with surcharge

bids $(b_1, b_2) \Rightarrow$ charge v_1 buyer with $v_2 + 2(v_1 - v_2)$

Buyer value to bid mapping:

$v_1 \rightarrow b_1$

$v_2 \rightarrow b_2$

…

$v_m \rightarrow b_m$

Future Work

Future Work

- Can we understand optimal revenue in the non-overbidding learning buyer setting?

Future Work

- Can we understand optimal revenue in the non-overbidding learning buyer setting?
- What would happen if the buyers are a mixture over different types of learning (some more naïve than others)?

Future Work

- Can we understand optimal revenue in the non-overbidding learning buyer setting?
- What would happen if the buyers are a mixture over different types of learning (some more naïve than others)?
- In the Ads auction setting, the sellers are typically bidding for multiple different keywords simultaneously. What would happen in the multiple item multiple buyer case?

THANK YOU!