# Auction Bot Competition: Report

Venkatakrishnan Asuri

EP23B031

## 1 Introduction

I have chalked out the strategies I used for each of the three variants of the auction bot. Though I would also like to add, that I strongly can say that there (most probably) is **no optimal strategy** but only a robust one, as each strategy can be defeated by an optimised, better one. We shall denote the private value assigned to each player as $v$ instead of $x_i$ for LaTeX convenience purposes.

## 2 Variant 1: Simple First Price Auction

### 2.1 Strategy

This type of an auction is called a private-value first price sealed bid auction, and from a standard game theoretic result, we quote a result below. The strategy is pretty straightforward. We bid

$$\frac{n-1}{n} \cdot v$$

in each round, where $v$ is the private value.

### 2.2 Mathematics

Let $n$ be the number of bidders, and $v \sim U[0, 100]$ be the private value for bidder $i$. The optimal bidding function $b(v)$ is:

$$b(v) = \frac{n-1}{n} \cdot v$$

This strategy is derived from the Bayesian Nash Equilibrium for first-price auctions with independent private values. Result can be found at here. .

### 2.3 Justification

In a first-price sealed-bid auction with $n$ risk-neutral bidders and values uniformly distributed on $[0, 100]$, the symmetric Bayesian Nash Equilibrium bidding strategy is to bid a fraction $\frac{n-1}{n}$ of one's value.

## 2.4 Implementation

**Algorithm 1** First Price Sealed Bid Auction

**Input:** current_value, num_bidders, capital
bid_fraction $\leftarrow (num\_bidders - 1)/num\_bidders$
bid $\leftarrow bid\_fraction \cdot current\_value$
bid $\leftarrow \max(0, \min(capital, bid))$
**return** bid

# 3 Variant 2: Modified First Price Auction

## 3.1 Strategy

The strategy for the second variant varies starkly from that of the first variant in the fact that it uses an adaptive approach compared to the first variant, which had a fixed strategy. The bidding behaviour of the auction bot is adjusted based on the data of the previous rounds given and current behaviour of the auction.

## 3.2 Mathematics

Let $\alpha_{base}$ be the base aggressiveness factor, which determines how aggressive the bot bids, $\beta$ be the volatility sensitivity, which is responsible for the change in the extent of the change in bidding behaviour depending on the bids and $\sigma_t$ be the current market volatility. The adaptive factor $\alpha$ is calculated as:

$$\alpha = \alpha_{base} \cdot \left(1 + \beta \cdot \frac{\sigma_t - \sigma_{t-1}}{\max(\sigma_{t-1}, 1)}\right)$$

We calculate this, instead of maintaining a constant $\alpha$ so that we make sure that we adapt to the aggressive or conservative nature of the auction.

The bidding function $b(v)$ is then defined as:

$$b(v) = v - \theta \cdot |v - \bar{b}| + \epsilon$$

where $\theta$ is a factor based on the current value and market conditions (shown below), $\bar{b}$ is the average of previous winning bids, and $\epsilon$ is a small random noise term.

## 3.3 Components of the Strategy

### 3.3.1 Estimating Volatility

Market volatility $\sigma_t$ is estimated using the standard deviation of recent winning bids:

$$\sigma_t = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(b_i - \bar{b})^2}$$

where $N$ is the number of recent bids considered, $b_i$ are the winning bids, and $\bar{b}$ is their mean.

### 3.3.2 Adaptive Bidding Factor

The $\theta$ parameter is adjusted based on the current value $v$ relative to the estimated maximum value:

$$\theta = \begin{cases} 0.95 & \text{if } v < 25 \\ 0.9 & \text{if } 25 \leq v < 50 \\ 0.85 & \text{if } 50 \leq v < 75 \\ 0.8 & \text{if } v \geq 75 \end{cases}$$

Here, it can be added that I found these values of $\theta$ to be optimal after running a lot of simulations/rounds, but I was unable to proceed in making a more adaptive $\theta$, rather I'd put it as I didn't have much time.

### 3.3.3 Gaussian Noise

A small Gaussian noise $\epsilon$ is added to introduce unpredictability:

$$\epsilon \sim \mathcal{N}(0, \sigma_{noise})$$

where $\sigma_{noise}$ is a small constant, a noise factor (e.g., 0.1). This will make sure my bids aren't predictable by other bids, even though the bots get to know only the previous highest and second highest bids.

## 3.4 Implementation

**Algorithm 2** Modified First Price Auction

**Input:** current_value, previous_winners, capital, num_bidders
Calculate $\sigma_t$ from previous_winners
Update $\alpha$ using $\sigma_t$ and $\sigma_{t-1}$
Determine $\theta$ based on current_value
Calculate average_bid from previous_winners
bid $\leftarrow current\_value - \theta \cdot |current\_value - average\_bid|$
Add Gaussian noise to bid
bid $\leftarrow \max(0, \min(bid, current\_value, capital))$
**return** bid

# 4 Variant 3: Dangerous First Price Auction

## 4.1 Strategy

The strategy for this variant is a more sophisticated adaptive approach which estimated the maximum value of $X_max$ and adjusts the bidding based on various factors which include a risk factor, an aggression factor and also adjusting the bid based on the remaining capital so we don't run out of money.

## 4.2 Mathematics

Let $X_{max}$ be the estimated maximum value, $n$ be the number of bidders, and $v$ be the current value. The strategy estimates $X_{max}$ and uses it to determine the bidding function $b(v)$.

### 4.2.1 Maximum Value Estimation

$$X_{max} = \max\left(\frac{100n}{n+1}, \max(v_1, \ldots, v_k)\right) \cdot f_{aggr}$$

where $\frac{100n}{n+1}$ is the theoretical maximum for uniformly distributed values, $\max(v_1, \ldots, v_k)$ is the observed maximum from recent rounds, and $f_{aggr}$ is an aggressiveness factor.

### 4.2.2 Bidding Function

The bidding function is defined piece-wise based on the relationship between $v$ and $X_{max}$:

$$b(v) = \begin{cases} 0.98v & \text{if } v > X_{max} + \sigma \\ 0.95v & \text{if } X_{max} \leq v \leq X_{max} + \sigma \\ 0.90v & \text{if } X_{max} - \sigma \leq v < X_{max} \\ 0.80v & \text{if } v < X_{max} - \sigma \end{cases}$$

where $\sigma$ is the estimated standard deviation of values.

## 4.3 Key Components

### 4.3.1 Risk Management

The final bid is adjusted based on remaining capital:

$$b_{final}(v) = \begin{cases} \min(0.8b(v), 0.4c) & \text{if } c < 200 \\ \min(1.05b(v), 0.3c) & \text{if } c > 1000 \text{ and } b(v) > 0.9X_{max} \\ \min(b(v), 0.25c) & \text{otherwise} \end{cases}$$

where $c$ is the current capital.

### 4.3.2 Adaptivity

The aggressiveness factor $f_{aggr}$ is updated based on win rate and second-place rate:

$$f_{aggr} = \begin{cases} \min(1.1, f_{aggr} \cdot 1.05) & \text{if win\_rate} < 0.2 \text{ and second\_rate} < 0.1 \\ \max(0.9, f_{aggr} \cdot 0.95) & \text{if second\_rate} > 0.15 \\ \max(0.95, f_{aggr} \cdot 0.98) & \text{if win\_rate} > 0.3 \end{cases}$$

## 4.4 Implementation

**Algorithm 3** Dangerous First Price Auction Strategy

**Input:** current_value, previous_winners, previous_second_highest_bids, capital, num_bidders
Update value_history with current_value
Estimate $X_{max}$ using value_history and num_bidders
Calculate $\sigma$ based on num_bidders
Determine bidding function $b(v)$ based on current_value, $X_{max}$, and $\sigma$
Apply risk management to get $b_{final}(v)$
Update win_rate and second_rate
Adjust $f_{aggr}$ based on win_rate and second_rate
**return** $b_{final}(v)$

# 5 Comparison and Analysis

## 5.1 Strategy Comparison

- Variant 1 (Simple):
  - Pros: Theoretically optimal for uniform distributions, simple to implement
  - Cons: Does not adapt to changing market conditions or opponent behavior
- Variant 2 (Adaptive):
  - Pros: Adapts to market volatility, considers recent bid history
  - Cons: May overreact to short-term fluctuations, noise factor could lead to suboptimal bids
- Variant 3 (Sophisticated Adaptive):
  - Pros: Comprehensive strategy considering multiple factors, risk management, adaptive learning

  - Cons: Complex implementation, may require fine-tuning of multiple parameters

### 5.2 Theoretical Performance

- Variant 1 performs optimally in a static environment with uniform value distributions.
- Variant 2 may outperform in volatile markets or when the strategies of opponents vary.
- Variant 3 has the potential for best overall performance due to its adaptability and risk management, but its success depends on accurate parameter tuning. This variant in general is pretty volatile, from my observations.

## 6 Conclusion

Was a fun experience doing this, I used a combination of my brains and Claude Sonnet to come up with the ideas and the code, with the former contributing mostly to the strategy and the latter a lot to the code, which I admit, I was lazy to code up when I had a tool. My only regret is the shortage of time and the fact that I still am not satisfied with the strategy, but on the other hand there is always room for optimisation for these variants of auction.