CS5330 Randomized Algorithms Final Project

Extension on the paper "Competitive Auctions"

Alexis de Colnet - A0179633U
Xu Chen - A0144702N

Introduction

The initial paper ended up presenting two competitive randomized auction: DSOT (Dual-Price Sampling Optimal Threshold) auction and SCS (Sampling Cost Sharing) auction. These two auctions relies on the same pattern:

1. Partition bids from $\mathbf{b}$ uniformly at random into two sets: for each each bid, with probability $1/2$ put the bid in $\mathbf{b'}$ and otherwise in $\mathbf{b''}$.
2. Compute $\mathbf{p'}$ and $\mathbf{p''}$, the optimal fixed prices for $\mathbf{b'}$ and $\mathbf{b''}$ respectively.
3. Do something with $\mathbf{p''}$ on $\mathbf{b'}$ and with $\mathbf{p'}$ on $\mathbf{b''}$

Both auctions differ at the third point only.

From our little experience working with randomized algorithms we know that when some computation has to be performed from an input vector, partitioning the input vector into two sets is one classic way to go. For instance the *quicksort* algorithm does such partitioning recursively. In such methods it is legitimate to ask "why are we limiting ourselves to a 2-sets partitioning ?". After all *quicksort* can be implemented with multiple pivots and it actually works better in practice. So maybe the randomized auctions would benefit from a partitioning in three, four or many more sets. This paper extension try to answer this question. How do the algorithms performs with partitioning in more than 2 sets ? We will mostly focused on DSOT. The DSOT auction presented in the original paper will be renamed DSOT^2 and its alternate versions using K-sets partitioning will be named DSOT^K.

DSOT^K Auction

1. Partition bids $\mathbf{b}$ uniformly at random into K sets: for each bid, with probability $1/K$ put the bid in $\mathbf{b}^i$, $i \in [0, K-1]$.
2. Let $p^i = opt_r(\mathbf{b}^i)$, the optimal fixed price thresholds for $\mathbf{b}^i$
3. Use p^i as a threshold for all bids in $\mathbf{b}^{(i+1) mod\ K}$ so that a bidder j wins if $\mathbf{b}^i_j \leq p^{i-1}$.

Some immediate questions come to mind:

- Is the DSOT^K auction relying still competitive for $K > 2$?
- Are there some bid vectors on which DSOT^K is better than DSOT^2 ?
- Are there some bid vectors on which DSOT^K is worse than the DSOT^2 ?
- Which of DSOT^K and DSOT^2 is better on average ? And on the worst case scenario ?

Our report is organized in three parts. In the first part we try to analyze the DSOT^K auctions theoretically, proving some results regarding the truthfulness and competitiveness of these auctions. Next we will try to develop some intuition about this auction, specifically we will explain that DSOT^K systematically outperforms DSOT^2 and some specific bid vectors and that the reverse situation is also possible. Finally since the theory and the qualitative approach are not enough to answer our questions, in the third part we design some experiments to compare these DSOT^K auctions for different values of K.

Analysis of DSOT^K

We first justify that the DSOT^K auctions are truthful. This is straightforward using the bid-independence property. DSOT^K is truthful for the same reason that DSOT is.

Theorem 1. For any $K > 2$, DSOT^K is truthful.

Proof. In any DSOT^K auction, a bidder in $\mathbf{b}^i$ is assigned a price calculated from the bids in $\mathbf{b}^{i-1 \mod K}$. $K > 2$ implies $(i - 1 \mod K) \neq i$ so the price offer of the bidder is independent of its own bid. This holds for all bidder so the auction is bid-independent. Therefore it is truthful. □

Notations.

- For any value ν, $F_v^k = vn_v^{(k)}$ is the profit from using v as a threshold for $\mathbf{b}^k$.
- For a set of bids S, $\#^{(k)}(S)$ denotes the number of these bids in partition k.
- R is the profit of the auction DSOT$_r^k$. $R = \sum_{k=0}^{K-1} F_{p^{(k)}}^{k+1 \mod K}$
- E_α is the event $R \geq (1-\alpha)\left(\sum_{k=0}^{K-1} F_{p^{(k)}}\right)$
- $r = \lceil \frac{m}{K}(1-\delta) \rceil$.
- Suppose that $\text{opt}_m(\mathbf{b}) = v_k$, H is the event: in all $\mathbf{b}^k$ there are at least r bids that are at least v_k.

Definition. We say that the set of the j highest bids $B_j = \{v_1, \cdots, v_j\}$ is c-good for the partition k (with c in $]0,1[$) if there is:

$$\lceil cj \rceil \leq \#^{(k)}(B_j) \leq j - \lfloor cj \rfloor$$

Exactly like in the initial paper, we prove that B_j is likely to be c-good for a partition using the Chernoff Bound.

Lemma 1. Consider the DSOT^K auction. With the same definition of B_j as before, the following result holds:

$$\Pr\left[B_j \text{ is } c\text{-bad for partition } 1\right] \leq K \exp\left(\frac{(1-Kc)^2 j}{3K}\right)$$

Proof. Call X_i^k the random variable such that:

$$\begin{aligned} X_i^k &= 1 \text{ if } v_i \text{ ends up in partition } k \\ &= 0 \text{ otherwise} \end{aligned}$$

There is $\mathrm{E}\left[X_i^k\right] = 1/K$.

$$\Pr\left[\#^{(1)}(B_j) < cj\right] = \Pr\left[\sum_{i=1}^{j} X_i^k < (1-\delta)\frac{j}{K}\right]$$

With $\delta = 1 - Kc$. Since the assignments of each bid to a partition are i.i.d, assuming $c < 1/K$ we can derive from the Chernoff bound:

$$\Pr\left[\#^{(1)}(B_j) < cj\right] \leq \exp -\frac{(1-Kc)^2 j}{3K}$$

Now for the other bound, when $\#^{(k)}(B_j) > j - \lfloor cj \rfloor$

$$\Pr\left[\#^{(1)}(B_j) > j - cj\right] = \Pr\left[\sum_{i=2}^{K} \#^{(i)}(B_j) \leq cj\right] \leq \sum_{i=2}^{K} \Pr\left[\#^{(i)}(B_j) \leq cj\right] \leq (K-1)\exp -\frac{(1-Kc)^2 j}{3K}$$

Therefore by union bound it comes:

$$\Pr\left[B_j \text{ is } c\text{-bad for partition } 1\right] \leq K \exp\left(-\frac{(1-Kc)^2 j}{3K}\right)$$

With $\delta = (K-1) - Kc$. □

From this lemma we deduce the following result:

Lemma 2.

$$\Pr\left[B_j \text{ is } c\text{-bad for some partition for some } j > t\right] \leq K^2 \frac{e^{-\frac{(1-Kc)^2 t}{3K}}}{1 - e^{-\frac{(1-Kc)^2}{3K}}} = o_t(1)$$

Proof. We first focus on one of the partition set:

$$\begin{aligned}
\Pr\left[B_j \text{ is } c\text{-bad for partition 1 for some } j > t\right] &\leq K \sum_{j \geq t} \exp\left(\frac{(1-Kc)^2 j}{3K}\right) \\
&= K \sum_{j \geq 0} \exp\left(\frac{-(1-Kc)^2 j}{3K}\right) - K \sum_{j=0}^{t-1} \exp\left(\frac{(1-Kc)^2 j}{3K}\right) \\
&= \frac{K e^{-\frac{(1-Kc)^2 t}{3K}}}{1 - e^{-\frac{(1-Kc)^2}{3K}}}
\end{aligned}$$

And by union bound over the partition sets:

$$\Pr\left[B_j \text{ is } c\text{-bad for some partition for some } j > t\right] \leq K^2 \frac{e^{-\frac{(1-Kc)^2 t}{3K}}}{1 - e^{-\frac{(1-Kc)^2}{3K}}}$$

□

An immediate corollary is that:

$$\Pr\left[B_j \text{ is } c\text{-good for all some partition for all} j > t\right] \geq 1 - o_t(1)$$

From here we wanted to use the same proof scheme and see how it scales with DSOT^k. The first thing to notice is that, we can only follow the proof scheme proving that DSOT^k is competitive against $\mathcal{F}^{(K)}$. This is a problem: we are increasing the number of possible prices not to get rid of some worst case scenarios but because we try to follow a proof.

Theorem 2. There is a constant β such that DSOT^K is β competitive against $\mathcal{F}^{(K)}$.

Proof. We denote v_i the value of the i-th largest bid of $\mathbf{b}$. Suppose $\text{opt}_K(\mathbf{b}) = v_l$ and therefore $\mathcal{F}^{(K)} = lv_l$. The proof restricts to a certain distribution of the K highest bids. Proving that the profit when this distribution occurs is always greater than a constant ratio of $\mathcal{F}^{(K)}$ will prove the competitiveness of DSOT^K against $\mathcal{F}^{(K)}$, but will give a poor and not representative lower bound.

The distribution cases we restrict to is when $v_1 \in \mathbf{b}^1$, $v_2 \in \mathbf{b}^2$, $\cdots$,$v_K \in \mathbf{b}^K$ and $v_l \in \mathbf{b}^K$ (could be v_K but by definition can not be v_k for $k < K$). Such distribution represented by the event G occurs with probability $1/K^k$. Let H be the event that for all $j \geq K$ and for all $k \leq K$, B_j is $1/x$-good and event G holds (this makes a lot of conditions).

Choose y such that $y > \Pr\left[\overline{G}\right]$. Thanks to the following lemma we know there exist a x greater than K such that:

$$\Pr\left[\text{for all } k \leq K \text{ , } B_j \text{ is } 1/x\text{-good for all } j > x\right] \geq y$$

If G holds, all $j \in [K, x]$ are $1/x$-good, since there is $v_k \in \mathbf{b}^k$ for all $k \leq K$. Thus event H occurs with probability at least $y - \Pr\left[\overline{G}\right] = z > 0$.

Define for any k, $n_v^{(k)}$ the number of bids in $\mathbf{b}^k$ greater than v. Let $p_K = \text{opt}(\mathbf{b}^K)$. Since v_l is also in $\mathbf{b}^K$ by definition there is:

$$p_K n_{p_K}^{(K)} \geq v_l n_{v_l}^{(K)}$$

Conditioned on event H it comes inequality

$$p_K n_{p_K}^{(1)} \geq \frac{v_l n_{v_l}^{(K)}}{n_{p_K}^{(K)}} n_{p_K}^{(1)}$$

Thanks to H being satisfied, there is

$$n_{p_K}^{(1)} \geq \frac{n_{p_K}}{x} \text{ , } n_{p_K}^{(2)} \geq \frac{n_{p_K}}{x} \text{ , } \cdots \text{ , } n_{p_K}^{(K-1)} \geq \frac{n_{p_K}}{x}$$

and since

$$n_{p_K}^{(1)} + \cdots + n_{p_K}^{(K-1)} + n_{p_K}^{(K)} = n_{p_K}$$

it comes:

$$n_{p_K}^{(K)} \leq n_{p_K} \left(1 - \frac{K-1}{x} \right)$$

using this inequality and $n_{p_K}^{(1)} \geq \frac{n_{p_K}}{x}$ in (*) yields to:

$$p_K n_{p_K}^{(1)} \geq \frac{v_l n_{v_l}^{(K)}}{x - K + 1}$$

Finally using that

$$n_{v_l}^{(K)} = n_{v_l} - \left(n_{v_l}^{(1)} + \cdots + n_{v_l}^{(K-1)} \right) \geq n_{v_l} \left(1 - (k-1)(1 - 1/x) \right)$$

it comes:

$$p_K n_{p_K}^{(1)} \geq v_l n_{v_l} \frac{1 - (k-1)(1-1/x)}{x-K+1} = v_l l \frac{1 - (k-1)(1-1/x)}{x-K+1} = \mathcal{F}^{(K)}(\mathbf{b}) \frac{1 - (k-1)(1-1/x)}{x-K+1}$$

This implies that the expected profit of DSOT^K is greater than a ratio of $\mathcal{F}^{(K)}$ on any bids vector. In other words DSOT^K is competitive against $\mathcal{F}^{(K)}$. □

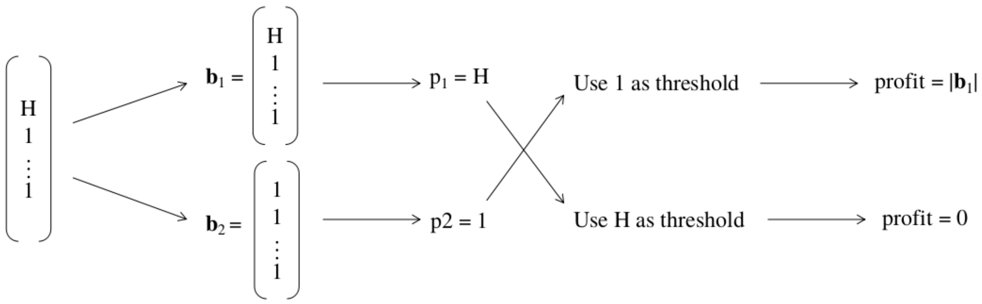
So even if there is no certainty that DSOT^K is competitive against $\mathcal{F}^{(2)}$, we know that we will always do better than a ratio of $\mathcal{F}^{(K)}$. We will keep aside the bound found in the previous proof as it is quite poor and derived from an analysis restricted to a particular distribution of bids and therefore is not representative of the auction's performances. The study of the DSOT^K auctions' average performances seems quite difficult and instead we will focus on giving intuitions on how well they perform. We will also conduct experiments to compare the auctions for different value of K.

Comparing DSOT^K, DSOT^2 and DSOT^n

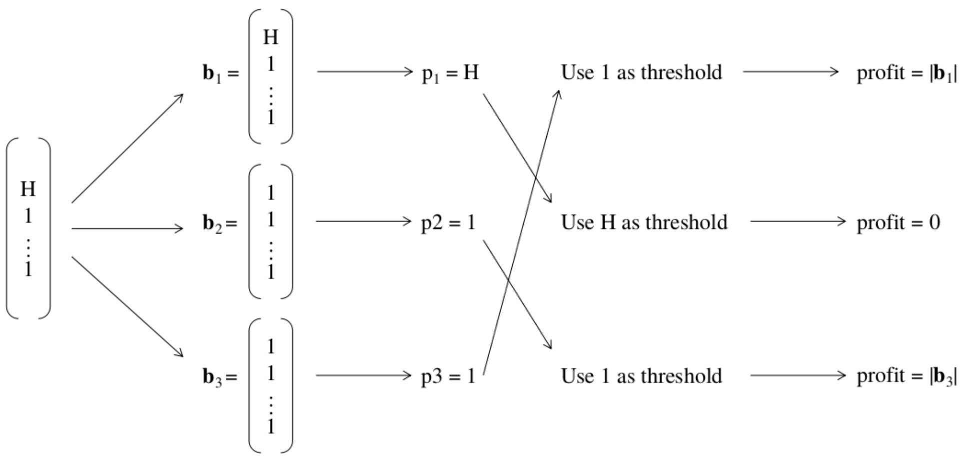
The meaning behind the comparison of two auctions is unclear. Do we compare them on random inputs or on their worst case inputs ? Do we try to prove that one auction systematically do better than the other ? ...

Can DSOT^K do better than DSOT^2 ?

Are there inputs on which DSOT^K outperforms DSOT^2 with high probability ? It turns out the answer is yes and that such inputs are not complicated to find. We just have to look at pathological inputs such as the "crazy rich guy" input: every bidders have a bid negligible against one specific bidder. Call H the bid of this specific bidder and take 1 for all the other bids. On this input DSOT^2 behaves as shown in the following figure. Only one of the two sets can generate a profit which is its size (since all bidders pay 1). In expectation the profit is $n/2$ where n is the total number of bidders.



Now when we look at what is happening in DSOT^3, we see that two partitions generate profit. In expectation the size of any partition is $n/3$ so the expected profit is likely to be around $2n/3$ which is an improvement compared to DSOT^2.

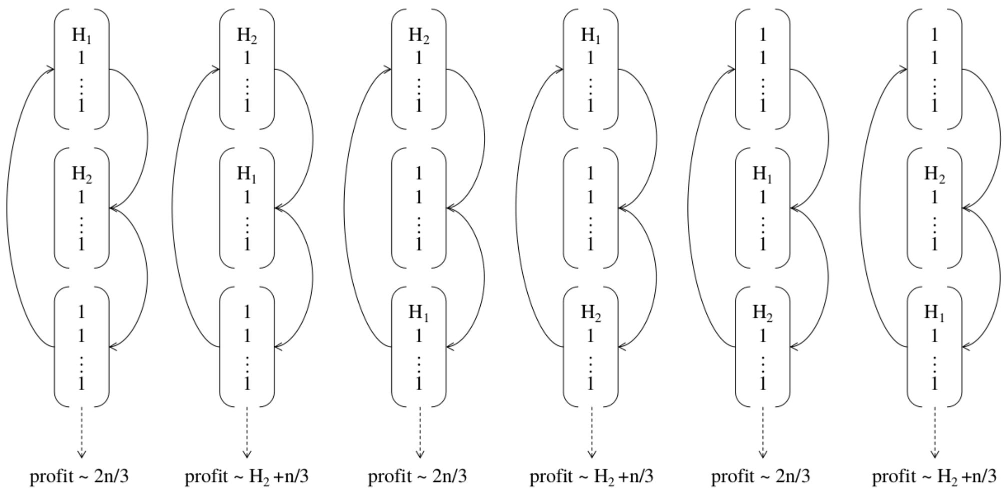


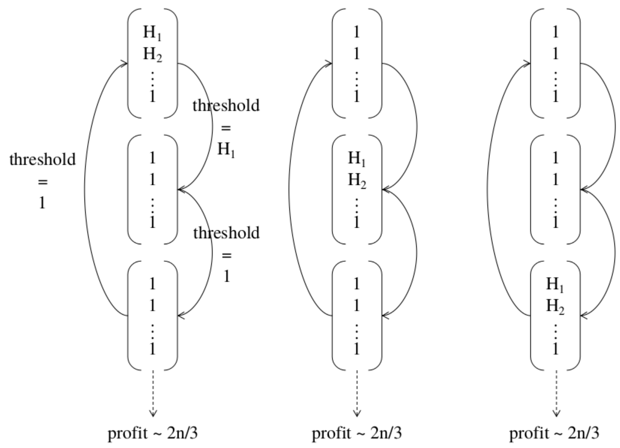
And if we can keep going with K even larger, the expected profit on this input keep getting better until we reach DSOT^n for which it is maximum.

Can DSOT^K do worse than DSOT^2 ?

We have seen that there exist bid vectors in which DSOT^K does better than DSOT^2. A legitimate question is to search for the reverse situation: are they bid vectors easy to come up with in which DSOT^K is worsed than DSOT^2 with high probability ?
We believe such bid vectors exist. Let us first compare DSOT^2 with DSOT^3 as we did in the previous section. Consider the input bid vector , with two bids H_1 and H_2 such that $H_1 > H_2 \gg 1$ and all the others at 1 (negligible). It is important that H_1 is strictly greater than H_2.

- In DSOT^2, H_1 and H_2 end up in the same set with probability $1/2$. In such configuration the profit will be the size of one of the two sets, so in expectation $n/2$. The other configuration is: H_1 and H_2 are separated, this also occurs with probability $1/2$ but the profit is H_2 now.
- In DSOT^3, H_1 and H_2 end up in the same set with probability $1/3$. In such configuration the profit will be the size of two sets, so in expectation $2n/3$. The other configuration is that H_1 and H_2 are separated. This occurs with probability $1/2$ but there are 6 different situations and they are not equivalent. Specifically there will be a profit of more than H_2 in only three situations: when H_2 is in the set i and that H_1 is in $i+1$ (modulo 3). Since each situation occurs with probability $1/9$, the profit is around $2n/3$ with probability $1/3$ and around $H_2 + n/3$ with probability $1/3$.





Therefore the expected profit should be around $\frac{H_2}{2} + \frac{n}{4}$ for DSOT^2 and around $\frac{H_2}{3} + \frac{5n}{9}$ for DSOT^3. With H_2 large enough DSOT^3 is worse than DSOT^2 on this bid vector.

Note: As you can see we used the expected sizes of the partitions in the analysis. Rigor forbids this usage and this is why we only present our results as intuition and not as an indisputable truth. However consider the following modification on the auctions' mechanisms: instead of attributing a bid to one of the K sets uniformly at random, randomly partition the initial bid vector into K sets of equal size. On these modified auctions (most likely worst than the originals), our analysis becomes correct.

For this specific bid input vector, we believe the profit given by DSOT^K become even poorer as K grows. This is because the probability of H_2 ending in the set direct predecessor of the one containing H_1 decreases as the number of partitions increases and therefore a profit of more than H_2 is rarer when K increases.

Experiments

It is difficult to get an intuition of how two auctions compare on an arbitrary vector ,and therefore we decided to conduct some experiments to help us get an rough idea of how the DSOT^K auctions compare on average. The experiments are just about computing the mean of the profits (estimator of the expected profit) for several DSOT^K auctions on many different bid vectors.

Context and simplifications

Since the whole idea behind competitiveness is to give a performance bound for an auction on all inputs, we have thought about computing the mean of the profit of the DSOT^K auctions for all possible bid vectors made of bids between 1 and some max value H. But the number of inputs would then have been something like H^n and running all the auctions on all inputs (many times since we need a mean) would have been too long. So we decided to restrict to a set of randomly generated bid vectors. However we made sure to uniformly select the bid vectors, specifically we did not construct them by assigning random bids values as it would have generated vectors following a normal distribution. Furthermore while we should have conducted experiments for different values of n, we decided to give it a fixed value. We made sure that the maximum bid value would be far greater than n and that the authorized values would mostly be not too big compared to n.

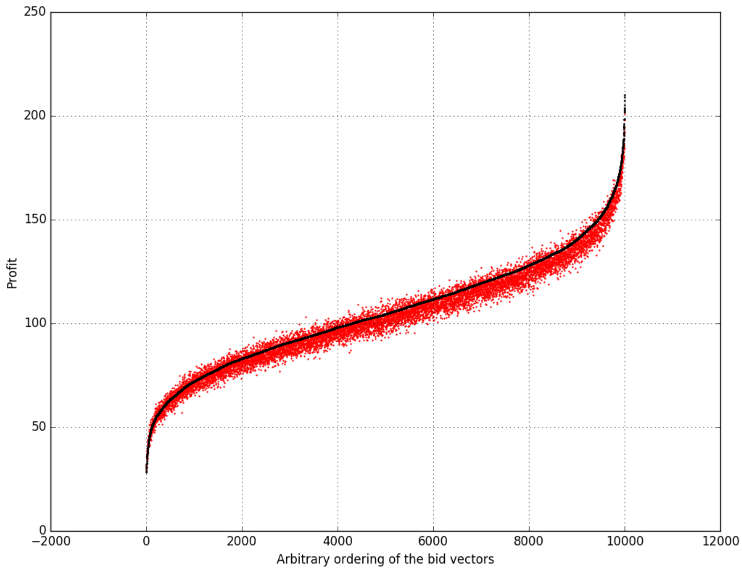
Settings

The settings of the experiments are:

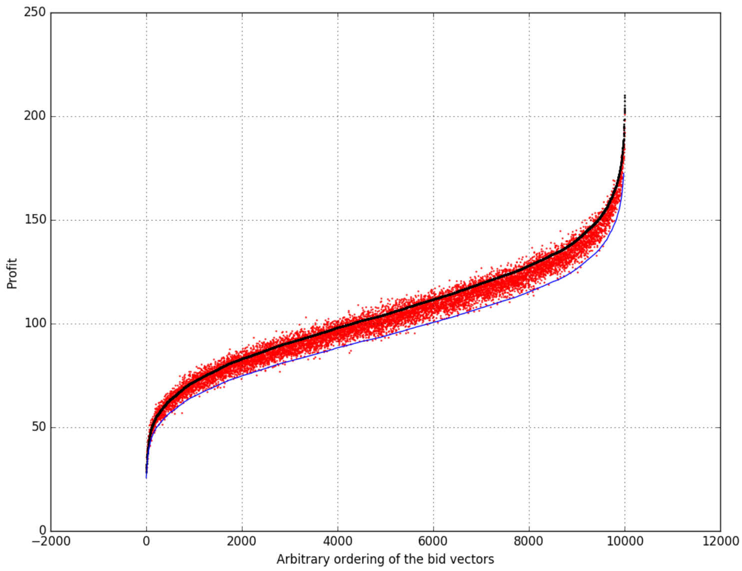
- Authorized bid values: 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 250, 500, 750, 1000.
- Size of bids vector n: 20.
- Values for K: between 2 and 20 (both included).
- Mean of profit computed over 100 replications.
- Number of uniformly sample bid vectors: 10000.

Results

We take a look at some results. Let us first study the results of DSOT^3 compared to DSOT^2 since we use it a lot in the previous section. In the following figure there are two series of profits. The profits of DSOT^2 have been sorted for more visibility and form this neat black trail in the figure, the permutation resulting from the sorting has been applied on the profits on DSOT^3 as well, these profits are represented in red.

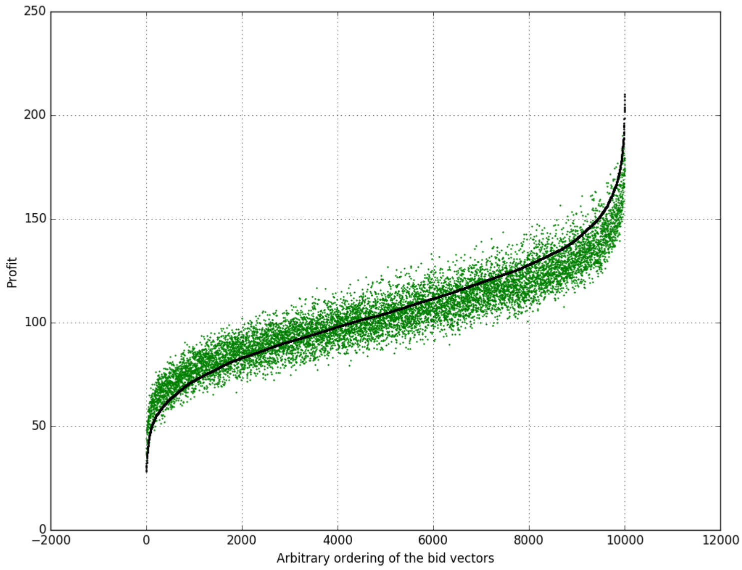


The red trail is mostly lying under the black one, this suggests that on average the $DSOT^2$ auction generates more profit than $DSOT^3$. However the red trail does not deviate too much from the black one. Most of the time a profit of $DSOT^3$ on a certain input is lower-bounded by a ratio of 9/10 of the profit $DSOT^2$ on the same input (blue line in the next figure).



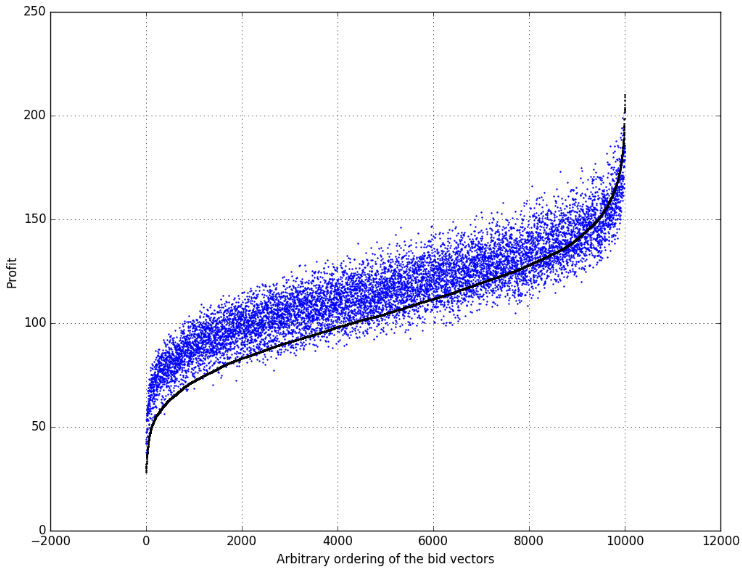
But this experimental ratio does not mean much since it is most likely dependent of the maximum bid value which is limited in our experiment.

Now let us look at the same result but for a larger K, specifically $K = 8$. The profits of DSOT^8 form the green trail in the following figure. The black trail is still the profits of DSOT^2 (exactly the same as before, we are working with the same inputs).



Something interesting is happening: *DSOT^8 outperforms DSOT^2 when the profit of DSOT^2 is the poorest.* On the other hand DSOT^8 appears to do worse than DSOT^2 for the high profits. In the in-between cases it is not clear which auction is the best. We note that the dispersion to the profits of DSOT^2 is greater than the dispersion observed with DSOT^3 (the green trail is more sparse than the red trail).

Considering the evolution jumping from $K = 3$ to $K = 8$. We legitimately want to see what happens with $K = n = 20$. The following figure shows the profits of DSOT^n on the same inputs through the blue trail.



This time DSOT^n outperforms DSOT^2 on most of the inputs. Similarly to the two previous auctions, it is more efficient on the bid vectors on which DSOT^2 scores poorly but worsen a bit on the bid vectors on which DSOT^2 scores a high profit. The dispersion has again increased but is now one-sided (mostly upper bound).

Conclusion

In conclusion, we have proven that the extended version of the DSOT^2: DSOT^K auction can be truthful, and it is competitive against $\mathcal{F}^{(K)}$, even though we are not able to derive a rigorous proof to determine its competitiveness against $\mathcal{F}^{(2)}$. Using the DSOT^2 as a reference, we have also shown that there exists bid vectors on which DSOT^2 and DSOT^K can outperform each other. In particular, from running the experiments to explore the average performance of DSOT^K when K scales (from 2 to 20), we also find out intriguing patterns between profits of DSOT^K and DSOT^2, that is DSOT^K tends to do much better on bid vectors that do poorly on DSOT^2. In addition, as K scales, DSOT^K tends to outperform DSOT^2 in most of the cases.