Competing Mechanisms

This note is meant to illustrate the strange outcomes that are possible in modern computerized markets. Many very unusual things are possible because the mechanisms that sellers use to choose prices are computer programs that can potentially interact with one another. In this note, we'll illustrate how this works when these programs learn what other sellers' programs are doing by interacting with buyers.

A digression on double auctions

Digital markets can be complex. My favorite example is airline ticket pricing. Everyone now knows that prices adjust dynamically in response to what we think are fluctuations in demand. For example, we expect that prices rise when planes become full. Yet it also seems that prices change every time we visit a website to get a quote.

To try to model all this would be a bit complex at this point, so well take simplified approach and let prices be set in something called a *double auction*. This is similar to the auctions that you have already studied, except that in addition to allowing buyers to submit bids describing what they are willing to pay, we'll let sellers submit *ask prices* which describe the price they need in order to sell. The trading price will be set in such a way that the number of buyers who want to buy at that price is equal to the number of sellers who want to sell at that price. So the procedure resembles the simple demand supply model that is taught in first year economics.

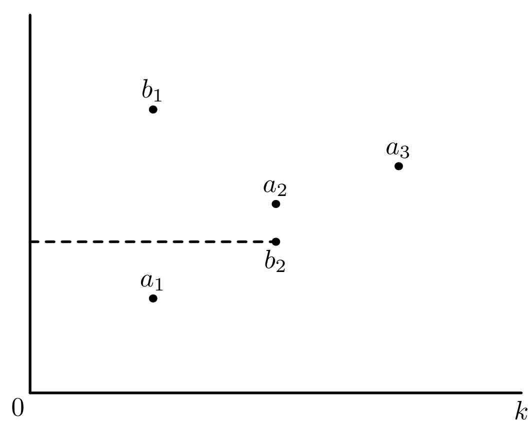
Particulars. In particular, suppose there are n buyers and m sellers. If you want to think of a specific market, think of the set of all cameras that are currently up for auction at eBay. Each camera will be sold by a potentially different seller. More often than not, each of the buyers on eBay wants to buy just one camera. If you visit the eBay website and search for cameras, you see a list of all the auctions for cameras. If you look through all the auctions, you can identify the total number of buyers who are bidding on these cameras. These numbers are the n and m I mentioned in the first sentence of this paragraph.

Instead of using the eBay bidding robot, lets just ask each of the buyers to submit a *bid* describing the amount they want to pay for a camera. Each seller will be asked to submit an *ask* price, which is the price they need to convince them to sell their camera. The auction we are going to use is called a *sellers' offer double auction.* The way it works is that the auction begins by collecting each of the bid and ask prices from the sellers and buyers. There will be a total of $n+m$ of these. Lets name them so we can refer to them again. Let $b = (b_1, \dots, b_n)$ be the profile of bids of the buyers, and $a = (a_1, \ldots, a_m)$ be the profile of ask prices.

The bids and asks are then merged into a single profile, and sorted from highest to lowest. Lets refer to $b_{(i)}$ as the i^{th} highest bid or ask. Remember that $b_{(i)}$ might belong to either a buyer or seller, so it might be an ask price even though it is called $b_{(i)}$. The highest bid or ask is $b_{(1)}$. Of course, if this is a buyer's bid, we probably want to make sure that this buyer trades with some seller, because all the sellers would have asked for less than this buyer says she is willing to pay.

We'll be interested in one element of this profile in particular - $b_{(m+1)}$ - because this is the one we are going to use as a trading price. We are also going to arrange things so that the buyers or sellers who submit the m highest bids or asks end up with a camera. In particular, if a seller submits an ask that is among the m highest bids and asks, we want that seller to keep his camera. In the case where there is a 'tie', that is, the bid price of some buyer is equal to the ask price of some seller, and that common price turns out to be the m^{th} highest bid or ask, then we'll assume that the buyer who submitted the bid will will be named as the m^{th} highest bidder (or asker), while the seller will be treated as the $m + 1^{st}$ highest bid or ask. As always, this means that the buyer would end up with a camera which she would get from the seller who submitted the identical ask price. This also means that the trading price would be equal to this common bid-ask. The buyer would end up paying his bid for the camera, while the seller would en

An example. Here is a simple example



There are two bids by buyers, b_1 and b_2, and three asks by sellers, a_1, a_2 and a_3. If we put these in order we get $b_{(\cdot)} = (b_1, a_2, a_3, b_2, a_1)$. In this case, there are three sellers. We want the buyers or sellers with the three highest bids and asks to end up with a camera, and that would be buyer 1, and sellers 2 and 3. We want the trading price to be the 4^{th} highest bid or ask, and that is b_2. So the outcome in the double auction pictured here is that buyer 1 buys a camera from seller 1 at price b_2.

Notice the way this resembles demand and supply. The horizontal axis represents quantity - k in the picture means number of units. The first unit has a demand 'price' equal to b_1, second unit has price b_2. If you draw a line connecting $b_1$and b_2, it looks like a standard demand curve that you might see in a first year economics course. Do the same with the ask prices by drawing a line joining a_1, a_2, and a_3, so that it looks like a supply curve. These line cross somewhere between b_2 and a_2, and the buyers whose bids are above this market clearing price buy, while sellers whose asks are below this price sell.

What is special about the sellers' offer double auction. If you want to understand why I am going to use b_2 as the trading price just notice that buyer 1 who actually gets to buy a camera can't affect this price by changing her bid, though she could end up not being able to buy anything if she lowers her bid price enough. Since the price is always set by someone who doesn't end up with a camera, a buyer can never affect the price at which she buys. Then just like a second price auction, a buyer can always find a best reply by bidding her value.

The same is unfortunately not true for sellers. To see this, suppose that seller 1 were to raise this ask price just above b_2. He would then be the one with the fourth highest bid or ask, so his ask price would be the trading price. Since he would not be among the buyers or sellers with the 3 highest bids or asks, he should not end up with a camera after all is said and done. So he should end up trading. If he were to raise his ask a bit in this situation, he could raise the price.

So in the sellers' offer double auction, we always know what buyers will do - bid their values. To know what sellers are going to do, we need to know more

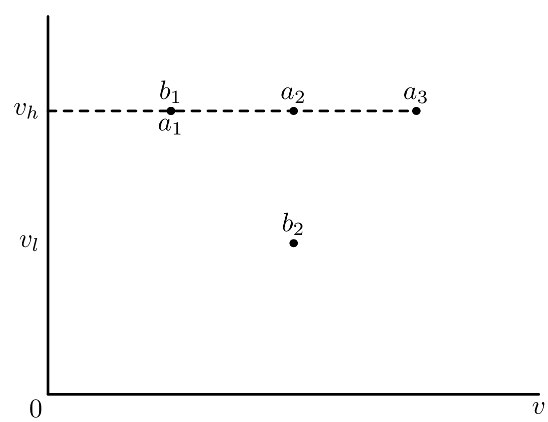
A Special Case

To prep for the competing auction story, lets consider a special case. Suppose there are three sellers and two buyers as above. However, suppose that everyone knows that the sellers have no costs, so they are willing to accept any price at all for their cameras. No one knows what buyers' values are however. To be precise, suppose that each of the three buyers has value v_h with probability π, and value $v_l < v_h$ with probability $(1 - \pi)$. The double auction then represents a game in which players have incomplete information.

As we describe above, the buyers are always going to bid their values in this game. So high value bidders will bid v_h while low value bidders will bid v_l. All we have to do is to figure out what the sellers are going to do. To do it, lets start as we often do by guessing that there is some common price that the sellers ask. The logic will be clear enough if we start by imagining that this price is v_h.

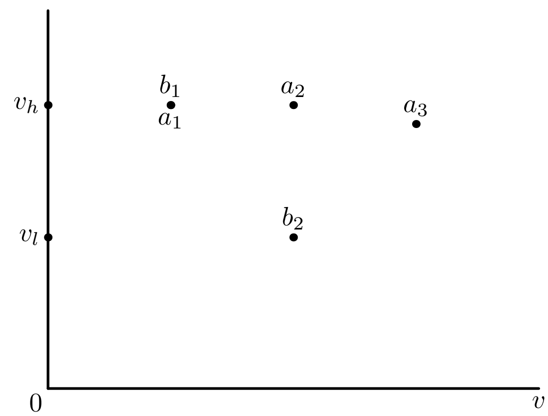
Of course, if both buyers have low values, this won't give the sellers much profit. If one of the buyers has high value, then one of the sellers will sell. If two buyers have high value, two sellers will sell.

One possible outcome. Here is one possible outcome for this situation



The sellers have all asked v_h, as has one buyer in this case. So the 3^{rd} and 4^{th} highest values are both equal to v_h. Since our rule is to favor the buyer when there is a tie, the buyer who bid v_h will trade in this case with one of the sellers who bid v_h. That seller will make a bunch of profit (since his cost is zero), but the other two will be left with nothing since they won't sell.

We can't really say in this case which seller will trade - each of them presumably has a $\frac{1}{3}$ chance of trading and making a profit. So for each of them there is another ask price which would be better. Here is a picture to help explain what it is.



Notice that in this picture, seller 3 has lowered his ask price just slightly. This makes a big difference because the 3 highest bids and asks are now bidder 1 and sellers 1 and 2. They are the ones who should end up with a camera, meaning that seller 3, who has the 4^{th} highest bid or ask, will sell for sure and make almost v_h on his sale, which is considerably better than selling with a $\frac{1}{3}$ chance at price v_h.

Why this isn't an equilibrium. To be complete, we should write out the payoffs properly, since sellers don't know the bids of the buyer or the other sellers at the time they submit their ask prices. In particular, when all the sellers are expected to ask v_h, the expected payoff of each seller can be written as

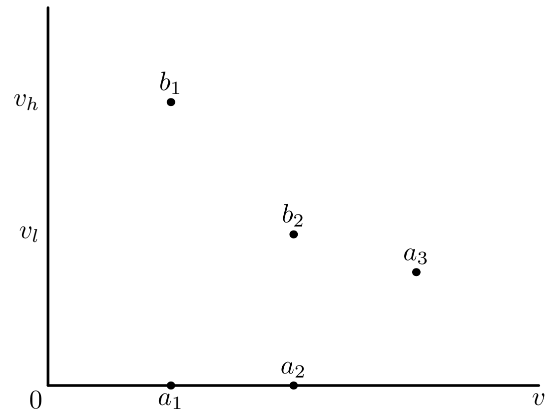
$$2\pi\left(1-\pi\right)\frac{1}{3}v_h+\pi^2\frac{2}{3}v_h$$

while a seller who deviates as describe above will surely trade when there is a high value buyer. His payoff will be arbitrarily close to

$$\left(1-\left(1-\pi\right)^2\right)v_h=2\pi\left(1-\pi\right)v_h+\pi^2 v_h.$$

So the deviation is always profitable.

An equilibrium with all ask prices equal to zero. If, on the other hand, the sellers all ask 0, both buyers will trade no matter what values they have. Since the 4^{th} highest bid or ask will be 0, no seller will make a profit. To show this is an equilibrium, it might help to see the picture.



Here, what seller 3 does is to raise his ask price to a_3, something a bit above 0. He will then be the trader with the 3^{rd} highest bid or ask, so he will end up with a camera - that is, he won't trade. The trading price will still be the 4^{th} highest bid or ask, which is zero. So deviating doesn't do the seller any good. This means there is a Bayesian equilibrium in which each seller asks 0.

A problem to work on. Can you explain why the profile of bids - two zeros and a_3 as given in the picture above, is *not* an equilibrium?

Theory Again - the Reverse Incentive Auction. Calculating payoffs for sellers is complicated in a double auction. However, there is another type of auction that is a bit easier to understand, which has many of the same properties.

A Reverse auction is any auction in which there is a single buyer who purchases one of more units of output using an auction in which all the possible sellers submit ask prices, and the buyer purchases units with low enough ask prices. A procurement auction held by the government would be an example of a reverse auction.

However, what we'll do here is to combine it with a standard 'forward' auction, in which buyers submit bids. Perhaps the best known recent example of such an auction is the recent reverse incentive auction for broadcast spectrum held by the US government over the last few years.[1]

In a reverse incentive auction, sellers submit ask prices for units of output to an intermediary (for example, the US government). At the same time, the intermediary accepts bids from consumers. The difference between a reverse incentive auction and a standard procurement

[1] A series of videos by Tim Roughgarden explain them quite well. https://www.youtube.com/watch?v=jf_2_XHrpmE&list=PLEGCF-WLh2RK6lq3iSsiU84rWVee3A-hz&index=38

auction is that the number of units the intermediary wants to buy is determined endogenously in the reverse incentive auction instead of being exogenously determined by a government department.

In the version of the auction we'll study here, we'll assume that the intermediary uses the pricing rule we described above - the trading price will be the $m+1^{st}$ highest bid or ask, where m is the number of sellers who participate in the auction. The intermediary will sell units of output acquired from sellers by paying them their ask prices. Sellers whose ask prices are no higher than the $m+1^{st}$ bid or ask will sell with positive probability.

As always, we can write down the payoff and describe the trading algorithm at the same time. Lets define the trading price in the following way:

$$p(b,a)=\begin{cases} b_i & \text{if } \exists b_i \left|\{i' \in N : b_{i'} \geq b_i\} \cup \{j' \in M : a_{j'} \geq b_i\}\right| = m \\ a_j & \left|\{i' \in N : b_{i'} \geq a_j\} \cup \{j' \in M : a_{j'} \geq a_j\}\right| = m \text{ otherwise.} \end{cases}$$

This makes it possible to describe payoff functions in the reverse incentive auction.

$$V_b(b_i, b_{-i}, a, v_i) =$$
$$\begin{cases} v_i - p(b_i, b_{-i}, a) & \text{if } b_i > p(b_i, b_{-i}, a) \\ \frac{|\{j \in M : a_j = p(b_i, b_{-i}, a)\}|}{|\{i' \in N : b_{i'} = p(b_i, b_{-i}, a)\}|}\left(v_i - p(b_i, b_{-i}, a)\right) & \text{if } b_i = p(b_i, b_{-i}, a) \\ 0 & \text{otherwise.} \end{cases}$$

For the seller, the payoff is comparable:

$$V_s(b, a_j, a_{-j}) =$$
$$\begin{cases} a_j - c_j & a_j < p(b, a_j, a_{-j}) \\ \frac{|\{i' : b_{i'} \geq p(b, a_j, a_{-j})\}| - \left|\left\{j' : a_{j'} < p(b, a_j, a_{-j})\right\}\right|(a_j - c_j)}{\left|\left\{j' \neq j : a_{j'} = p(b, a_j, a_{-j})\right\}\right| + 1} & a_j = p(b, a_j, a_{-j}) \\ 0 & \text{otherwise.} \end{cases}$$

The tie-breaking rules are complex. Fortunately, we can stick with values and costs that have differentiable distribution functions. Then ties won't matter because they will 'almost never' occur. (That means that a probability that two or more sellers or buyers will submit the same bid or ask is 0).

We'll stick with the 2 buyer 3 seller case and go over the how sellers choose their ask prices and how the selling price is determined. To do this suppose that the distribution of the buyers' values are independently distributed $F(\cdot)$ on $[0,1]$ while sellers costs are independently distributed $G(\cdot)$ on $[0,1]$. Each buyer or seller knows their own values or costs but believes that the values and costs of the others are distributed by F and G respectively. At the point where they submit their bids or asks no buyer or seller knows whether their bid will be successful.

We'll use the sellers' offer pricing rule - the trading price is equal to the $m+1^{st}$ bid or ask. The buyers or sellers with the m highest bids or asks will end up with a unit of output at the end of the auction. Recall that this rule means that the price is determined by someone who doesn't end up with any output, so buyers can do not better than to submit their true values to the auction. Matters are different for sellers.

Sellers will sell and be paid their ask price if they submit an ask that is fourth or fifth highest among all the bids and asks. We can break this probability down into three cases - no buyers are willing to pay is ask price, one is, or both are. These events happen with probability $F^2(a)$, $2(1-F(a))F(a)$ and $(1-F(a))^2$ respectively.

In the first case, the seller isn't going to sell because there is no one to buy. In the second case, he'll sell if both of the other sellers submit higher ask prices. In the third case, he sells as long as at least one of the other sellers has a higher ask price. To calculate these probabilities, we could use the approach we used with first price auctions. We would imagine that all sellers want to use the ask price $a(c)$. Following that story, our seller is trying to choose an ask $a(c')$. The expected profit associated with his choice is going to be

$$2(1-F(a(c')))F(a(c'))(1-G(c'))^2(a(c')-c)+$$
$$\left(1-F^2(a(c'))\right)2(1-G(c'))G(c')(a(c')-c)=$$
$$Q(c',a(c'))(a(c')-c)$$

where

$$Q(c',a(c'))=$$
$$2(1-F(a(c')))F(a(c'))(1-G(c'))^2+\left(1-F^2(a(c'))\right)2(1-G(c'))G(c').$$

Look familiar?

The appropriate choice of c' is given by the first order condition

$$Q(c',a(c'))\frac{da(c')}{dc'}+(a(c')-c)\left(\frac{\partial Q(c',a(c'))}{\partial c'}+\frac{\partial Q(c',a(c'))}{\partial a}\frac{da(c')}{dc'}\right)=0.$$

If the bidding rule a is an equilibrium, this condition evaluated at $c'=c$ should hold uniformly in c. Then we get the differential equation

$$\frac{da(c)}{dc}=\frac{-\left((a(c)-c)\frac{\partial Q(c',a(c'))}{\partial c'}\right)}{Q(c,a(c))+(a(c')-c)\frac{\partial Q(c',a(c'))}{\partial a}}$$

So suppose that exactly one of the two buyers has a bid above a. From your auction theory, you know that this happens with probability $2F(a)(1-F(a))$. Since there is only one buyer willing to pay his ask, our seller is basically bidding in an auction where the lowest ask wins the auction. If he has the lowest ask in this case, he must have the fourth highest bid or ask. One buyer has a value higher than a (1), and there are two losing sellers whose asks are higher than a (2+1=3).

Assuming the seller's ask a is strictly above his cost, his profit in this case is $(a-c)$. The probability that a is the lowest ask is equal to the probability that the other sellers submit higher bids. To find this, lets use the same technique we used with auctions. We'll assume that all sellers use the same monotonically increasing ask function $a(c)$. If our seller bids as if his cost were c' then his profit would be $(a(c')-c)$ (which would be strictly positive if $c' > c$).

Then the probability of this event is just

$$2F(a(c'))(1-F(a(c')))(1-G(c'))^2 \tag{0.1}$$

It is also possible that both buyers submit bids above $a = a(c')$. Now if our seller has the lowest ask, he will certainly sell, because the other four would all have higher bids or asks. Yet since he has the lowest bid or ask, the selling price he gets will depend on the fourth highest bid or ask, which, in this case must be one of the other sellers.

Competing Mechanisms

Now we can explain the strange way that a modern digital market could work. Suppose that everyone knows that each seller's ask price is submitted by a computer program that responds to the number of buyers who visit the seller's website. No one sees exactly what this program is, but they know it is there and that it can't be easily changed. They can also see the price that this program will submit if each buyer simply visits the website once and looks at this price. This price is the analog of the price you are offered for a flight the first time you ask for it. Though the airline is currently committed to that price, everyone knows it might change if you don't take it right away.

Of course, in the double auction, you can't 'take it right away'. So this opening price is understood by everyone to be the ask price the program will submit if both buyers visit the website once, then never return. What happens if buyers come back a second time is not known to anyone - but this is what everyone thinks the program on the website will do:

$$a(m) = \begin{cases} v_h & m \leq 1 \\ 0 & \text{otherwise.} \end{cases} \tag{0.2}$$

Here m is the number of buyer who come back to visit the website a second time. When no buyers come back, or if only one of them does, the seller's program will stick with the initial offer v_h, while if both come back for a second visit, the program will adjust the price downward to 0.

This is an extensive game of incomplete information. Sellers write their computer programs and display their initial price on their websites. Buyers visit the websites, and depending on what they see, they can choose to visit websites for the second time. Finally buyers submit their bid prices, while sellers' programs submit their ask prices for them.

We can now describe the equilibrium using backward induction. Sellers have nothing to do but write their programs, so we'll leave that until last. Buyers make their final decision when they submit their bids. As we use a seller's offer double auction, a best reply for buyers will always involve bidding their value, whether it is high or low.

Stepping back one decision, buyers have to decide just before this about website visits. Lets suppose that if they see both sellers opening prices are equal to v_h, then they don't bother to visit websites, and just wait for the double auction to occur. If any seller's opening offer is different from v_h, then the buyers re-visit all three websites. At their first move, they simply visit all three websites to see the opening offers).

For sellers, lets suppose that each of them uses the program described by (0.2). Now lets argue that this is a Bayesian equilibrium.

Notice that the only thing that buyers do that depends on their type is their final bid in the double auction. If buyers see only v_h when they visit websites and expect sellers to use programs as described by (0.2), then there is no point visiting websites. They don't expect the other bidders to visit websites, so if they decide to visit on their own, each of the sellers will see $m = 1$ and set price v_h. As the buyers believe the sellers are using the program (0.2), there is no gain to re-visiting if the buyer only sees the offer v_h.

This argument would be strengthened if buyers incur a cost when they visit websites.

The interesting case occurs when a buyer sees a price different from v_h at one of the seller's websites. In that case, he believes that the other buyer will also see that price, and he expects that will cause the other buyer to revisit. For that reason, each of the sellers will receive $m = 1$ website visit if the buyer decides not to go. Each of them, apart from the deviator will then set price v_h according to (0.2). On the other hand, if he decides revisit himself, that will change the number of website re-visits for each seller from 1 to 2, causing at least two of the sellers to lower their ask price to zero. Whatever the deviator's ask is, the 4^{th} highest bid or ask will then be 0.

So the buyer has a lot to gain by revisiting as well.

On the other side of the market, the deviating seller can't win in this situation - if he changes his price from v_h he will end up not selling, so he is better to leave well enough alone.