Bayesian Nash Equilibrium in First-Price Auctions with Discrete Value Distributions

Introduction To Game Theory - Spring 2021 - Reading Assignment

Team 17

International Institute of Information Technology, Hyderabad

April 2021

Team Members

- **Mentor:**
    - Dr. Sujit Gujar (Course Instructor)
- **Students:**
    - Khadiravana
    - Anurag
    - Krishnashis Banerjee
    - Akshat Chhajer
    - Abhyudit Mohla

Model

- Seller : No private information

Model

- Seller : No private information
- 1 item for Sale

Model

- Seller : No private information
- 1 item for Sale
- N : set of potential buyers, $N = \{1,2,...,n\}$

Model

- Seller : No private information
- 1 item for Sale
- N : set of potential buyers, N $= \{1,2,...,n\}$
- Every buyer has a private value for the item.

Model

- Seller : No private information
- 1 item for Sale
- N : set of potential buyers, $N = \{1,2,...,n\}$
- Every buyer has a private value for the item.
- The private value for each buyer is drawn from a **publicly** known value distribution which is **discrete**.

Model

- Seller : No private information
- 1 item for Sale
- N : set of potential buyers, $N = \{1,2,\ldots,n\}$
- Every buyer has a private value for the item.
- The private value for each buyer is drawn from a **publicly** known value distribution which is **discrete**.
- Value Support of buyer i is $\{v_i^1, v_i^2, \ldots, v_i^{d_i}\}$.

Model

- Seller : No private information
- 1 item for Sale
- N : set of potential buyers, $N = \{1,2,...,n\}$
- Every buyer has a private value for the item.
- The private value for each buyer is drawn from a **publicly** known value distribution which is **discrete**.
- Value Support of buyer i is $\{v_i^1, v_i^2, ..., v_i^{d_i}\}$.
- Without loss of generality, let $0 \leq v_i^1 < v_i^2 < .... < v_i^{d_i}$

Model

- Seller : No private information
- 1 item for Sale
- N : set of potential buyers, N = $\{1,2,...,n\}$
- Every buyer has a private value for the item.
- The private value for each buyer is drawn from a **publicly** known value distribution which is **discrete**.
- Value Support of buyer i is $\{v_i^1, v_i^2, ..., v_i^{d_i}\}$.
- Without loss of generality, let $0 \leq v_i^1 < v_i^2 < .... < v_i^{d_i}$
- Let G_i denote the CDF of the **value** distributions of player i. i.e., $G_i(v) = \text{Prob}(v_i \leq v)$

Model

- Seller : No private information
- 1 item for Sale
- N : set of potential buyers, N $= \{1,2,\ldots,n\}$
- Every buyer has a private value for the item.
- The private value for each buyer is drawn from a **publicly** known value distribution which is **discrete**.
- Value Support of buyer i is $\{v_i^1, v_i^2, \ldots, v_i^{d_i}\}$.
- Without loss of generality, let $0 \leq v_i^1 < v_i^2 < \ldots. < v_i^{d_i}$
- Let G_i denote the CDF of the **value** distributions of player i. i.e., $G_i(v) = \text{Prob}(v_i \leq v)$
- Similarly, let $F_i(b)$ denote the CDF of buyer i's bids.

Model Contd ..

- **Assumptions:**

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \in N$

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \in N$
  - $b_i \geq 0, \forall i \in N$

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \in N$
  - $b_i \geq 0, \forall i \in N$
  - Bids are placed simultaneously

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \in N$
  - $b_i \geq 0, \forall i \in N$
  - Bids are placed simultaneously
- **Rules:**

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \in N$
  - $b_i \geq 0, \forall i \in N$
  - Bids are placed simultaneously
- **Rules:**
  - **Allocation:** Buyer with the largest bid

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \ \in N$
  - $b_i \geq 0, \forall i \ \in N$
  - Bids are placed simultaneously
- **Rules:**
  - **Allocation:** Buyer with the largest bid
  - **Payment:** His bid, i.e. the largest bid.

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \in N$
  - $b_i \geq 0, \forall i \in N$
  - Bids are placed simultaneously
- **Rules:**
  - **Allocation:** Buyer with the largest bid
  - **Payment:** His bid, i.e. the largest bid.
- **Strategy:**

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \in N$
  - $b_i \geq 0, \forall i \in N$
  - Bids are placed simultaneously
- **Rules:**
  - **Allocation:** Buyer with the largest bid
  - **Payment:** His bid, i.e. the largest bid.
- **Strategy:**
  - $S_i(v_i^j)$: Set of all possible bids for player i for his j^{th} valuation , v_i^j. Closed Set.

Model Contd ..

- **Assumptions:**
  - Buyers have quasi-linear utilities
  - No buyer overbids, i.e. $b_i \leq v_i, \forall i \ \in N$
  - $b_i \geq 0, \forall i \ \in N$
  - Bids are placed simultaneously
- **Rules:**
  - **Allocation:** Buyer with the largest bid
  - **Payment:** His bid, i.e. the largest bid.
- **Strategy:**
  - $S_i(v_i^j)$: Set of all possible bids for player i for his j^{th} valuation , v_i^j. Closed Set.
  - $S_i = \bigcup_j S_i(v_i^j)$

Recall

- The strategies, $S^* = s_1^* \times s_2^* \times s_3^* ... \times s_N^*$ form a **Bayesian Nash Equilibrium** if no bidder has an incentive to change his strategy unilaterally in the Bayesian setting.

Recall

- The strategies, $S^* = s_1^* \times s_2^* \times s_3^* ... \times s_N^*$ form a **Bayesian Nash Equilibrium** if no bidder has an incentive to change his strategy unilaterally in the Bayesian setting.
- That is by *unilaterally* deviating from s_i^* , bidder i's **expected** utility will not increase.

Recall

- The strategies, $S^* = s_1^* \times s_2^* \times s_3^* ... \times s_N^*$ form a **Bayesian Nash Equilibrium** if no bidder has an incentive to change his strategy unilaterally in the Bayesian setting.
- That is by *unilaterally* deviating from s_i^* , bidder i's **expected** utility will not increase.
- In the continuous case, the strategy maps each buyer's value to **a single** bid.

Recall

- The strategies, $S^* = s_1^* \times s_2^* \times s_3^* ... \times s_N^*$ form a **Bayesian Nash Equilibrium** if no bidder has an incentive to change his strategy unilaterally in the Bayesian setting.
- That is by *unilaterally* deviating from s_i^* , bidder i's **expected** utility will not increase.
- In the continuous case, the strategy maps each buyer's value to **a single** bid.
- Where as in the discrete setting, each buyer's strategy is randomized and maps a particular value to a **set** of bids with a certain probability distribution.

Objective

- Our main objective is to find the bidding strategies that constitute a BNE.
- That is we want to map each value of every bidder to a set of possible bids and also find the probability distribution function for each of those bids.

Tie Breaking Rule

Tie Breaking Rule

Ties are broken by running a Vickrey auction among the highest bidders.

Tie Breaking Rule

Tie Breaking Rule

Ties are broken by running a Vickrey auction among the highest bidders.

- For the discrete case, randomized allocation in case of a tie could lead to the non-existence of a BNE. In order to guarantee BNE, Maskin and Riley[19] have introduced this alternative tie breaking rule.

Tie Breaking Rule

Tie Breaking Rule

Ties are broken by running a Vickrey auction among the highest bidders.

- For the discrete case, randomized allocation in case of a tie could lead to the non-existence of a BNE. In order to guarantee BNE, Maskin and Riley[19] have introduced this alternative tie breaking rule.
- Vickrey Auction – Second Price Auction

Tie Breaking Rule

Tie Breaking Rule

Ties are broken by running a Vickrey auction among the highest bidders.

- For the discrete case, randomized allocation in case of a tie could lead to the non-existence of a BNE. In order to guarantee BNE, Maskin and Riley[19] have introduced this alternative tie breaking rule.
- Vickrey Auction – Second Price Auction
- Since it is truthful , the seller will get to know the true valuations of each bidder.

Tie Breaking Rule

Tie Breaking Rule

Ties are broken by running a Vickrey auction among the highest bidders.

- For the discrete case, randomized allocation in case of a tie could lead to the non-existence of a BNE. In order to guarantee BNE, Maskin and Riley[19] have introduced this alternative tie breaking rule.
- Vickrey Auction – Second Price Auction
- Since it is truthful , the seller will get to know the true valuations of each bidder.
- Bidder with the highest valuation gets the object.

Winning Bid

Definition

Given the strategies of all the bidders, if any buyer can win the object with a certain probability by bidding b, then b is called the winning bid.

Lemma 1: Minimum Winning Bid

Lemma 1

Assume buyer i^* has the largest smallest value, i.e., $v^1_{i^*} = max_j v^1_j$. Then the smallest winning bid $\underline{b}$ is

$$\underline{b} = \arg\max_b (v^1_{i^*} - b) \prod_{i \neq i^*} G_i(b) \tag{1}$$

,
where $G_i(v) = \text{Prob}(v_i \leq v)$

- Recall that our value support had values of player i in increasing order.
- The smallest bid placed by player i^* will be exactly $\underline{b}$. Because:
  1. $\underline{b}$ has to be greater than or equal to the smallest bid of at least one of the bidders. Otherwise $\underline{b}$ will not be a *winning bid*. Intuitively, there will be no incentive for other bidders to bid $\underline{b}$.
  2. $\underline{b}$ will have to less than or equal to $v^1_{i^*}$, otherwise player i^* will be overbidding and his utility would be negative.

Computation of Minimum Winning Bid

- Compute and store $v^1_{i^*}$ and i^* using **Lemma 1**
- For each bidder, compute the utility of each of their bids which are less than or equal to $v^1_{i^*}$
- Report bid b as $\underline{\text{b}}$ if the expected utility of player i^* is maximum on bidding b. Note that on bidding $\underline{\text{b}}$, player i^* will get his maximum expected utility among lowest possible bids $b \leq v^j_{i^*}$.

Example

Suppose there are 4 buyers with the following discrete value distributions:

- $$G_1(x) = \begin{cases} 1 & x = 20 \\ \frac{11\sqrt{7}}{24\sqrt{3}} & x = 10 \\ \frac{\sqrt{77}}{12\sqrt{2}} & x = 2 \end{cases}$$
- $$G_2(x) = \begin{cases} 1 & x = 14 \\ \frac{4}{\sqrt{21}} & x = 13 \\ \frac{2\sqrt{22}}{7\sqrt{7}} & x = 1 \end{cases}$$
- $$G_3(x) = \begin{cases} 1 & x = 20 \\ \frac{11}{12} & x = 9 \end{cases}$$
- $$G_4(x) = \begin{cases} 1 & x = 12 \\ \frac{3\sqrt{3}}{2\sqrt{7}} & x = 1 \end{cases}$$

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}$,$v_4 = \{1, 12\}$

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}$,$v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore$ $i^* = 3$ and $v_3^1 = 9$

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}$,$v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2$,$v_2^1 = 1$,$v_4^1 = 1$.

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}$,$v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore$ $i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2$,$v_2^1 = 1$,$v_4^1 = 1$.
- Our $\underline{b}$ will be one of v_1^1, v_2^1, v_4^1. Which one it will exactly be will be computed using Lemma 1.

b	$(v_3^1 - b) \prod_{i \neq 3} G_i(b)$
$v_1^1 = 2$	

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}$,$v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2$,$v_2^1 = 1$,$v_4^1 = 1$.
- Our $\underline{b}$ will be one of v_1^1, v_2^1, v_4^1. Which one it will exactly be will be computed using Lemma 1.

b	$(v_3^1 - b)\prod_{i \neq 3} G_i(b)$
$v_1^1 = 2$	$(9-2) \times (\frac{\sqrt{77}}{12\sqrt{2}} \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{1.8003}$

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}$,$v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2$,$v_2^1 = 1$,$v_4^1 = 1$.
- Our $\underline{b}$ will be one of v_1^1, v_2^1, v_4^1. Which one it will exactly be will be computed using Lemma 1.

b	$(v_3^1 - b)\prod_{i \neq 3} G_i(b)$
$v_1^1 = 2$	$(9-2) \times (\frac{\sqrt{77}}{12\sqrt{2}} \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{1.8003}$
$v_2^1 = 1$	$(9-1) \times (0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}, v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2, v_2^1 = 1, v_4^1 = 1$.
- Our $\underline{b}$ will be one of v_1^1, v_2^1, v_4^1. Which one it will exactly be will be computed using Lemma 1.

b	$(v_3^1 - b)\prod_{i \neq 3} G_i(b)$
$v_1^1 = 2$	$(9-2) \times (\frac{\sqrt{77}}{12\sqrt{2}} \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{1.8003}$
$v_2^1 = 1$	$(9-1) \times (0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$
$v_3^1 = 1$	$(9-1) \times (0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$

Example : Computing $\underline{b}$

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}, v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2, v_2^1 = 1, v_4^1 = 1$.
- Our $\underline{b}$ will be one of v_1^1, v_2^1, v_4^1. Which one it will exactly be will be computed using Lemma 1.

b	$(v_3^1 - b)\prod_{i\neq 3} G_i(b)$
$v_1^1 = 2$	$(9-2)\times(\frac{\sqrt{77}}{12\sqrt{2}} \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{1.8003}$
$v_2^1 = 1$	$(9-1)\times(0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$
$v_3^1 = 1$	$(9-1)\times(0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$

Table: Computation of $\underline{b}$

Example : Computing $\underline{b}$ (Contd...)

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}, v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore$ $i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2, v_2^1 = 1, v_4^1 = 1$.
- Our $\underline{b}$ will be one of v_1^1, v_2^1, v_4^1. Which one it will exactly be will be computed using Lemma 1.

b	$(v_3^1 - b)\prod_{i \neq 3} G_i(b)$
$v_1^1 = 2$	$(9-2) \times (\frac{\sqrt{77}}{12\sqrt{2}} \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{1.8003}$
$v_2^1 = 1$	$(9-1) \times (0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$
$v_3^1 = 1$	$(9-1) \times (0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$

Table: Computation of $\underline{b}$

Example : Computing $\underline{b}$ (Contd...)

- $v_1 = \{2, 10, 20\}$, $v_2 = \{1, 13, 14\}$, $v_3 = \{9, 20\}, v_4 = \{1, 12\}$
- $i^* = \arg\max_j v_j^1$, $\therefore i^* = 3$ and $v_3^1 = 9$
- Values that are less than or equal to v_3^1 are, $v_1^1 = 2, v_2^1 = 1, v_4^1 = 1$.
- Our $\underline{b}$ will be one of v_1^1, v_2^1, v_4^1. Which one it will exactly be will be computed using Lemma 1.

b	$(v_3^1 - b)\prod_{i\neq 3} G_i(b)$
$v_1^1 = 2$	$(9-2) \times (\frac{\sqrt{77}}{12\sqrt{2}} \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{1.8003}$
$v_2^1 = 1$	$(9-1) \times (0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$
$v_3^1 = 1$	$(9-1) \times (0 \times \frac{2\sqrt{22}}{7\sqrt{7}} \times \frac{3\sqrt{3}}{2\sqrt{7}}) = \mathbf{0}$

Table: Computation of $\underline{b}$

- Therefore, our $\underline{\mathbf{b}} = \mathbf{2}$ using **Lemma 1**.

Lemma 2

Lemma 2

At BNE, for any winning bid b, there are at least two players i and j such that, $b \in S_i$ and $b \in S_j$.

Lemma 2

Lemma 2

At BNE, for any winning bid b, there are at least two players i and j such that, $b \in S_i$ and $b \in S_j$.

- We can prove this by intuition.

Lemma 2

Lemma 2

At BNE, for any winning bid b, there are at least two players i and j such that, $b \in S_i$ and $b \in S_j$.

- We can prove this by intuition.
- Note that we are talking about a situation at equilibrium.

Lemma 2

Lemma 2

At BNE, for any winning bid b, there are at least two players i and j such that, $b \in S_i$ and $b \in S_j$.

- We can prove this by intuition.
- Note that we are talking about a situation at equilibrium.
- If there is no other player who bids b with some probability then player i has the incentive to bid $b - \epsilon$ and increase his utility. So if BNE exists and $b \in S_i$ then $\exists\ j \in N - i$, s.t. $b \in S_j$.

Lemma 3

Lemma 3

For any buyer i, there is no mass point above $\underline{b}$ in buyer i's bid distribution.

- This simply implies that a buyer would not give a *particular* bid positive probability of winning when that bid is larger than $\underline{b}$.

Lemma 4

Lemma 4

For each buyer, his bidding strategy is monotone in value , i.e., max $S_i(v_i^j) \leq$ min $S_i(v_i^{j+1})$ for $v_i^j \geq \underline{b}$.

Algorithm : Computing BNE

- We have computed $\underline{b}$ ussing **Lemma 1**.
- For the computation of the bid-distributions, i.e., F_i, we will need $\overline{b}$. i.e., we will need the maximum possible bid that will be placed by any buyer.
- Unfortunately there is no straight forward method for the computation of $\overline{b}$ like we had for the computation of $\underline{b}$.
- The way we are going to solve this issue is first starting with a guess of $\overline{b}$ and then computing the bid distributions for each bidder from $\overline{b}$.
- If $\underline{b}(\overline{b}) = \underline{b}$, then we stop.
- This kind of approach is generally known as **backward-shooting**. That is we start from the maximum and compute our way down until we converge.

Algorithm : Backward Shooting Algorithm

Algorithm 1 Backward Shooting Algorithm

Input: Buyer's Value distributions G_i

Output: Buyer's bid distributions F_i

Compute the smallest winning bid $\underline{\mathrm{b}}$ using **Lemma 1**;
$UB \leftarrow \max\{\bigcup_{i \in N} supp(G_i)\}$, $LB \leftarrow 0$;
while some exit condition is not met **do**
    $\underline{b} \leftarrow \frac{1}{2}(UB + LB)$
    Compute F_i all the way down from $\overline{b}$ to the corresponding smallest winning bid $\underline{\mathrm{b}}(\overline{b})$
    if $\underline{b}(\overline{b}) > \underline{b}$ **then**
        $UB \leftarrow \overline{b}$;
    else
        $LB \leftarrow \overline{b}$;
    end if
end while
return F_i

Computation of $\underline{\mathrm{b}}(\overline{b})$

- Algorithm 1 terminates when $\underline{\mathrm{b}}(\overline{b}) = \underline{\mathrm{b}}$ computer from **Lemma 1**.
- But how do we compute $\underline{\mathrm{b}}(\overline{b})$?
- In the continous case the computation of $\underline{\mathrm{b}}(\overline{b})$ given $\overline{b}$ is done using differential-equations. But such a methodology does not work in the discrete case.
- For the computation of $\underline{\mathrm{b}}(\overline{b})$ in the discrete case the paper introduces some new data structures and new concepts.

Bidding Set and Waiting List

Bidding Set

The set of buyers whose bidding strategies include bid x is called the **bidding set**, denoted by $\Lambda(\mathrm{x})$, i.e., $\Lambda(\mathrm{x}) = \{i \mid x \in S_i\}$

Bidding Set and Waiting List

Bidding Set

The set of buyers whose bidding strategies include bid x is called the **bidding set**, denoted by $\Lambda(\mathrm{x})$, i.e., $\Lambda(\mathrm{x}) = \{i \mid x \in S_i\}$

Waiting List

The set of other buyers $N - \Lambda(\mathrm{x})$ is called the **waiting list** at bid x.

Bidding Set and Waiting List

Bidding Set

The set of buyers whose bidding strategies include bid x is called the **bidding set**, denoted by $\Lambda(\mathrm{x})$, i.e., $\Lambda(\mathrm{x}) = \{i \mid x \in S_i\}$

Waiting List

The set of other buyers $N - \Lambda(\mathrm{x})$ is called the **waiting list** at bid x.

Cummulative Bid Distribution

The product of bid distributions of buyers in the **bidding set**. Denoted by $F_\Lambda(x) = \Pi_{i \in \Lambda(x)} F_i(x)$

Notations

- When there is no ambiguity about the bid point x of interest, we will represent $\Lambda(x)$ as Λ.
- $v_i(x)$: represents the value of player i when he bids x in the equilibrium.

Theorem 1

Theorem 1

Suppose $\Lambda(x)$ does not change in bid interval (b_1, b_2), and $v_i(x)$ is constant for $x \in (b_1, b_2)$, $i \in \Lambda(x)$. Then the bid distribution of every buyer in Λ is differentiable in (b_1, b_2). For any $x \in (b_1, b_2)$, we have

$$\frac{f_i(x)}{F_i(x)} = h_i(x), \forall i \in \Lambda$$

Where $f_i(x)$ represents the **PDF** and $F_i(x)$ represents the **CDF** of the bid distribution of bidder i and $h_i(x)$ is defined as follows:

$$h_i(x) = \frac{1}{|\Lambda(x)| - 1} \left(\sum_{j \in \Lambda(x)} \frac{1}{v_j - x} \right) - \frac{1}{v_i - x}$$

Theorem 1: Proof

- For any bid $x \in (b_1, b_2)$, the utility of bidder i is:

$$u_i(v_i) = (v_i - x)F_{N\setminus\Lambda}(x)F_{\Lambda\setminus i}(x), \forall i \in \Lambda \tag{2}$$

- Buyers in the waiting list $N \setminus \Lambda$ do not bid in (b_1, b_2). $\therefore$ we have, $F_{N\setminus\Lambda}(x) = F_{N\setminus\Lambda}(b_1)$
- Note that $\Pi_{i\in\Lambda}F_{\Lambda\setminus i}(x) = (F_\Lambda(x))^{|\Lambda|-1}$
- Therefore after multiplication over $i \in \Lambda$ equation 2 becomes,

$$\Pi_{i\in\Lambda}u_i(v_i) = \Pi_{i\in\Lambda}(v_i - x)(F_{N\setminus\Lambda}(b_1))^{|\Lambda|}(F_\Lambda(x))^{|\Lambda|-1}, \forall x \in \Lambda \tag{3}$$

- using equation 2 , we can deduce $F_i(x)$ as :

$$F_i(x) = \frac{v_i - x}{u_i(v_i)}\left(\Pi_{i\in\Lambda}\frac{u_i(v_i)}{v_i - x}\right)^{\frac{1}{|\Lambda|-1}}(F_{N\setminus\Lambda}(x))^{\left(-\frac{1}{|\Lambda|-1}\right)} \tag{4}$$

Theorem 1: Proof

- Differentiating on LHS and RHS of equation 4 will give us the desired result, i.e.,

$$f_i(x) = \left[\frac{1}{|\Lambda| - 1} \left(\sum_{j \in \Lambda} \frac{1}{v_j - x} \right) - \frac{1}{v_i - x} \right] F_i(x)$$

$$f_i(x) = h_i(x) F_i(x)$$

- $f_i(x)$: **PDF** of the bid distribution of player i.
- $F_i(x)$: **CDF** of the bid distribution of player i.
- Notice that $f_i(x)$ was derived by differentiating $F_i(x)$ in equation 4

Theorem 2

Theorem 2

$S_i(v_i^j)$ is an interval when $v_i^j \geq \underline{b}$.

Theorem 2

Theorem 2

$S_i(v_i^j)$ is an interval when $v_i^j \geq \underline{b}$.

- Basically says that bids of the values of a player i will lie in an interval if his valuation is greater than the minimum winning bid. Similar to **Lemma 3**.

Theorem 2

Theorem 2

$S_i(v_i^j)$ is an interval when $v_i^j \geq \underline{b}$.

- Basically says that bids of the values of a player i will lie in an interval if his valuation is greater than the minimum winning bid. Similar to **Lemma 3**.
- **Proof:** Quite Lengthy. Appendix A of the Paper

Change Points of Λ

Definition 2

When the bidding set changes at x, we use $\Lambda^+(x)$ and $\Lambda^-(x)$ to denote the buyers who bid in the upper neighbourhood and lower neighbourhood around x, i.e.,

$$\Lambda^+(x) = \{i \mid \exists\epsilon > 0, (x, x+\epsilon) \subseteq S_i\}$$

$$\Lambda^-(x) = \{i \mid \exists\epsilon > 0, (x-\epsilon, x) \subseteq S_i\}$$

Example

Recall our Example.

Suppose there are 4 buyers with the following discrete value distributions:

- $$G_1(x) = \begin{cases} 1 & x = 20 \\ \frac{11\sqrt{7}}{24\sqrt{3}} & x = 10 \\ \frac{\sqrt{77}}{12\sqrt{2}} & x = 2 \end{cases}$$

- $$G_2(x) = \begin{cases} 1 & x = 14 \\ \frac{4}{\sqrt{21}} & x = 13 \\ \frac{2\sqrt{22}}{7\sqrt{7}} & x = 1 \end{cases}$$

- $$G_3(x) = \begin{cases} 1 & x = 20 \\ \frac{11}{12} & x = 9 \end{cases}$$

- $$G_4(x) = \begin{cases} 1 & x = 12 \\ \frac{3\sqrt{3}}{2\sqrt{7}} & x = 1 \end{cases}$$

Example

- Turns out that using the algorithms used in the paper, we get the following $F_i(x)$ for this case:

- $$F_1(x) = \begin{cases} \dfrac{11}{20-x} & x \in (8,9] \\ \dfrac{11}{12}\sqrt{\dfrac{2(20-x)}{(14-x)(12-x)}} & x \in (6,8] \\ \dfrac{77}{48}\sqrt{\dfrac{10-x}{(9-x)(13-x)}} & x \in [2,6] \end{cases}$$

- $$F_2(x) = \begin{cases} \sqrt{\dfrac{8(14-x)}{(20-x)(12-x)}} & x \in (6,8] \\ \dfrac{8}{7}\sqrt{\dfrac{13-x}{(10-x)(9-x)}} & x \in [2,6] \\ \dfrac{2\sqrt{22}}{7\sqrt{7}} & x = 1 \end{cases}$$

Example

- Similarly

- $$F_3(x) = \begin{cases} \dfrac{11}{20-x} & x \in [8,9] \\ \dfrac{11}{6}\sqrt{\dfrac{7(9-x)}{3(10-x)(13-x)}} & x \in [2,6] \end{cases}$$

- $$F_4(x) = \begin{cases} \sqrt{\dfrac{18(12-x)}{(20-x)(14-x)}} & x \in [6,8] \\ \dfrac{3\sqrt{3}}{2\sqrt{7}} & x = 1 \end{cases}$$

Example

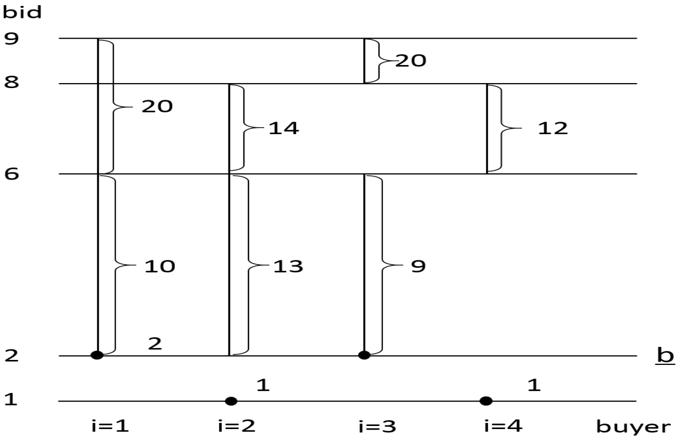

Figure: Caption

Change Points of Λ : Example

- $\Lambda(8.5)$=?
  - $\Lambda(8.5) = \{i \mid 8.5 \in S_i\}$. $\implies \Lambda(8.5) = \{1, 3\}$
- $\Lambda^+(6)$ =?
  - $\Lambda^+(6) = \{i \mid \exists\epsilon > 0, (6, 6 + \epsilon) \subseteq S_i\}$. $\implies \Lambda^+(6) = \{1, 2, 4\}$
- $\Lambda^-(6)$ =?
  - $\Lambda^-(6) = \{i \mid \exists\epsilon > 0, (6 - \epsilon, 6) \subseteq S_i\}$. $\implies \Lambda^-(6) = \{1, 2, 3\}$
- **Waiting List** at x = 8.5 ?
  - $N - \Lambda(8.5) = \{1, 2, 3, 4\} - \{1, 3\} = \{2, 4\}$

Change Points of Λ Contd..

- Notice that $\Lambda(x)$ changes if the bidding interval $S_i(v_i^j)$ either stars or ends **at** x.
- Therefore to compute the changes in the bidding set it suffices to compute the starting and ending points of $S_i(v_i^j)$, $\forall i \forall j$.
- Since method falls in the **backward shooting category** we will analyze the bids from maximum possible bid to the minimum possible bid.

Change Points of Λ Contd..

Definition 3

We say a buyer enters the bidding set with value v_i at bid x if $x = max\ S_i(v_i)$. Similarly we say a buyer leaves the bidding set if $x = min\ S_i(v_i)$

Change Points of Λ Contd..

Definition 3

We say a buyer enters the bidding set with value v_i at bid x if $x = max\ S_i(v_i)$. Similarly we say a buyer leaves the bidding set if $x = min\ S_i(v_i)$

- Entering the bidding set with value i at x is not same as saying $i \in \Lambda^+(x)$ and $i \in \Lambda^-(x)$

Change Points of Λ Contd..

Definition 3

We say a buyer enters the bidding set with value v_i at bid x if $x = max\ S_i(v_i)$. Similarly we say a buyer leaves the bidding set if $x = min\ S_i(v_i)$

- Entering the bidding set with value i at x is not same as saying $i \in \Lambda^+(x)$ and $i \in \Lambda^-(x)$
- Since a bidder might leave the bid set at x with some value v_i^j and enter it with another value v_i^{j-1}. His value has changed and his bidding set has changed but his entry and exit points have not.

Example

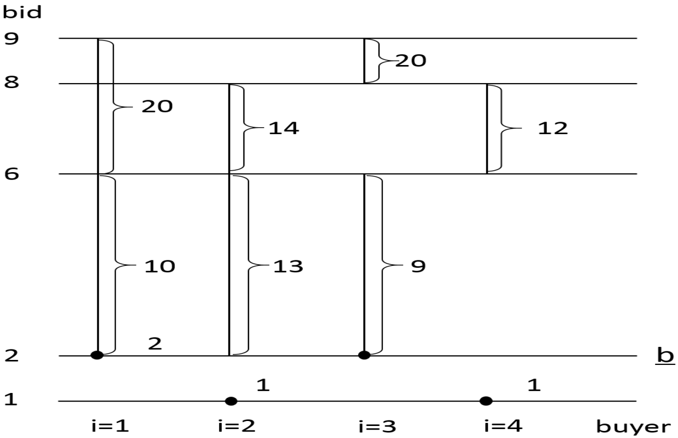

Figure: Bid Chart

Entry and Exit Points of Λ

Lemma 5

If buyer i with value v_i enters the bidding set at point b, then we have

$$\frac{1}{|\Lambda^+(b)| - 1}\left(\sum_{j \in \Lambda^+(b)} \frac{1}{v_j - x}\right) \leq \frac{1}{v_i - x} \forall x \in (b, b+\epsilon)$$

$$\frac{1}{|\Lambda^-(b)| - 1}\left(\sum_{j \in \Lambda^-(b)} \frac{1}{v_j - x}\right) \geq \frac{1}{v_i - x} \forall x \in (b-\epsilon, b)$$

- The first inequality is because buyer i has no extra incentive to bid $b + \epsilon$.

Entry and Exit Points of Λ

Lemma 5

If buyer i with value v_i enters the bidding set at point b, then we have

$$\frac{1}{|\Lambda^+(b)|-1}\left(\sum_{j\in\Lambda^+(b)}\frac{1}{v_j-x}\right)\leq\frac{1}{v_i-x}\forall x\in(b,b+\epsilon)$$

$$\frac{1}{|\Lambda^-(b)|-1}\left(\sum_{j\in\Lambda^-(b)}\frac{1}{v_j-x}\right)\geq\frac{1}{v_i-x}\forall x\in(b-\epsilon,b)$$

- The second inequality is because of **Theorem 1**. From **Theorem 1** we have $f_i(x)=h_i(x)F_i(x)$. Any **valid pdf** has to have a **non-negative area less than 1 for any interval**. Therefore, $f_i(x)\geq 0$ and since $F_i(x)\geq 0$ by definition (it is a **cdf**), it implies that $h_i(x)\geq 0$.

Virtual Value $\phi^*(x)$

Definition 4

Given the bid x, we define virtual value $\phi^*(x)$ as a number that satisfies the following equation:

$$\frac{1}{|\Lambda(x)|-1}\left(\sum_{i\in\Lambda(x)}\frac{1}{v_i-x}\right)=\frac{1}{\phi^*(x)-x}.$$

Virtual Value $\phi^*(x)$

Definition 4

Given the bid x, we define virtual value $\phi^*(x)$ as a number that satisfies the following equation:

$$\frac{1}{|\Lambda(x)|-1}\left(\sum_{i\in\Lambda(x)}\frac{1}{v_i-x}\right)=\frac{1}{\phi^*(x)-x}.$$

- Notice that since $\phi^*(x)$ is based on $\Lambda(x)$, when $\Lambda(x)$ changes, $\phi^*(x)$ also changes.

Virtual Value $\phi^*(x)$

Definition 4

Given the bid x, we define virtual value $\phi^*(x)$ as a number that satisfies the following equation:

$$\frac{1}{|\Lambda(x)|-1}\left(\sum_{i\in\Lambda(x)}\frac{1}{v_i-x}\right)=\frac{1}{\phi^*(x)-x}.$$

- Notice that since $\phi^*(x)$ is based on $\Lambda(x)$, when $\Lambda(x)$ changes, $\phi^*(x)$ also changes.

Theorem 3

$\phi^*(x)$ strictly decreases with respect to x.

When to enter $\Lambda(x)$

Theorem 4

Suppose buyer i has the largest unconsumed value v_i in the waiting list, he will enter the bidding set immediately when either one of the following two conditions is satisfied:

- $|\Lambda| \leq 1$ and $v_i > x$; or,
- $h_i(x) \geq 0$ and $v_i > x$

Recall,

$$h_i(x) = \frac{1}{|\Lambda(x)| - 1} \left(\sum_{j \in \Lambda(x)} \frac{1}{v_j - x} \right) - \frac{1}{v_i - x}$$

When to exit $\Lambda(x)$

Theorem 5

Buyer i with value v_i^k leaves the bidding set at x when the cumulative probability of the bidding set equals the probability of the value, i.e., $F_i(max\ S_i(v_i^k)) - F_i(x) = G_i(v_i^k) - G_i(v_i^{k-1})$.

- $G_i(x)$: cdf of player i's **values**.
- $F_i(x)$: cdf of player i's **bids**.

Example

Recall our running example

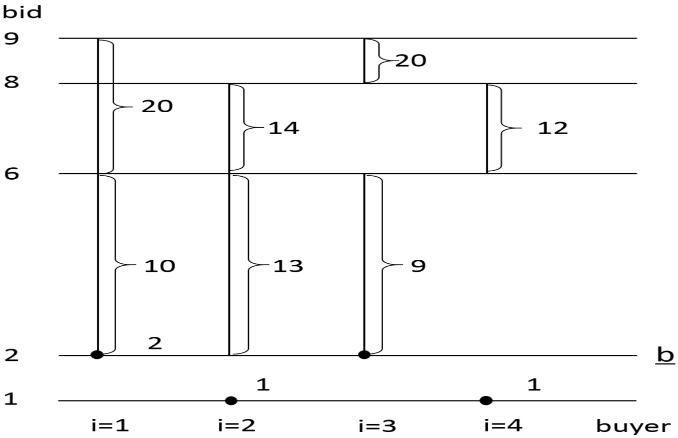

Figure: Bid Chart

Example

- Let us consider $S_1(20)$ from the bid chart.
- It begins at bid 9 and ends at bid $= 6$.
- Using **Theorem 5**, bidder 1 will only exit at bid 6 if $F_1(9) - F_1(6) = G_1(20) - G_1(10)$:
- $F_1(x)$ computed by our algorithms for this example is :

$$F_1(x) = \begin{cases} \dfrac{11}{20-x} & x \in (8,9] \\ \dfrac{11}{12}\sqrt{\dfrac{2(20-x)}{(14-x)(12-x)}} & x \in (6,8] \\ \dfrac{77}{48}\sqrt{\dfrac{10-x}{(9-x)(13-x)}} & x \in [2,6] \end{cases}$$

$$\implies F_1(9) - F_1(6) = \frac{11}{20-9} - \frac{77}{48}\sqrt{\frac{10-6}{(9-6)(13-6)}} = 1 - \frac{11}{24}\sqrt{\frac{7}{3}}$$

- $F_1(9) - F_1(6) = G_1(20) - G_1(10)$. $\therefore$ bidder 1 will exit the $\Lambda(6)$ and enter the waiting list.

Example

- $\Lambda^{+}(6) = \{1, 2, 4\}$
- Also, $v_1 = 20, v_2 = 14, v_4 = 12$
- $\Lambda(6) = \phi$; **waiting-list** $= \{1, 2, 3, 4\}$
- There corresponding available values are $\{10, 13, 9, 1\}$
- Buyer 2 has the largest valuation.
- Using Theorem 4, we can add bidder 2 to the bidding set at 6.
- $\Lambda(6) = \{2\}$; **waiting-list** $= \{1, 3, 4\}$
- Similarly, add player 1(Second highest bid) and $v_i \geq x$ and $|\Lambda(6)| = 1$.
- Now the waiting list has only two bidders $\{3, 4\}$.
- For Player 3, Cond 1 does not apply since $|\Lambda(6)| > 1$. Where as in condition 2, both the conditions , $h_i(x) \geq 0$ and $v_i > x$ are satisfied. Hence he will be added to the bidding set.
- For player 4, his value is 1. He cannot enter because $v_4 < 6$.
- Therefore $\Lambda^{+}(6)$ should have bid distributions of only 1,2,3.

Example

Verify our analysis with the bid distribution chart.

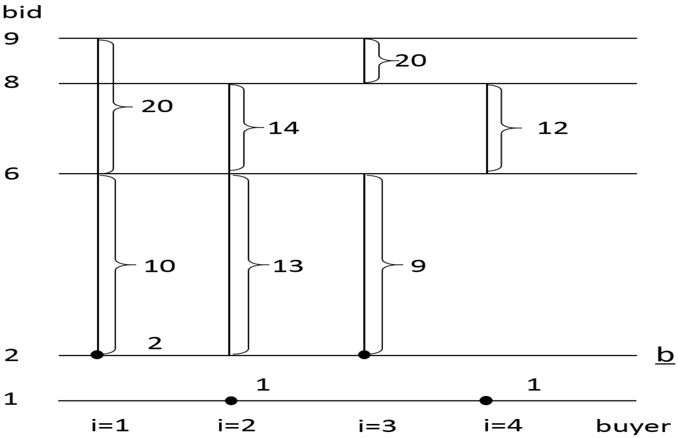

Figure: Bid Chart

Algorithm : Computation of $\underline{\mathrm{b}}(\overline{b})$

Algorithm 2 Algorithm to compute the minimum winning bid given the maximum bid

Input: The largest winning bid guess, $\overline{b}$
Output: The smallest winning bid $\underline{\mathrm{b}}(\overline{b})$

Initialize $b \leftarrow \overline{b}$; $\Lambda(b) \leftarrow \Phi$
Update $\Lambda(b)$ by repeatedly adding bidders to $\Lambda(b)$ using **Theorem 4**.
while $|\Lambda(b)| > 2$ and $b > 0$ **do**
    Predict the next change position $b^{'}$ using **Theorem 4** and **Theorem 5.**
    Set $b \leftarrow b^{'}$
    Update the bidding set $\Lambda(b^{'})$ by removing bidders using **Theorem 4** and **Theorem 5.**
    Update the bidding set $\Lambda(b^{'})$ by adding bidders using **Theorem 4** and **Theorem 5.**
end while
return $\underline{\mathrm{b}}(\overline{b})$=b

Monotonicity of Entry and Exit points

- Let us define

$$p_i^j = \ln G_i(v_i^j) - \ln G_i(v_i^{j-1}), \forall i = 1, \dots, n, j = 2, \dots, i_k.$$

- Let $\mathcal{E}(\bar{b}, \{p_i^j\})$ denote the set of bidding intervals given by Algorithm 2 with a guessed largest bid $\bar{b}$ and distribution G

Monotonicity of Entry and Exit points

Theorem 6

The extreme points of every bid interval in $\mathcal{E}(\bar{b}, \{p_i^j\})$ is monotone in $\bar{b}$.

Corollary 1

The position $\underline{b}(\bar{b})$ where Algorithm 2 stops is strictly monotone in $\bar{b}$.

- Analysis of this can be done by seeing an example.

Monotonicity of $\underline{b}(\bar{b})$

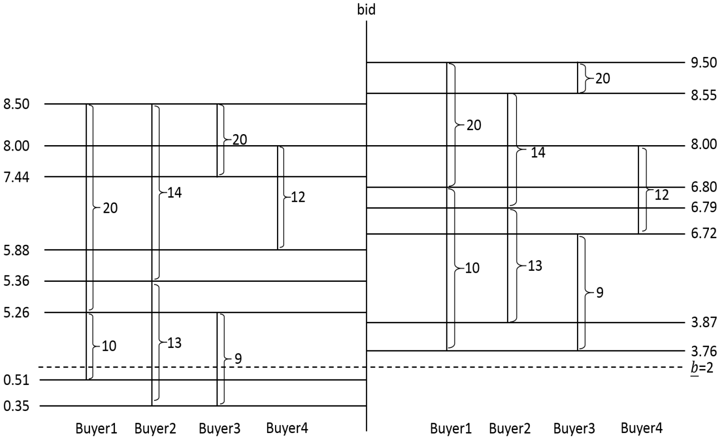

Figure: Monotonicity of $\underline{b}(\bar{b})$

Continuity of extremum points on changing $\bar{b}$

Theorem 7

The limit of each bid interval constructed by Algorithm 2 with the largest winning bid approaching to $\bar{b}^1$, is same as the bid interval constructed with the largest winning bid $\bar{b}^1$

Properties

- If $\underline{\mathrm{b}}(\bar{b})$ matches the smallest winning bid $\underline{\mathrm{b}}$, the bidding strategies given by Algorithm 2 is a BNE. If $\underline{b}(\bar{b}) \neq \underline{b}$ then the strategies formed do not form a BNE.
- A Bayesian Nash Equilibrium always exists when buyers have discrete value distributions.
- Buyers with identical bid distributions have identical bidding strategies in the BNE. Furthermore, if all buyers have identical value distributions, i.e. symmetric distributions, the BNE is also symmetric.
- There exists a unique BNE when buyers have discrete value distributions.

Quick Recap

- Algorithm 1 computes the BNE of the FPA by starting with a guess of $\bar{b}$ and then from there on move in a direction towards the actual $\bar{b}$ – Similar to a binary search. Note that this is possible only because $\underline{\mathrm{b}}(\bar{b})$ is monotone in $\bar{b}$.
- Our termination point was when $\underline{\mathrm{b}}(\bar{b}) = \underline{\mathrm{b}}$.
- Algorithm 2 gave us procedure to compute $\underline{\mathrm{b}}(\bar{b})$.

Predicting Change Points of Λ Effectively:

- Given a guess of the maximum bid $\bar{b}$, our algorithm computes the bidding strategies all the way down from $\bar{b}$ to $\underline{\mathrm{b}}(\bar{b})$.
- Such a computation would involve knowing the points at which the bidding set changes. A naive way of solving this is to decrease the bid gradually and check if any buyer would leaver or enter the bidding set.
- However, there is a more efficient way to compute this directly.

Predicting Change Points of Λ Effectively:

- Let the current change point be b.
- We want to predict the next change point of the bidding set. Let it be $b^{'}$
- Let the change occur due to player i.
- There are two possibilities that will lead to this change:
  - buyer i enters the bidding set with value v_i^j
  - buyer i exits the bidding set with value v_i^j

Computation of b' when player i enters Λ

- If player i enters the bidding set at $b^{'}$ then $b^{'}$ would have to satisfy the two conditions in **Theorem 4**
  i.e. $v_i^j \geq b^{'}$ and $h_i(b^{'}) \geq 0$.
- Since there is no other change point between b and $b^{'}$, the Λ wont change in the interval $(b, b^{'})$. Which implies that $h_i(x)$ can be computed in this interval and the largest bid $b^{'}$ that satisfies the above condition is chosen. [**Theorem 1**]
- Solving $h_i(x)$ could be hard sometimes, however this can be easily overcome by solving the following polynomial and discarding inappropriate solutions:

$$0 = \left[(v_i - x)\Pi_{j\in\Lambda}(v_j - x)\right] h_i(x) = \frac{v_i - x}{|\Lambda| - 1}\sum_{j\in\Lambda}\Pi_{k\in\Lambda\setminus j}(v_k - x) - \Pi_{j\in\Lambda}(v_j - x)$$

Computation of b' when player i exits Λ

- Player i with value v_i^j exists the bidding set only when the probability of v_i^j has been consumed completely [**Theorem 5**].
- Let α be the remaining probability of v_i^j at point b:

$$\alpha = F_i(b) = G_i(v_i^{j-1})$$

From Theorem 1 we have $\dfrac{d \ln F_i(x)}{dx} = h_i(x)$

We shall define $H_i(x) = \ln(v_i - x) - \dfrac{1}{|\Lambda(x)| - 1} \sum_{j \in \Lambda}(v_j - x)$

- The final equation to be solved will is

$$\ln F_i(b) - ln(F_i(b) - \alpha) = \ln F_i(b) - \ln G_i(v_i^{j-1}) = H_i(b) - H_i(b^{'})$$

Computation of b' when player i exits Λ

- Note that any bidder i could cause the change in the bidding set either by entering or exiting.
- Therefore one will have to compute $b^{'}$ for every player i. Let it be denoted by $b_i^{'}$. Then we choose the next changing point $b^{'}$ to be:

$$b^{'} = max_i\{b_i^{'}\}$$

- Note that in the above equation it is the maximum $b^{'}$ we are taking and not minimum because we are computing our strategies backwards. i.e., from maximum to minimum. Hence to be inclusive of all bidders, we need to take maximum and not minimum.

Updating the bidding set Λ

- We first remove the players who are going to leave the bidding set, and then add players in decreasing order of their unconsumed values. If any bidder j does not enter the bidding set then no other player will.

Time Complexity

- For Algorithm 2:
  - Each buyer can change the bidding set atmost twice at x , which implies $2\sum_i d_i = 2m$ times. And there are n such players. So there will be at most 2mn steps.
- For Algorithm 1:
  - If $L = max\{U_i supp(G_i)$. Suppose we use $UB - LB < \epsilon$ as the exit condition, then algorithm is of the order O($mn\log(L/\epsilon)$)

Thank You!