Bidding Games

Matvey Borodin, Kaylee Ji, Yifan Kang
Mentor: Chun Hong Lo

What are Bidding Games?

Win n Times in a Row
Win 2 times in a row
Approx. algorithm

December 7, 2021
What are Bidding Games?

Imagine a game of Tic-Tac-Toe: instead of alternating turns, players get to make a move if they out-bid the other player.
What are Bidding Games?

Imagine a game of Tic-Tac-Toe: instead of alternating turns, players get make a move if they out-bid the other player.
What are Bidding Games?

Imagine a game of Tic-Tac-Toe: instead of alternating turns, players get to make a move if they out-bid the other player.
What are Bidding Games?

Imagine a game of Tic-Tac-Toe: instead of alternating turns, players get make a move if they out-bid the other player.
What are Bidding Games?

Imagine a game of Tic-Tac-Toe: instead of alternating turns, players get to make a move if they out-bid the other player.
What are Bidding Games?

Imagine a game of Tic-Tac-Toe: instead of alternating turns, players get make a move if they out-bid the other player.
What are Bidding Games?

Definition (Bidding Games).

- two player zero sum games on a graph where each player has an objective node
- each turn, highest bidding player moves
- players bid simultaneously
- players know each other’s bidding history and budgets
All Pay Bidding Games

Both players pay their bid (as opposed to only the highest bidding paying)
All Pay Bidding Games

Both players pay their bid (as opposed to only the highest bidding paying)
All Pay Bidding Games

Both players pay their bid (as opposed to only the highest bidding paying)
Bidding Games

Matvey Borodin, Kaylee Ji, Yifan Kang
Mentor: Chun Hong Lo

What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm

All Pay Bidding Games

Both players pay their bid (as opposed to only the highest bidding paying)
All Pay Bidding Games

Both players pay their bid (as opposed to only the highest bidding paying)
All Pay Bidding Games

Both players pay their bid (as opposed to only the highest bidding paying)
Win $n$ Times in a Row

**Definition** (Win $n$ Times in a Row Game).
- all-pay bidding game with $\leq n$ turns
- player 1 wins if they out-bids player 2 $n$ times in a row
- player 2 wins if they out-bids player 1 any turn
- assumes money is infinitely divisible
- tie breaking: if both players bid the same value, we consider player 1’s bid higher

**Figure:** Visualizing WnR($n$) on a graph
Consider a win 3 times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.
Win $n$ Times in a Row

Consider a win 3 times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.

- Alice bids 2 and Bob bids 0.2
Win $n$ Times in a Row

Consider a win 3 times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.

- Alice bids 2 and Bob bids 0.2
- Alice bids 1.1 and Bob bids 0.6
What are Bidding Games?

Win \( n \) Times in a Row

Consider a win 3 times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.

- Alice bids 2 and Bob bids 0.2
- Alice bids 1.1 and Bob bids 0.6
- Alice bids 0.9 and Bob bids 1.2

Important notes:
- same game if Alice has budget 2 and Bob has budget 1 and each player halves their bids
- budget ratio - ratio of player 1’s budget to player 2’s budget
- we will set players 2’s budget as 1 in later games
Win $n$ Times in a Row

Consider a win $3$ times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.

- Alice bids 2 and Bob bids 0.2
- Alice bids 1.1 and Bob bids 0.6
- Alice bids 0.9 and Bob bids 1.2

Bob wins!
Consider a win 3 times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.

- Alice bids 2 and Bob bids 0.2
- Alice bids 1.1 and Bob bids 0.6
- Alice bids 0.9 and Bob bids 1.2

Bob wins!

Important notes:

- same game if Alice has budget 2 and Bob has budget 1 and each player halves their bids
Win n Times in a Row

Consider a win 3 times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.

- Alice bids 2 and Bob bids 0.2
- Alice bids 1.1 and Bob bids 0.6
- Alice bids 0.9 and Bob bids 1.2

Bob wins!

Important notes:

- same game if Alice has budget 2 and Bob has budget 1 and each player halves their bids
- budget ratio - ratio of player 1’s budget to player 2’s budget
What are Bidding Games?

Win $n$ Times in a Row

Consider a win 3 times in a row game where Alice, player 1, has a budget of 4 and Bob, player 2, has a budget of 2.

- Alice bids 2 and Bob bids 0.2
- Alice bids 1.1 and Bob bids 0.6
- Alice bids 0.9 and Bob bids 1.2

Bob wins!

Important notes:

- same game if Alice has budget 2 and Bob has budget 1 and each player halves their bids
- \textit{budget ratio} - ratio of player 1’s budget to player 2’s budget
- we will set players 2’s budget as 1 in later games
To analyze the game, we assume both players use randomized strategies (eg. a strategy for Player 1 on the their first turn is to bid 1 or 0.5, each with probability $\frac{1}{2}$).
To analyze the game, we assume both players use randomized strategies (eg. a strategy for Player 1 on their first turn is to bid 1 or 0.5, each with probability $\frac{1}{2}$).

Lower Value ($val\downarrow$): Player 1’s probability of winning in the worse case scenario (ie. when Player 2 always plays the best strategy to counteract Player 1’s strategy)
Bidding Games

Matvey Borodin, Kaylee Ji, Yifan Kang
Mentor: Chun Hong Lo

What are Bidding Games?

Win n Times in a Row

Win 2 times in a row

Approx. algorithm

Value

To analyze the game, we assume both players use randomized strategies (eg. a strategy for Player 1 on the their first turn is to bid 1 or 0.5, each with probability \( \frac{1}{2} \)).

Lower Value (\( val_\downarrow \)): Player 1’s probability of winning in the worse case scenario (ie. when Player 2 always plays the best strategy to counteract Player 1’s strategy)

Upper Value (\( val_\uparrow \)): Player 1’s maximum probability of winning when Player 2’s plays a strategy that maximizes their worse case scenario
What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm

To analyze the game, we assume both players use randomized strategies (e.g., a strategy for Player 1 on their first turn is to bid 1 or 0.5, each with probability $\frac{1}{2}$).

Lower Value ($val^\downarrow$): Player 1’s probability of winning in the worse case scenario (i.e., when Player 2 always plays the best strategy to counteract Player 1’s strategy)

Upper Value ($val^\uparrow$): Player 1’s maximum probability of winning when Player 2’s plays a strategy that maximizes their worse case scenario

When the Lower Value is equal to the Upper Value, we call this quantity Value.
Simple cases in WnR(2)

- $B_1 = 2$: Bid 1 on both turns guarantees winning, so the value of the game is 1.
Simple cases in WnR(2)

- \( B_1 = 2 \): Bid 1 on both turns guarantees winning, so the value of the game is 1.

- \( B_1 = 1 \): If player 1 wins the first round, player 2 will win the second bidding. Player 1 has no chance of winning two times in a row so the value of the game is 0.
The value of the game

Theorem

In the “win twice in a row” game, given initial budget ratio $B_1$, the value of the game is 1 for $B_1 \geq 2$, 0 for $B_1 \leq 1$ and $\frac{1}{n}$ for $B_1 \in [1 + \frac{1}{n}, 1 + \frac{1}{n-1})$ with $n \in \mathbb{Z}_{\geq 2}$.

Proof.

- Let $B_1 = 1 + \frac{1}{n} + \epsilon$ with $n \in \mathbb{Z}_{\geq 2}$ and $\epsilon \in [0, \frac{1}{n-1} - \frac{1}{n})$. 
The value of the game

**Theorem**

*In the “win twice in a row” game, given initial budget ratio $B_1$, the value of the game is 1 for $B_1 \geq 2$, 0 for $B_1 \leq 1$ and $\frac{1}{n}$ for $B_1 \in [1 + \frac{1}{n}, 1 + \frac{1}{n-1})$ with $n \in \mathbb{Z}_{\geq 2}$.***

**Proof.**

- Let $B_1 = 1 + \frac{1}{n} + \epsilon$ with $n \in \mathbb{Z}_{\geq 2}$ and $\epsilon \in [0, \frac{1}{n-1} - \frac{1}{n})$.
- Next, we want to show a strategy for player 1 that has at least $\frac{1}{n}$ chance of winning.
Player 1’s strategy in WnR(2)

- In the first bidding, choose $\frac{m}{n}$ which $1 \leq m \leq n$ uniformly at random.
Player 1’s strategy in WnR(2)

- In the first bidding, choose $\frac{m}{n}$ which $1 \leq m \leq n$ uniformly at random.
- By this we divided $[0, 1]$ into $n$ intervals, $[0, \frac{1}{n}], [\frac{1}{n}, \frac{2}{n}], \ldots, [\frac{n-1}{n}, 1]$. 
Player 1’s strategy in WnR(2)

- In the first bidding, choose $\frac{m}{n}$ which $1 \leq m \leq n$ uniformly at random.
- By this we divided $[0, 1]$ into $n$ intervals, $[0, \frac{1}{n}], [\frac{1}{n}, \frac{2}{n}], \ldots, [\frac{n-1}{n}, 1]$. 
- Any bid value that player 2 play must fall into some intervals $[\frac{k}{n}, \frac{k+1}{n}]$ above. Now, denote $B'_1, B'_2$ as player 1 and 2’s budget after the first bidding.
Player 1’s strategy in WnR(2)

- In the first bidding, choose \( \frac{m}{n} \) which \( 1 \leq m \leq n \) uniformly at random.
- By this we divided \([0, 1]\) into \(n\) intervals, 
  \([0, \frac{1}{n}], [\frac{1}{n}, \frac{2}{n}], \ldots, [\frac{n-1}{n}, 1]\).
- Any bid value that player 2 play must fall into some intervals 
  \([\frac{k}{n}, \frac{k+1}{n}]\) above. Now, denote \(B'_1, B'_2\) as player 1 and 2’s budget after the first bidding.
- If player 1 plays \(\frac{k+1}{n}\):
  \[B'_1 = B_1 - b_1 = \frac{n-k}{n} + \epsilon > \frac{n-k}{n} \geq 1 - b_2 = B'_2\]
  player 1 has more budget so player 1 always wins.
Player 1’s strategy in WnR(2)

- In the first bidding, choose $\frac{m}{n}$ which $1 \leq m \leq n$ uniformly at random.
- By this we divided $[0, 1]$ into $n$ intervals, $[0, \frac{1}{n}], [\frac{1}{n}, \frac{2}{n}], \ldots, [\frac{n-1}{n}, 1]$.
- Any bid value that player 2 play must fall into some intervals $[\frac{k}{n}, \frac{k+1}{n}]$ above. Now, denote $B_1', B_2'$ as player 1 and 2’s budget after the first bidding.
- If player 1 plays $\frac{k+1}{n}$:
  $$B_1' = B_1 - b_1 = \frac{n-k}{n} + \epsilon > \frac{n-k}{n} \geq 1 - b_2 = B_2'$$
  player 1 has more budget so player 1 always wins.
- Since player 1 would pick $\frac{k+1}{n}$ with probability $\frac{1}{n}$, the lower value is $\frac{1}{n}$. 

What are Bidding Games?
Win n Times in a Row
Win 2 times in a row
Approx. algorithm
Player 2’s strategy in WnR(2)

- We also find a player 2 strategy that guarantees player 1 cannot win with probability over $\frac{1}{n}$. 
What are Bidding Games?

Win n Times in a Row

Win 2 times in a row

Approx. algorithm

Player 2’s strategy in WnR(2)

- We also find a player 2 strategy that guarantees player 1 cannot win with probability over $\frac{1}{n}$.
- Notice that $\epsilon < \frac{1}{n-1} - \frac{1}{n}$. Then there exists an $\epsilon'$ such that $\epsilon' \in (\epsilon, \frac{1}{n-1} - \frac{1}{n})$. 
We also find a player 2 strategy that guarantees player 1 cannot win with probability over $\frac{1}{n}$.

Notice that $\epsilon < \frac{1}{n-1} - \frac{1}{n}$. Then there exists an $\epsilon'$ such that $\epsilon' \in (\epsilon, \frac{1}{n-1} - \frac{1}{n})$.

Consider the strategy of choosing $b_2$ from the set $\{k\left(\frac{1}{n} + \epsilon'\right) | 0 \leq k \leq n - 1\}$ uniformly at random.
We also find a player 2 strategy that guarantees player 1 cannot win with probability over $\frac{1}{n}$.

Notice that $\epsilon < \frac{1}{n-1} - \frac{1}{n}$. Then there exists an $\epsilon'$ such that $\epsilon' \in (\epsilon, \frac{1}{n-1} - \frac{1}{n})$.

Consider the strategy of choosing $b_2$ from the set \( \{k(\frac{1}{n} + \epsilon')|0 \leq k \leq n - 1\} \) uniformly at random.

If $b_1 < b_2$, player 1 loses immediately.
Player 2’s strategy in WnR(2)

- Else if \( b_1 > b_2 + \frac{1}{n} + \epsilon \). The budget ratio would be
  \[
  \frac{B_1 - b_1}{1 - b_2} < \frac{(1 + \frac{1}{n} + \epsilon) - (b_2 + \frac{1}{n} + \epsilon)}{1 - b_2} < 1
  \]

  so player 1 will lose the second bidding.
Player 2’s strategy in WnR(2)

- Else if $b_1 > b_2 + \frac{1}{n} + \epsilon$. The budget ratio would be
  \[
  \frac{B_1 - b_1}{1 - b_2} < \frac{(1 + \frac{1}{n} + \epsilon) - (b_2 + \frac{1}{n} + \epsilon)}{1 - b_2} < 1
  \]
  so player 1 will lose the second bidding.

- Hence, the only way for player 1 to win is play $b_1 \in [b_2, b_2 + \frac{1}{n} + \epsilon]$. 
Player 2’s strategy in WnR(2)

- Else if $b_1 > b_2 + \frac{1}{n} + \epsilon$. The budget ratio would be

$$\frac{B_1 - b_1}{1 - b_2} < \frac{(1 + \frac{1}{n} + \epsilon) - (b_2 + \frac{1}{n} + \epsilon)}{1 - b_2} < 1$$

so player 1 will lose the second bidding.

- Hence, the only way for player 1 to win is to play

$b_1 \in [b_2, b_2 + \frac{1}{n} + \epsilon]$. 

- However, $\frac{1}{n} + \epsilon < \frac{1}{n} + \epsilon'$, which means that for every $b_1$ there’s at most 1 value of $b_2$ that player 1 could win.
Player 2’s strategy in WnR(2)

- Else if \( b_1 > b_2 + \frac{1}{n} + \epsilon \). The budget ratio would be

\[
\frac{B_1 - b_1}{1 - b_2} < \frac{(1 + \frac{1}{n} + \epsilon) - (b_2 + \frac{1}{n} + \epsilon)}{1 - b_2} < 1
\]

so player 1 will lose the second bidding.

- Hence, the only way for player 1 to win is play

\( b_1 \in [b_2, b_2 + \frac{1}{n} + \epsilon] \).

- However, \( \frac{1}{n} + \epsilon < \frac{1}{n} + \epsilon' \), which means that for every \( b_1 \) there’s at most 1 value of \( b_2 \) that player 1 could win.

- This shows us that the upper value of the game is \( \frac{1}{n} \). Thus, the value is \( \frac{1}{n} \).
What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm
Motivation

- The game is much more complicated for higher $n$
Motivation

- The game is much more complicated for higher $n$
- Computer algorithm to approximate lower value
Motivation

- The game is much more complicated for higher $n$
- Computer algorithm to approximate lower value
- Simplify by assuming strategies consider finitely many bid values
Motivation

- The game is much more complicated for higher $n$
- Computer algorithm to approximate lower value
- Simplify by assuming strategies consider finitely many bid values
- Uses linear programming to solve for optimal strategy
Example with $\epsilon = 1$

First, an example of how the algorithm runs in WnR(3)

- Budgets $B_1 = 1.75$ and $B_2 = 1$
Example with $\epsilon = 1$

First, an example of how the algorithm runs in WnR(3)

- Budgets $B_1 = 1.75$ and $B_2 = 1$
- $b_1, b_2 \in \{0, 1\}$
Example with $\epsilon = 1$

First, an example of how the algorithm runs in WnR(3)

- Budgets $B_1 = 1.75$ and $B_2 = 1$
- $b_1, b_2 \in \{0, 1\}$
- Assume access to $f(x, y)$
Example with $\epsilon = 1$

First, an example of how the algorithm runs in WnR(3)

- Budgets $B_1 = 1.75$ and $B_2 = 1$
- $b_1, b_2 \in \{0, 1\}$
- Assume access to $f(x, y)$
- $f(x, y)$ is value in WnR(2) with starting budgets $x$ and $y$
Example with $\epsilon = 1$

First, an example of how the algorithm runs in WnR(3)

- Budgets $B_1 = 1.75$ and $B_2 = 1$
- $b_1, b_2 \in \{0, 1\}$
- Assume access to $f(x, y)$
- $f(x, y)$ is value in WnR(2) with starting budgets $x$ and $y$
Example with $\epsilon = 1$

First, an example of how the algorithm runs in WnR(3)

- Budgets $B_1 = 1.75$ and $B_2 = 1$
- $b_1, b_2 \in \{0, 1\}$
- Assume access to $f(x, y)$
- $f(x, y)$ is value in WnR(2) with starting budgets $x$ and $y$

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$f(1.75, 1) = 0.5$</td>
<td>$f(0.75, 1) = 0$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>$f(0.75, 0) = 1$</td>
</tr>
</tbody>
</table>

Table: Payoff matrix $A$
Goal is to optimize lower value

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$f(1.75, 1) = 0.5$</td>
<td>$f(0.75, 1) = 0$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>$f(0.75, 0) = 1$</td>
</tr>
</tbody>
</table>

Table: Payoff matrix $A$
Goal is to optimize lower value
- Player 1 strategy assuming player 2 plays optimally

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$f(1.75, 1) = 0.5$</td>
<td>$f(0.75, 1) = 0$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>$f(0.75, 0) = 1$</td>
</tr>
</tbody>
</table>

Table: Payoff matrix $A$
Goal is to optimize lower value

- Player 1 strategy assuming player 2 plays optimally
- Find best 1 by 2 vector \( \mathbf{p} \) such that \( \min(A \cdot \mathbf{p}) \) is maximized

Table: Payoff matrix \( A \)

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>( f(1.75, 1) = 0.5 )</td>
<td>( f(0.75, 1) = 0 )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>( f(0.75, 0) = 1 )</td>
</tr>
</tbody>
</table>
Goal is to optimize lower value
- Player 1 strategy assuming player 2 plays optimally
- Find best 1 by 2 vector $\mathbf{p}$ such that $\min(A \cdot \mathbf{p})$ is maximized
- $\max_{p_1, p_2} \min(0.5p_1 + 0p_2, 0p_1 + 1p_2)$

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$f(1.75, 1) = 0.5$</td>
<td>$f(0.75, 1) = 0$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>$f(0.75, 0) = 1$</td>
</tr>
</tbody>
</table>

Table: Payoff matrix $A$
What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm

Optimization

- Goal is to optimize lower value
  - Player 1 strategy assuming player 2 plays optimally
  - Find best 1 by 2 vector $p$ such that $\min(A \cdot p)$ is maximized
    - $\max_{p_1,p_2} \min(0.5p_1 + 0p_2, 0p_1 + 1p_2)$
    - $p_1 = \frac{2}{3}, p_2 = \frac{1}{3}$

Table: Payoff matrix $A$

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$f(1.75, 1) = 0.5$</td>
<td>$f(0.75, 1) = 0$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>$f(0.75, 0) = 1$</td>
</tr>
</tbody>
</table>
What are Bidding Games?

Win \( n \) Times in a Row

Win 2 times in a row

Approx. algorithm

**Optimization**

- **Goal is to optimize lower value**
  - Player 1 strategy assuming player 2 plays optimally
- **Find best 1 by 2 vector \( p \) such that \( \min(A \cdot p) \) is maximized**
  - \( \max_{p_1, p_2} \min(0.5p_1 + 0p_2, 0p_1 + 1p_2) \)
  - \( p_1 = \frac{2}{3}, \ p_2 = \frac{1}{3} \)
- Note we consider \( \min \), not weighted average for player 2 strategy

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>( f(1.75, 1) = 0.5 )</td>
<td>( f(0.75, 1) = 0 )</td>
</tr>
<tr>
<td>1</td>
<td>( f(0.75, 0) = 1 )</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table**: Payoff matrix \( A \)
Another example

\[ n = 3, \ B_1 = 2, \ \epsilon = 0.25 \]

\[ \max_p \min (A \cdot p) \]

\[ p = \begin{pmatrix} 0.368 \\ 0.158 \\ 0.158 \\ 0.0 \\ 0.316 \end{pmatrix} \]

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0.25</th>
<th>0.5</th>
<th>0.75</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
<td>0.33</td>
<td>0.2</td>
<td>0</td>
</tr>
<tr>
<td>0.25</td>
<td>0</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.25</td>
</tr>
<tr>
<td>0.5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>0.75</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**Table:** Payoff matrix \( A \)
What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm

---

**Algorithm** Approximate value of $W_{nR}(n)$

**function** $\text{VALUE}(n, \epsilon, B)$

$$b \leftarrow \{n \cdot \epsilon : 0 \leq n \leq \frac{1}{\epsilon}\}$$
Algorithm Approximate value of WnR(n)

function VALUE(n, $\epsilon$, $B$)
    $b \leftarrow \{ n \cdot \epsilon : 0 \leq n \leq \frac{1}{\epsilon} \}$
    for $b_1 \in b$, $b_2 \in b$ do
        $B' \leftarrow \frac{B-b_1}{1-b_2}$
        if $b_1 \geq b_2$ then
            payoff($b_1, b_2$) $\leftarrow$ VALUE($n-1, \epsilon, B'$)
**Algorithm** Approximate value of $W_{nR}(n)$

```plaintext
function VALUE(n, $\epsilon$, $B$)
    $b \leftarrow \{n \cdot \epsilon : 0 \leq n \leq \frac{1}{\epsilon}\}$
    for $b_1 \in b$, $b_2 \in b$ do
        $B' \leftarrow \frac{B-b_1}{1-b_2}$
        if $b_1 \geq b_2$ then
            payoff($b_1, b_2$) $\leftarrow$ VALUE($n - 1, \epsilon, B'$)
        else
            payoff($b_1, b_2$) $\leftarrow$ 0
    end for
    $p \leftarrow \max_{i} \sum_{j} payoff(j, i) \cdot p(j)$
    return $\min_{i} \sum_{j} payoff(j, i) \cdot p(j)$
```

What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm
Algorithm Approximate value of $\text{WnR}(n)$

function $\text{VALUE}(n, \epsilon, B)$

    $b \leftarrow \{ n \cdot \epsilon : 0 \leq n \leq \frac{1}{\epsilon} \}$

    for $b_1 \in b, b_2 \in b$ do
        $B' \leftarrow \frac{B - b_1}{1 - b_2}$
        if $b_1 \geq b_2$ then
            $\text{payoff}(b_1, b_2) \leftarrow \text{VALUE}(n - 1, \epsilon, B')$
        else
            $\text{payoff}(b_1, b_2) \leftarrow 0$
        end if
    end for

    $p \leftarrow \max_p \min_i \sum_j \text{payoff}(j, i) \cdot p(j)$

return $\min_i \sum_j \text{payoff}(j, i) \cdot p(j)$
Algorithm Approximate value of \( W_nR(n) \)

\[
\text{function VALUE}(n, \epsilon, B) \\
\quad b \leftarrow \{n \cdot \epsilon : 0 \leq n \leq \frac{1}{\epsilon}\} \\
\quad \text{for } b_1 \in b, \ b_2 \in b \text{ do} \\
\quad \quad B' \leftarrow \frac{B - b_1}{1 - b_2} \\
\quad \quad \text{if } b_1 \geq b_2 \text{ then} \\
\quad \quad \quad \text{payoff}(b_1, b_2) \leftarrow \text{VALUE}(n - 1, \epsilon, B') \\
\quad \quad \text{else} \\
\quad \quad \quad \text{payoff}(b_1, b_2) \leftarrow 0 \\
\quad \quad \text{end if} \\
\quad \text{end for} \\
\quad p \leftarrow \max_p \min_i \sum_j \text{payoff}(j, i) \cdot p(j) \\
\quad \text{return } \min_i \sum_j \text{payoff}(j, i) \cdot p(j) \\
\text{end function}
\]
What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm
What are Bidding Games?

Win n Times in a Row

Win 2 times in a row

Approx. algorithm
What are Bidding Games?

Win $n$ Times in a Row

Win 2 times in a row

Approx. algorithm