Allocative Externalities in Auctions

An Efficiency & Revenue Analysis for Standard Sealed Bid and Vickrey Auctions

---

Author

Bart de Koning

Supervisor

Dr. Vladimir Karamychev

Study Programme

Master in Economics & Business

Institute

Erasmus University Rotterdam

Date

6 August 2017

Abstract

In this paper, the first price sealed bid auction, second price sealed bid auction and Vickrey auction are compared in terms of efficiency and revenue in presence of heterogeneous allocative externalities. The results show that the smaller the number of participating firms, the better the standard sealed bid auctions perform in terms of efficiency. When the size of the externalities compared to the firms' payoffs of winning themselves decrease, standard sealed bid auctions are also more efficient. Revenue of the standard sealed bid auctions is generally higher than that of the Vickrey auction for a low number of participating firms. However, if externalities are strongly heterogeneous, the Vickrey auction may yield higher revenue. Moreover, the Vickrey auction's revenue increases in the number of participating firms more rapidly than that of the standard sealed bid auctions. The paper also shows how a market model underlying the auction may influence the auctions' outcomes and is thus an important consideration in evaluating auctions.

1 Introduction

Auctions have been a way to sell goods for centuries. They are especially common for selling art, classic cars and flowers, where they generally ensure that the goods are won by the bidder that values them most. In these auctions, the valuation of the bidders depends only on what they feel the auctioned good is worth to them. However, this is not always the case. In some auctions, bidders' valuations depend also on how much they value someone else not getting the good; there are allocative externalities.

A typical example is the auction of mining leases. In such an auction, multiple companies bid for the right to extract raw material from a mine. A firm that intends to resell the raw materials may be very worried about another reseller getting its hands on the license as it worsens its market position. On the other hand, such a firm may barely care about firms with intentions other than resale getting the license.

Another example of an auction with allocative externalities is the patent auction. Although patents can be licensed, the patent itself is a single, indivisible good and can be held by only one company. A patent can at times be put to use in a range of markets. For any firm in a patent auction, the worst case scenario is that another firm in the same market gets the patent as this leads to a comparative disadvantage. Indeed, auctions with allocative externalities are relatively common in the real world. The first price auction is most often used for selling both mining leases (McAfee and McMillan, 1987) and patents (Haouideg, 2016). However, this may not be the right choice.

This paper focuses on the performance of different auctions in terms of efficiency (welfare optimisation) and revenue in presence of allocative externalities. Efficiency is often important in government-run auctions such as that of mining leases. Instead of maximising the auctioneer's revenue, maximising social welfare is paramount. In privately run auctions, such as patent auctions, revenue is often what the auctioneer is most concerned with. Obviously, the goals are not mutually exclusive and it may be that an auctioneer cares about both efficiency and revenue.

The goal of this paper is to find out how different auctions for single, indivisible goods perform in terms of efficiency and revenue in the presence of heterogeneous allocative externalities. This is done to provide the reader with insights into what auction is best given a context and a goal. The approach is two-sided. Firstly, the theory behind the different auctions is covered. The paper shows the inner workings of the auctions and why and when they fail at attaining efficiency or high revenue. Secondly, by performing Monte Carlo simulations under a set of assumptions, the paper provides the reader with a comparison of the auctions' performance in (simulated) practice.

The auctions considered are the first price sealed bid auction, second price sealed bid auction and the Vickrey auction. The standard sealed bid auctions are commonly used in practice. As stated, the first price sealed bid auction is often used for the auction of mining leases and patents. An example of (a modified version of) the second price sealed bid auction being used in practice is Google's AdWords auction (Google, 2017). The Vickrey auction is not as common in practice, but is a nice benchmark as the auction is efficient in a wide range of contexts, including that of allocative externalities (MacKie-Mason and Varian, 1994; Ausubel et al., 1999; Morgan, 2001).

This paper's results show that the Vickrey auction's revenue is rather low when the number of participating firms is low as well or when externalities are relatively homogeneous. However, as the number of firms increases or externalities become more heterogeneous, the Vickrey auction's revenue grows rapidly. The Vickrey auction does come with the highest variance in revenue.

The first price sealed bid auction and second price sealed bid auction perform very similar to each other. If the number of participating firms is low or the firms' payoffs of receiving the good themselves are much higher than the average externality, the mentioned auctions allocate the good such that welfare is optimised way more often than if it were allocated randomly. However, as the number of firms increases, the standard sealed bid auctions perform worse and worse when it comes to efficiency. The average revenue of the first price sealed bid auction and second price sealed bid auction is equal. It increases steadily as the number of firms increases, although not as rapid as the Vickrey auction's revenue. The second price sealed bid auction does have higher variance in revenue than the first price sealed bid auction.

The paper also shows that the performance of an auction is strongly dependent on the underlying market model. Thus, auctions cannot be adequately reviewed without considering market dynamics in combination with the auctioned good. In this paper, a slightly modified Hotelling model is considered. In this model, the standard sealed bid auctions are shown to actually perform rather well in terms of efficiency when the auctioned good either has a small or large effect on the market, regardless of the number of participants. When it comes to revenue, the Vickrey auction performs worse than the standard sealed bid auction in almost all scenarios.

The remainder of this paper is organised as follows. Section 2 provides an overview of the literature on the topic. In section 3, the mathematical model and the auction rules are laid out. Section 4 and 5 cover a theoretical analysis of the Vickrey auction and the standard sealed bid auctions, respectively. In section 6, results from a Monte Carlo simulation assuming uniform distribution of parameters are presented. Section 7 covers the analysis for when there is a market model underlying the auction. Section 8 concludes.

2 Literature Review

In 1961, Vickrey showed that, for a single indivisible good, in the case of an independent private value auction without externalities, an efficient auction exists in the form of the second price sealed bid auction. The mechanism behind this was later generalised by Clarke (1971) and Groves (1973) and is often referred to as the 'Vickrey-Clarke-Groves mechanism'. An auction that applies this mechanism to ensure truthful bidding and efficient allocation is called a 'Vickrey auction'. Generally, the Vickrey auction makes use of the Clarke pivot rule. This rule allows the auctioneer to actually turn a profit instead of having to pay participating firms to truthfully reveal their preferences. The idea is that a participating firm pays a price to the auctioneer that is equal to the welfare of other firms that would have been realised had the firm not participated subtracted by the realised welfare of the other firms. It is important to note that the second price sealed bid auction is a Vickrey auction in the case of a single good auction with independent private valuations and no externalities, but may not be a Vickrey auction under different circumstances.

MacKie-Mason and Varian (1994), Ausubel et al. (1999) and Morgan (2001) proved that the Vickrey auction leads to truthful revelation and subsequent efficient allocation for divisible goods, multiple goods, an auction with both a buyer and a seller, interdependent valuations and congestion externalities as well. Parkes and Ungar (2001) showed that the Vickrey auction is also individual-rational and weakly budget balanced if there is choice set monotonicity, there are no negative externalities and the *no-single agent effect* condition holds. However, while the Vickrey auction always leads to an efficient allocation of the auctioned good(s), it has its drawbacks. The Vickrey auction is very sensitive to collusion, may not yield high revenue and is vulnerable to shill bids (Ausubel et al., 2006).

Because of its drawbacks, the Vickrey auction is often substituted for standard auctions such as the first price sealed bid auction and the second price sealed bid auction. The first price sealed bid auction is particularly strong at deterring collusion. Even if colluders are promised a share of the ex-post profits and valuations can be credibly revealed, collusion is still not a Nash equilibrium (Robinson, 1985). Revenue in these auctions is generally higher because firms have to incorporate the possibility of incurring externalities into their valuation and thus bid (Jehiel et al., 1999). Shill bids have no function in the standard sealed bid auctions as they can never decrease the price the winner has to pay.

While these auctions have some favourable characteristics compared to the Vickrey auction, they are not necessarily efficient (Das Varma, 2002) (Jehiel and Moldovanu, 2005). Only when externalities are equal among firms are the standard sealed bid auctions efficient (Das Varma, 2002). This lack of efficiency is caused by the fact that, in standard sealed bid auctions, bidders can only post a single bid that represents their valuation for all possible scenarios (Jehiel et al., 1999).

The equilibrium bidding strategies for a firm in the first price sealed bid auction is to bid the expected value of the highest valuation that is not its own, given that it has the highest valuation itself (Acemoglu and Ozdaglar, 2009). The second price sealed bid auction has a more straightforward equilibrium bidding strategy; firms then bid their valuation (Jehiel et al., 1999).

Jehiel et al. (1999) showed that in the case of allocative externalities, the second price sealed bid auction with entry fee maximises revenue for the auctioneer. However, it does not go into detail on what the revenue properties of the regular first price and second price sealed bid auction are. In 2005, Jehiel and Moldovanu showed that the standard auctions need not be efficient, but did not proceed to show how bad exactly they perform at optimising welfare.

The above literature is the point of departure for this paper. As stated, from the existing knowledge an attempt is made to answer the questions of the auctions' efficiency and revenue properties in the presence of heterogeneous allocative externalities.

3 The Model

3.1 Notation & Definitions

The model in this paper considers n firms, indexed by $i = 1, ..., n$. All firms participate in the auction. The auctioneer is not considered to be a participant and will always allocate the good to a firm.

Following Jehiel et al. (1999), a firm's preferences are defined by a vector $\boldsymbol{t^i} = \{t_1^i, ..., t_n^i\}$, where t_i^i denotes firm i's payoff when it gets the auctioned object and t_j^i, where $j \neq i$, denotes firm i's payoff when firm j gets the object. The total welfare given that firm i wins, is defined by $w_i = t_i^i + \sum_{j \neq i}^{n} t_i^j$. Table 1 provides a visual representation of the payoffs and the corresponding welfare and vectors of preferences. The columns indicate which firm wins and the rows indicate for whom the payoff is.

Table 1: Visual Representation of Payoffs

	1	2	...	n	$\in$
1	t_1^1	t_2^1	...	t_n^1	$\boldsymbol{t^1}$
2	t_1^2	t_2^2	...	t_n^2	$\boldsymbol{t^2}$
...	...	...	...	...	...
n	t_1^n	t_2^n	...	t_n^n	$\boldsymbol{t^n}$
$\sum$	w_1	w_2	...	w_n	

It is assumed that t_i^i is a randomly distributed variable on range $[0, \alpha]$ with probability density function $f_{t_i^i}(x)$ and cumulative distribution function $F(x)$. t_j^i is assumed to be distributed on range $[\beta, 0]$ (where $\beta < 0$) with probability density function $f_{t_j^i}(x)$ and cumulative distribution function $G(x)$. The allocative externalities in this paper are thus considered to be non-positive. It is assumed that the distribution of parameters is symmetric; all t_i^i are equally and independently distributed and so are all t_j^i. The ex-ante distribution of t_i^i and t_j^i is common knowledge to all firms. Firms' actual preferences are private information.

A firm's utility is defined by $u^i = t_{winner}^i - p^i$, where 'winner' denotes the index of the firm that does get the good (wins) and p^i denotes the price firm i has to pay for the allocation.

An auction is considered to be (ex-post) efficient if the auction allocates the good such that $max\{w_1, ..., w_n\}$ is always attained. In practice, this means that after-auction trade cannot lead to a pareto improvement.

3.2 Rules of the Vickrey Auction

In the Vickrey auction, firms can freely communicate their preferences to the auctioneer and are thus not restricted to a single bid. The firm that, if it wins, maximises social welfare according to the communicated preferences is indeed declared the winner. It may be that multiple firms have to pay. A firm's payment is defined by $p^i = max\{w_1^{-i}, ..., w_n^{-i}\} - \sum_{j \neq i}^{n} t_{winner}^{j}$. Here, $max\{w_1^{-i}, ..., w_n^{-i}\}$ denotes the welfare that would have been realised if firm i's preferences were not taken into account. $\sum_{j \neq i}^{n} t_{winner}^{j}$ are the realised payoffs for all firms but i. To illustrate this, consider the payoff scheme as reported by the firms presented in table 2.

Table 2: Example Values Vickrey Auction

	1	2	3
1	0.8	−0.5	−0.6
2	−0.3	0.6	−0.9
3	−0.8	−0.6	0.7
$\sum$	−0.3	−0.5	−0.8

Firm 1 is rewarded the good as $w_1 > w_2, w_3$. Firm 1 has to pay $p^1 = (0.6 - 0.6) - (-0.3 - 0.8) = 1.1$, firm 2 has to pay $p^2 = (0.7 - 0.6) - (0.8 - 0.8) = 0.1$ and firm 3 has to pay $p^3 = 0.8 - 0.3 - (0.8 - 0.3) = 0$.

3.3 Rules of the Standard Sealed Bid Auctions

Contrary to the Vickrey auction, in a standard sealed bid auction, a firm has to post a single bid b^i to represent all of its preferences. The firm that posts the highest bid is declared winner. If there are multiple firms that posted the highest bid, the good will be allocated randomly to one of these firms. Also in contrast to the Vickrey auction, only the winner has to pay price p^i. For the first price sealed bid auction, if firm 1 wins, $p^1 = b^1$; the winning firm pays its own bid. If firm 1 wins the second price sealed bid auction, $p^1 = max\{b^2, ..., b^n\}$; the winning firm pays the maximum of all bids that are not its own.

4 Theoretical Analysis Vickrey Auction

4.1 Bidding Strategies

As proven by MacKie-Mason and Varian (1994) and Ausubel et al. (1999), firms participating in the Vickrey auction will always truthfully reveal their preferences. In light of the model, this means that each firm will make its vector $\boldsymbol{t^i}$ known to the auctioneer. To show that non-participation is a (weakly) dominated strategy in case of allocative externalities it should be proven that the outcome of participating in the mechanism is always higher or equal to no participation in the mechanism (Parkes and Ungar, 2001).

If the firm's participation does not change the final outcome of the auction compared to when it would have participated, $p^i = 0$. This is because the situation does not change: $max\{w_1^{-i}, ..., w_n^{-i}\} = \sum_{j \neq i}^{n} t_{winner}^{j}$. In this case, there is no unwanted cost as a result of participation.

If its participation does change the outcome, it is never in the participating firm's disadvantage. If $max\{w_1^{-i}, ..., w_n^{-i}\} - \sum_{j \neq i}^{n} t_{winner}^{j} > 0$, it must be that the firm changed the outcome of the auction. Therefore, $t_{winner}^{i} - t_{k}^{i} > max\{w_1^{-i}, ..., w_n^{-i}\} - \sum_{j \neq i}^{n} t_{winner}^{j}$, where k denotes the would-be winner had firm i not participated. The following example helps illustrate that participation is better than non-participation if a firm's participation indeed changes the outcome. Assume that, without loss of generality, firm 1 wins the auction and that firm 2 would have won it had firm i not participated. Firm i may or may not be firm 1. The fact that firm 1 won the auction means that $t_1^i + \sum_{j \neq i}^{n} t_1^j > t_2^i + \sum_{j \neq i}^{n} t_2^j$. Firm i's payment is positive if $\sum_{j \neq i}^{n} t_2^j > \sum_{j \neq i}^{n} t_1^j$. For both inequalities to hold, it follows that $t_1^i - t_2^i > \sum_{j \neq i}^{n} t_2^j - \sum_{j \neq i}^{n} t_1^j$, which means that $t_1^i - t_2^i > p^i$. As firm i's difference in utility from firm 1 winning and firm 2 winning equals $t_1^i - t_2^i$, its participation is beneficial to the firm in this case. This result combined with the result of non-influential participation shows that it is weakly better for a firm to participate in the auction than to not participate.

4.2 Efficiency

Because of truthful revelation, the auctioneer can accurately find $max\{w_1, ..., w_n\}$ and reward the good to the firm that maximises total welfare. The auction is thus efficient.

Consider the example values in table 2 again. It can be shown that there is indeed no trade possible that leads to a pareto improvement.

As firm 1 is awarded the good, the utilities in the current situation are $u^1 = 0.8 - 1.1 = -0.3$, $u^2 = -0.3 - 0.1 = -0.4$ and $u^3 = -0.8$. Firm 1 is indifferent between keeping the good and being compensated 1.3 for the good to be allocated to firm 2. It has to be compensated 1.4 to be indifferent between keeping the good and it being allocated to firm 3. Firm 2 has to be compensated 0.6 for re-allocation to be indifferent between the current situation and the good being allocated to firm 3. It is willing to pay 0.9 to get the good itself. Firm 3 is willing to pay 0.2 to have the good be awarded to firm 2 and is willing to pay up to 1.5 to get the good itself. The only feasible trade is between firm 3 and firm 1, but if this trade were to take place, firm 2 would be worse off, even if firm 1 and 3 allocate their surplus of the trade to firm 2. In short, it is not possible for after-auction trade to lead to a pareto improvement.

4.3 Revenue

Revenue in the Vickrey auction is defined by $\Pi = \sum^n p^i$. It is driven by both the firms' payoffs of getting the good themselves and the present allocative externalities.

The effect of the firms' payoffs of getting the good themselves is clear. The higher these payoffs, the higher $\mathbb{E}[max\{w_1^{-i}, ..., w_n^{-i}\}]$ and therefore the higher the first part of p^i.

The effect of the allocative externalities is less straightforward. Revenue is not increasing in the level of these externalities per se, but rather in the combination of their size and distribution. To illustrate this, consider the following three payoff cases of three firms participating in a Vickrey auction. It is assumed that firm 1 wins all the auctions.

Table 3: Example Values Case 1

	1	2	3
1	1.0	-1.0	0.0
2	0.0	1.0	-1.0
3	-1.0	0.0	1.0
$\sum$	0.0	0.0	0.0

The values as presented in table 3 are such that the size of the externalities firms cause to others are strongly heterogeneous. In this case, $p^1 = 2$, $p^2 = 1$ and $p^3 = 0$. The total revenue is then $\Pi = 3$, which is the maximum level of revenue that can be attained in the Vickrey auction for $n = 3$ and the maximum (minimum) possible value of t_i^i (t_j^i) being 1 (-1).

Table 4: Example Values Case 2

	1	2	3
1	1.0	-1.0	-1.0
2	-1.0	1.0	-1.0
3	-1.0	-1.0	1.0
$\sum$	1.0	1.0	1.0

In table 4, all externalities are large, but the externalities incurred and caused are homogeneous for all firms. In this case, $p^1 = 2$, $p^2 = p^3 = 0$. The total revenue is thus $\Pi = 2$. Even though the total value of present externalities is larger, the revenue is lower than it was before.

Table 5: Example Values Case 3

	1	2	3
1	1.0	-1.0	-1.0
2	0	1.0	-1.0
3	0	-1.0	1.0
$\sum$	1.0	1.0	1.0

In table 5, the size of the externalities a firm causes to others is homogeneous, but not always large. Now, $p^1 = p^2 = p^3 = 0$, which means $\Pi = 0$. Even though the total value of externalities present is still larger than in the first example, revenue is now at its absolute lowest for when $n = 3$ and the maximum (minimum) possible value of t_i^i (t_j^i) being 1 (-1).

The cases above show that the Vickrey auction's revenue is not simply increasing in the externalities present, but depends for a large part on the distribution of those externalities. If all firms incur a very strong externality of a certain competitor (that is different for each firm) winning and do not care as much about other firms, the Vickrey auction yields high revenue. If, however, the externalities are very similar for all firms, the revenue need not be high, even if the externalities are all large. If there is a participating firm that hardly causes any externalities, the revenue of the Vickrey auction is likely to be very low.

4.4 Collusion

While section 4.2 and 4.3 show that the Vickrey auction performs very well in terms of efficieny and may have high revenue as well, it is interesting to see what drives its biggest weakness: the sensitivity to collusion. To illustrate this, consider the actual firms' payoffs as presented in table 6.

Table 6: Actual Payoffs

	1	2	3
1	0.1	0.0	-0.4
2	0.0	0.1	-0.4
3	0.0	0.0	1.0
$\sum$	0.1	0.1	0.2

In the case of truthful revelation, firm 3 wins the auction and the following utilities are realised: $u^1 = -0.4$, $u^2 = -0.4$ and $u^3 = 1 - [0.1 - (-0.4 - 0.4)]$. While firms 1 and 2 have no incentive to deviate separately, they do have an incentive to collude. If firms 1 and 2 state that $t_3^1 = t_3^2 = -0.9$, the reported payoff scheme in table 7 is realised.

Table 7: Reported Payoffs When Colluding

	1	2	3
1	0.1	0.0	−0.9
2	0.0	0.1	−0.9
3	0.0	0.0	1.0
$\sum$	0.1	0.1	−0.8

Because of the exaggeration, firm 1 (or 2) winning seemingly optimises welfare. Consequently, the good is rewarded to either firm 1 or firm 2. Assume for simplicity's sake that firm 1 wins the auction. Then, $p^1 = 0.1 - 0 = 0.1$, $p^2 = p^3 = 0.1 - 0.1 = 0$. Because both firm 1 and 2 exaggerated, it seems like the auction's outcome is similar for when both participate and for when one of them does not. Therefore, the two firms' payments are very low, which is what makes colluding so attractive. The firms' utilities will be $u^1 = 0.1 - [0.1 - 0] = 0$. $u^2 = 0 - [0.1 - 0.1] = 0$ and $u^3 = 0$.

Clearly, firm 1 and 2 are better off now than they were before. Of course, the feasibility of collusion hinges on whether the firms trust each other to both exaggerate or not. However, in this case, the colluding firms do not have any incentive to deviate, so it is reasonable to believe that neither of them will.

5 Theoretical Analysis Standard Sealed Bid Auctions

5.1 Bidding Strategies

5.1.1 Valuations

In the standard sealed bid auctions, firms base their bids on a valuation defined by $v^i = t_i^i - \frac{1}{n-1}\sum_{j\neq i}^{n} t_j^i$ (Jehiel et al., 1999). It is clear that $\frac{\partial v^i}{\partial t_i^i} > 0$ and $\frac{\partial v^i}{\partial t_j^i} < 0$; a firm's valuation is increasing in the firm's payoff of receiving the good itself and grows smaller as the size of the average externality decreases.[1] This makes intuitive sense. If firm i incurs only a very small externality if firm j wins, acquiring the good will be less important to firm i than when it incurs a large externality if firm j wins. Note that, as n increases, the difference in valuations between firms depends more and more on the value of t_i^i. The reason for this

---

[1] Externalities are always negative, so a small externality is close to zero.

is that for all firms, $\frac{1}{n-1}\sum_{j\neq i}^{n} t_j^i$ will converge to the average value of t_j^i as n approaches infinity.

5.1.2 First Price Sealed Bid Auction

In the first price sealed bid auction, the bidding strategy for firm i in a symmetric equilibrium is formally defined by $b^{i^*} = \mathbb{E}[max\{\boldsymbol{v^j}\}|v^i > max\{\boldsymbol{v^j}\}] = \frac{1}{F_{v^j}^{max}(v^i)}\int_0^{v^i} y f_{v^j}^{max}(y)dy$. Here, $\boldsymbol{v^j}$ is a vector of all valuations except for v^i, $F_{v^j}^{max}(v^i)$ is the cumulative distribution of the maximum valuation of all firms other than i, $f_{v^j}^{max}(y)$ is its probability density function. The higher a firm's own valuation, the higher the expected value of the second highest valuation given that it wins. In short, b^{i^*} is increasing in v^i (Acemoglu and Ozdaglar, 2009). $F_{v_j^{max}}(v^j) = F_{v^j}(v^j)^{n-1}$ and $f_{v_j^{max}}(v^j) = (n-1)F_{v^j}(v^j)^{n-2}f_{v^j}(v^j)$. Thus, a firm's equilibrium bidding strategy in the first price sealed bid auction is defined by $b^{i^*} = \frac{1}{F_{v^j}(v^j)^{n-1}}\int_0^{v^i} y(n-1)F_{v^j}(y)^{n-2}f_{v^j}(y)dy$.

5.1.3 Second Price Sealed Bid Auction

The equilibrium bidding strategy in the second price sealed bid auction is more straightforward. As stated, firms then bid their valuation. This means that $b^{i^*} = v^i$ (Jehiel et al., 1999).

5.2 Efficiency

It is known that for both the first price sealed bid auction and the second price sealed bid auction, b^{i^*} is increasing in v^i. Therefore, the firm with the highest valuation will always win in equilibrium: $Pr[b^i > b^j|v^i > v^j] = 1$. As $\frac{\partial v^i}{\partial t_i^i} > 0$ and $\frac{\partial v^i}{\partial t_j^i} < 0$, if firm a has a higher bid than firm b: $\mathbb{E}[t_a^a|v^a > v^b] > \mathbb{E}[t_b^b|v^a > v^b]$ and $\mathbb{E}[t_j^a|v^a > v^b]\forall j \neq a < \mathbb{E}[t_j^b|v^a > v^b]\forall j \neq b$. Therefore, $\mathbb{E}[w_a|v^a > v^b] > \mathbb{E}[w_b|v^a > v^b]$: the higher a firm's valuation, the more likely it is that welfare is optimised if it wins. This means that, in equilibrium, the good is allocated such that it optimises welfare more often than when it were randomly allocated. However, this does not mean welfare is *always* optimised. It is possible that $[w_a|v^a > v^b] < [w_b|v^a > v^b]$. To illustrate this, consider the example values presented in table 8.

While firm 1 winning would lead to the highest welfare, firm 3 has the highest valuation and will thus win the auction. It is possible to show that, if firm 3 wins, after-auction trade

Table 8: Example Values Standard Sealed Bid Auctions

	1	2	3	v^i
1	0.8	−0.5	−0.6	1.35
2	−0.3	0.6	−0.9	1.2
3	−0.8	−0.6	0.7	1.4
$\sum$	−0.3	−0.5	−0.8	

can lead to a pareto improvement. The auction is thus not efficient. A second price sealed bid auction is considered because in this auction the firms' bidding strategies are more straightforward than in the first price sealed bid auction. In the current situation, utilities are $u^1 = -0.6$, $u^2 = -0.9$ and $u^3 = 0.7 - 1.35$. Firm 1 is willing to pay up to 1.4 to get the good itself and up to 0.1 for the good to be awarded to firm 2. Firm 2 is willing to pay 0.9 to get the good itself and willing to subsidise 0.6 for the good to be awarded to firm 1. Firm 3 is willing to hand over the good to firm 1 for 1.5 and to firm 2 for 1.3. A pareto improvement is possible if firm 3 sells the good for more than 1.5 to firm 1, who pays up to 1.4 and is subsidised for the remaining amount necessary by firm 2. This means the auction's outcome is inefficient.

5.3 Revenue

The valuations and bidding strategies described in section 5.1 result in equivalent revenue for the first price- and second price sealed bid auction. To see that this is the case, assume that, without loss of generality, firm i has the highest valuation. The expected revenue in the first price sealed bid auction then equals $b^{i^*} = \mathbb{E}[max\{\boldsymbol{v^j}\}|v^i > max\{\boldsymbol{v^j}\}]$. For the second price sealed bid auction, the auctioneer's expected revenue equals $\mathbb{E}[max\{\boldsymbol{b^j}\}|b^i > max\{\boldsymbol{b^j}\}]$. Because $b^i = v^i$ in the second price sealed bid auction equilibrium, this can be rewritten to $\mathbb{E}[max\{\boldsymbol{v^j}\}|v^i > max\{\boldsymbol{v^j}\}]$, which is equal to the expected revenue in the first price sealed bid auction.

While the expected value of revenue is equal, the variance of the revenue differs between the two auctions. A source of variance for both auctions is the randomness of which firm wins the auction and what that firm's valuation is. For the first price sealed bid auction, that is the only source of variance as firms pay their bid, which is defined by $b^{i^*} = \frac{1}{F^{max}_{v^j}(v^i)} \int_0^{v^i} y f^{max}_{v^j}(y)dy$. In the second price sealed auction, assuming that firm i has the highest valuation, the auctioneer's revenue is given by $max\{\boldsymbol{v^j}\}$ distributed according to $[f^{max}_{v^j}(v^i)|v^i > v^j]$. This is a random variable and thus another source of variance.

6 Monte Carlo Simulations

To compare the performance of the auctions dicussed in sections 4 and 5, a number of Monte Carlo simulations are run. For the code used to run these simulations, please see Appendix A 'Python Code Uniform Parameter Distribution'. The aim of this simulation is to uncover how the standard sealed bid auctions and Vickrey auction differ in terms of welfare optimisation and revenue for a given parameter distribution. Unless indicated otherwise, all t_i^i are distributed uniformly on range $[0, 1]$ and all t_j^i are distributed uniformly on range $[-1, 0]$. For $n = 3$ and $n = 4$, a billion iterations are done. For $n = 8$, 100 million iterations are done and for $n = 16$, 50 million iterations are done.

6.1 Welfare

Table 9 shows how often welfare is optimised. If the auctioned good is allocated randomly, the percentage of times it is allocated optimally equals $\frac{1}{n}$.

Table 9: % of Times Welfare Optimised

	Random Allocation	Standard Sealed Bid Auctions	Vickrey Auction
$n = 3$	33%	64.46%	100%
$n = 4$	25%	50.15%	100%
$n = 8$	12.5%	25.70%	100%
$n = 16$	6.25%	10.32%	100%

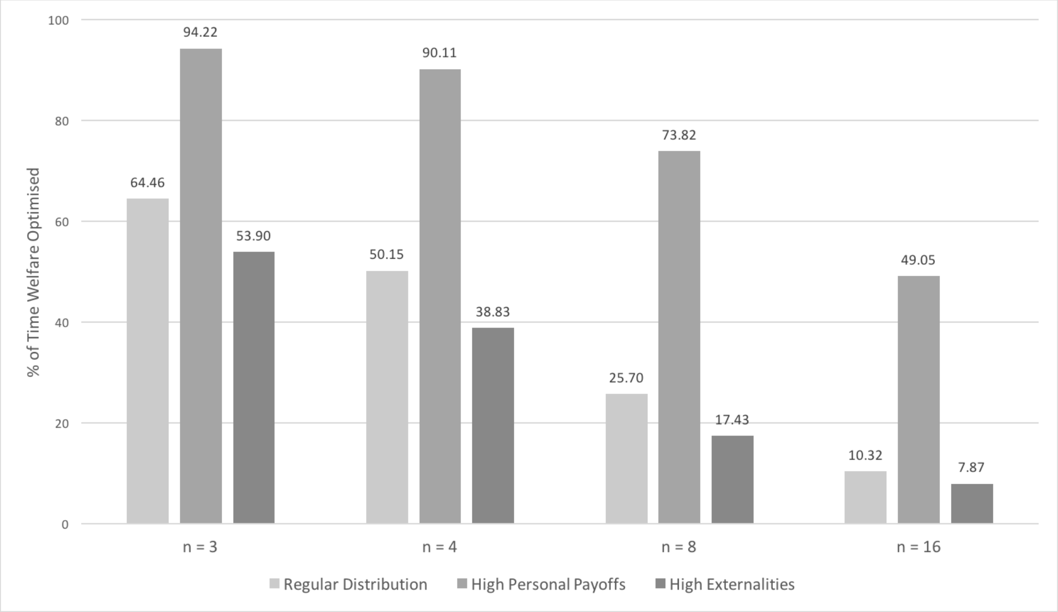
The results imply that for a small number of participating firms, the standard sealed bid auctions perform fairly well in terms of efficiency. However, the more firms participate in the auctions, the more often the goods are not efficiently allocated. The reason for this is that, in the standard sealed bid auctions, t_i^i accounts for half of a firm's valuation regardless of n. $\frac{1}{n-1}\sum_{j\neq i} t_j^i$ accounts for the other half. As stated, $\frac{1}{n-1}\sum_{j\neq i} t_j^i$ converges to 0.5 as n increases and thus, for high n, a firm's valuation is almost entirely dependent on t_i^i. This is in contrast to welfare. As t_i^i makes up only $\frac{1}{n}$ of w_i: the larger n, the less important t_i^i. This means that, in the limit, the allocation in the sealed bid auction will optimise welfare approximately as often as random allocation. The Vickrey auction, of course, always allocates the good efficiently.

Table 10 shows the average welfare realised by each auction. The expected level of welfare if goods were randomly allocated equals $\mathbb{E}[w_i] = \mathbb{E}[t_i^i] + \mathbb{E}[\sum_{j\neq i}^{n} t_i^j]$. As $\mathbb{E}[t_i^i] = 0.5$ and $\mathbb{E}[t_i^j] = -0.5$, ex-ante expected welfare is defined by: $\mathbb{E}[w_i] = 0.5 - (n-1)0.5 = 1 - 0.5n$.

Table 10: Average Realised Welfare

	Random Allocation	Standard Sealed Bid Auctions	Vickrey Auction
$n = 3$	-0.5	-0.197	-0.074
$n = 4$	-1	-0.653	-0.403
$n = 8$	-3	-2.574	-1.839
$n = 16$	-7	-6.561	-4.890

Figure 1: % of Times Welfare Optimised for Different Parameter Distributions



The results from table 10 are similar to those of table 9. The standard sealed bid auctions perform more like the Vickrey auction than random allocation for a small number of participants. Reversely, the standard sealed bid auctions perform closer to random allocation for larger n.

It is interesting to see how the difference between the interval of t_i^i and t_j^i influences the percentage of times the good is allocated such that it optimises welfare in the standard sealed bid auctions. Figure 1 shows what happens to the percentage of times welfare is optimised when the distributions of t_i^i and t_j^i are changed. The 'High Personal Payoffs' bar presents the results for when t_i^i is distributed on interval [0,10]. The 'High Externalities' bar shows what happens when t_j^i is distributed on interval [-10,0].[2]

The 'High Personal Payoffs' bar confirms that the standard sealed bid auctions indeed perform much better in terms of efficiency when the relative size of the firms' payoffs of getting the good themselves (t_i^i) are high compared to that of the externalities than when they are equal. This effect is present even for larger n. The reason for the better

---

[2] For all n, 10 million iterations are done.

performance is that as the difference between t_i^i and $\sum_{j \neq i}^n t_i^j$ increases, the more important t_i^i is for the realised welfare in case firm i wins. There is thus more congruence between the driver of a firm's valuation and the driver of welfare.

The 'High Externalities' bar shows what happens if externalities are high compared to the firms' payoffs of receiving the good themselves. Now, a firm's valuation is based mostly on the average externality a firm incurs from others winning ($-\frac{1}{n-1}\sum_{j \neq i}^n t_j^i$). For that reason, the winner generally is the firm that has the highest value of $-\frac{1}{n-1}\sum_{j \neq i}^n t_j^i$. While this value is correlated with welfare, it does not directly influence it like t_i^i does. Welfare is actually based on the externality the winning firm causes to others ($\sum_{j \neq i}^n t_i^j$). Therefore, if externalities are high compared to the payoffs of firms getting the good themselves, the standard sealed bid auctions do not perform well.

6.2 Revenue

Figure 2 shows how the average revenue of the second price sealed bid auction[3] and the Vickrey auction develops as the number of participating firms increases.[4] For both auctions, revenue increases as n increases.

For the standard sealed bid auctions, this result is driven by the increased likelihood of a very high valuation (a firm with a high value for t_i^i and a (reasonably) high value of $\frac{1}{n-1}\sum_{j \neq i}^n t_i^j$). This influences the revenue both through the increased expected value of the highest bid and the higher expected value of the second highest bid. Note that the revenue in the standard sealed bid auction is increasing and concave. The concavity is caused by the fact that the increased likelihood of high values of t_i^i is to some degree offset by $\frac{1}{n-1}\sum_{j \neq i}^n t_i^j$ converging to 0.5.

The increase in revenue from the Vickrey auction is caused by the the fact that the more firms are in the auction, the more firms will generally influence the outcome by participating and thus pay price p^i. What is striking is that while the Vickrey auction does not have a very high average level of revenue for lower n, it increases steadily as n increases. The Vickrey auctions' average revenue even surpasses the average revenue of the standard sealed bid auctions when $n = 12$. It thus seems like the Vickrey auction's low revenue is a problem only for low n.

---

[3] The reason that the first price sealed bid auction is not included is that its revenue is (on average) equivalent to the second price sealed bid auction, but takes more time to calculate.

[4] For $3 \leq n \leq 9$, 10 million iterations are done. For $10 \leq n \leq 16$, 1 million iterations are done.

Figure 2: Revenue Development as n Increases

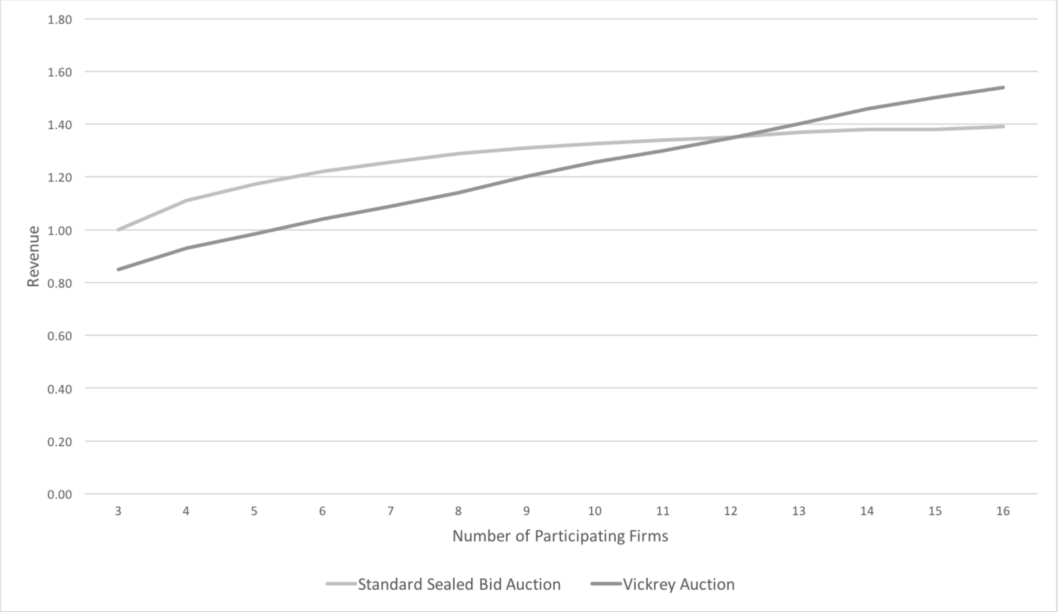


Table 11 presents the variance of the revenue for all auctions when $n = 3$ and for the second price sealed bid auction and Vickrey auction when $n = 4$, $n = 8$ and $n = 16$.[5] For the calculation of the firms' bidding strategies in the first price sealed bid auction for $n = 3$, please see Appendix C 'Distribution of b^i'.

Table 11: Variance of Revenue

	First Price Sealed Bid Auction	**Second Price Sealed Bid Auction**	**Vickrey Auction**
$n = 3$	0.007	0.021	0.053
$n = 4$		0.011	0.055
$n = 8$		0.0025	0.073
$n = 16$		0.0005	0.107

The results confirm that the first price sealed bid auction indeed has lower variance than the second price sealed bid auction. The reason for this is explained in the last paragraph of section 5.3. The Vickrey auction's variance is highest. What is interesting is that while the variance of revenue for the second price sealed bid auction decreases as n increases, the exact opposite holds for the Vickrey auction. The convergence of $\frac{1}{n-1}\sum_{j\neq i}^{n} t_i^j$ to 0.5 causes the decrease in variance for the second price sealed bid auction. The increase in the potential number of firms that influence the auction causes the increase in variance of the Vickrey auction.

---

[5] For all n, 10 million iterations are done.

7 Market Dynamics

While the assumption of uniformly distributed parameters is useful for providing insight into how the auctions work and perform in a general sense, it is not necessarily realistic. Generally, the firms' payoffs of winning or losing depend on the underlying model of competition. This section covers an analysis for when this is indeed the case.

The model considered is a Hotelling model with exogenous (randomly distributed) locations in different markets. Firms bid for a license that expands their location from a specific point to an interval. Their payoff of winning and losing the license depends on the proximity of the other firms in the market.

This model upholds the idea of allocative externalities: each firm incurs a unique non-positive externality of any firm j winning the auction and gets a positive payoff of winning the auction itself. It adds to the analysis by showing how market dynamics influence efficiency and revenue properties of the auctions considered in the paper.

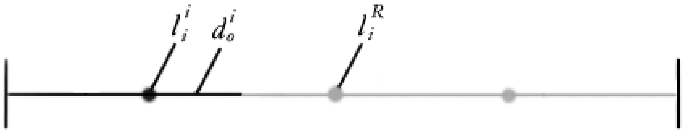
In contrast to when payoffs are just randomly distributed variables, there is now a correlation between t_i^i and t_j^i. This correlation may lead to the standard sealed bid auctions being more or less efficient and influence revenue in both the Vickrey and standard sealed bid auctions. In short, this analysis will show whether market dynamics matter when it comes to efficiency and revenue.

7.1 The Model

There are n firms and n markets à la Hotelling. Firms are indexed by $i = 1, 2, \ldots, n$, markets are indexed by $m = 1, 2, \ldots, n$. Each firm i operates in a single market $m = i$. Each firm has $n - 1$ allies that operate in markets $m \neq i$. The auction participants do not care about, or have any information on, markets other than their own. The price in all markets is 1 and there are no marginal costs of production. Profit is thus equal to demand. In a market, consumers are uniformly distributed on interval $[0, 1]$, with mass 1. Consumers buy from the firm that is closest to their location. The firms and their allies do not choose their location, it is given by independent randomly distributed variables on interval $[0, 1]$. Firm i's (or its ally's) location in market m is denoted by l_m^i.

Demand for firm i in market i is defined by the following function, where l_i^L and l_i^R denote the location of firm i's left and right neighbour respectively:

Figure 3: Visual Representation of Market



$$d_0^i = \begin{cases} l_i^i + \frac{l_i^R - l_i^i}{2} & \text{if } l_i^i = min\ \{l_i^1, ..., l_i^n\} \\ \frac{l_i^i - l_i^L}{2} + (1 - l_i^i) & \text{if } l_i^i = max\ \{l_i^1, ..., l_i^n\} \\ \frac{l_i^i - l_i^L}{2} + \frac{l_i^R - l_i^i}{2} & \text{otherwise} \end{cases}$$

Figure 3 provides a visual representation of firm i's location in market i and its corresponding demand.

A license of size z is auctioned that expands the location of the winner and its allies from point l_m^i to interval $[l_m^{iL},\ l_m^{iR}]$, where l_m^{iL} and l_m^{iR} are defined as follows:

$$l_m^{iL} = max\{l_m^i - z, 0\}$$
$$l_m^{iR} = min\{l_m^i + z, 1\}$$

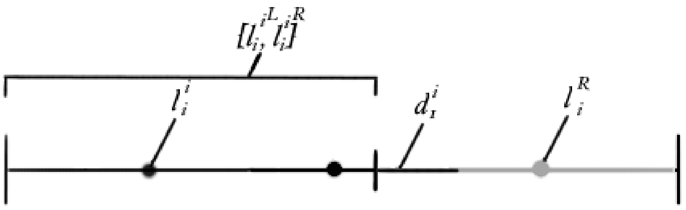
Allies can thus be viewed as firms that have a license sharing agreement with the firms participating in the auction. A real life example would be Google and Samsung's *"broad agreement to cross-license a range of each other's patents"* (Pfanner, 2014).

Demand for firm i in market i after winning the license is formally defined by:

$$d_1^i = \begin{cases} 1 & \text{if } l_i^{iL} < min\ \{l_i^1, ..., l_i^n\}\ \&\ l_i^{iR} > max\ \{l_i^1, ..., l_i^n\} \\ \frac{l_i^R - l_i^{iR}}{2} + l_i^{iR} & \text{if } l_i^{iL} < min\ \{l_i^1, ..., l_i^n\}\ \&\ l_i^{iR} < max\ \{l_i^1, ..., l_i^n\} \\ \frac{l_i^{iL} - l_i^L}{2} + (1 - l_i^{iL}) & \text{if } l_i^{iR} > max\ \{l_i^1, ..., l_i^n\}\ \&\ l_i^{iL} > min\ \{l_i^1, ..., l_i^n\} \\ \frac{l_i^{iL} - l_i^L}{2} + \frac{l_i^R - l_i^{iR}}{2} + (l_i^{iR} - l_i^{iL}) & \text{otherwise} \end{cases}$$

If firm i wins, it will always absorb some of the demand of its neighbours, as its location grows closer to them. Moreover, if the original location of a neighbour falls within the $[l_m^{iL},\ l_m^{iR}]$ interval, firm i will absorb all of that neighbour's demand. In this scenario, firm i may get a new neighbour and the value of l_i^L and/or l_i^R may thus change. d_1^i if firm i does not win the license is calculated the same way as d_0^i. However, an ally of the winning

Figure 4: Visual Representation of Market



firm may become the new neighbour or absorb firm i in its entirety. Figure 4 provides a visual representation of how L_i^R may change if firm i wins a license big enough so that its original neighbour's location in market i falls within firm i's location interval.

A firm's personal payoff of winning the auction is defined by $t_i^i = [d_1^i | i\ wins] - d_0^i$. The externality of another firm winning is defined by $t_j^i = [d_1^i | j\ wins] - d_0^i$. It is assumed that it is impossible for firms' allies to incentivise firms to bid higher. Because of the fact that firms do not care about other markets, the valuation is calculated as before: $v^i = t_i^i - \frac{1}{n-1} \sum_{j \neq i}^{n} t_j^i$.

7.2 Theoretical Analysis

7.2.1 Vickrey Auction

The Vickrey auction will of course remain efficient, regardless of the market model underlying it. However, the market model does have an effect on the auction's revenue. As stated in section 4.3, revenue in the Vickrey auction is highest when personal payoffs are large and externalities are strongly heterogeneous between firms. In the model, however, these two are mutually exclusive to some degree.

Personal payoff t_i^i is given by the possible gain of winning the auction. Its maximum value is defined by $t_i^i = 1 - d_0^i$. This means that, the larger d_0^i, the smaller the maximum value of t_i^i. For a high value of t_i^i to be possible, d_0^i thus has to be low. However, to attain the maximum value of t_i^i in this scenario, firm i should be able to absorb the entirety of the demand in its market. This is only possible if all other firms in the market are within distance z of firm i's original location. If that is indeed the case, all t_j^i will be exactly the same and revenue will thus not be particularly high.

It is possible that there are some firms located very far from firm i so that t_i^i is high, while not all externalities are the same. Even in this scenario, externalities will generally not differ by much. The reason for this is that the maximum value of t_j^i equals $-d_0^i$ and because d_0^i has to be low for a high value of t_i^i to be possible, it is not likely to differ strongly from zero.

While high revenue is thus unlikely in the Vickrey auction, it is not impossible. Consider the following example when $n = 3$ and $z = 0.5$ to illustrate.

Market 1: $l_1^1 = 0.48$, $l_1^2 = 0.50$, $l_1^3 = 1$

Market 2: $l_2^1 = 1$, $l_2^2 = 0.46$, $l_2^3 = 0.50$

Market 3: $l_3^1 = 0.50$, $l_3^2 = 1$, $l_3^3 = 0.46$

Current Demand: $d_0^1 = 0.49$, $d_0^2 = 0.48$, $d_0^3 = 0.48$.

Demand if firm 1 wins: $d_1^1 = 0.99$, $d_1^2 = 0.48$, $d_1^3 = 0.48$.

Demand if firm 2 wins: $d_1^1 = 0$, $d_1^2 = 0.98$, $d_1^3 = 0$.

Demand if firm 3 wins: $d_1^1 = 0.49$, $d_1^2 = 0$, $d^3 = 0.98$.

The payoff scheme in table 12 is then realised.

Table 12: Example Values Market Model

	1	2	3
1	0.5	−0.49	0.0
2	0.0	0.5	−0.48
3	−0.48	0.0	0.5
$\sum$	0.02	0.01	0.01

In this case, $p^1 = 0.5 + 0.48 = 0.98$, $p^2 = 0.48$ and $p^3 = 0$, which means $\Pi = 1.46$. As will be shown in section 7.2.2, the maximum revenue of the standard sealed bid auction is only 1. The maximum possible revenue increases as the license size increases. The reason for this is that the higher the license size, the larger the possible difference between externalities incurred by a firm. However, at the same time, it becomes more and more unlikely that the locations are such that a situation as the one above arises.

7.2.2 Standard Sealed Bid Auctions

As stated, the maximum values of t_i^i and t_j^i are defined by $1 - d_0^i$ and $-d_0^i$, respectively. Because $0 < d_0^i < 1$, revenue in the standard sealed bid auctions will never exceed 1. While the maximum level of revenue in the standard sealed bid auctions is thus lower than that of the Vickrey auction, the frequency at which maximum revenue is attained is higher. Any time a firm can absorb its market's entire demand, $t_i^i = 1 - d_0^i$ and $t_j^i = -d_0^i$ for all j. Regardless of firm i's original demand, $v^i = (1 - d_0^i) - \frac{1}{n-1} \sum_{j \neq i}^{n} -d_0^i = (1 - d_0^i) + d_0^i = 1$. As the license size increases, this valuation becomes more likely. Average revenue in the standard sealed bid auctions will thus approach 1.

The market model also influences the efficiency properties of the standard sealed bid auctions. Misallocation happens most frequently when a 'neutral' firm, that does not cause an externality to at least one other firm in the auction, is outbid by a second firm that causes larger externalities or causes externalities to more firms. The reason for this firm outbidding the 'neutral' firm may be that the firm incurs a disproportionally large externality from a third firm winning. If that third firm's valuation is low, both the 'neutral' firm and the second firm may benefit from the second firm decreasing its bid so that the 'neutral' firm wins. However, as the second firm has no information on which firm is most likely to win, it will not do so.

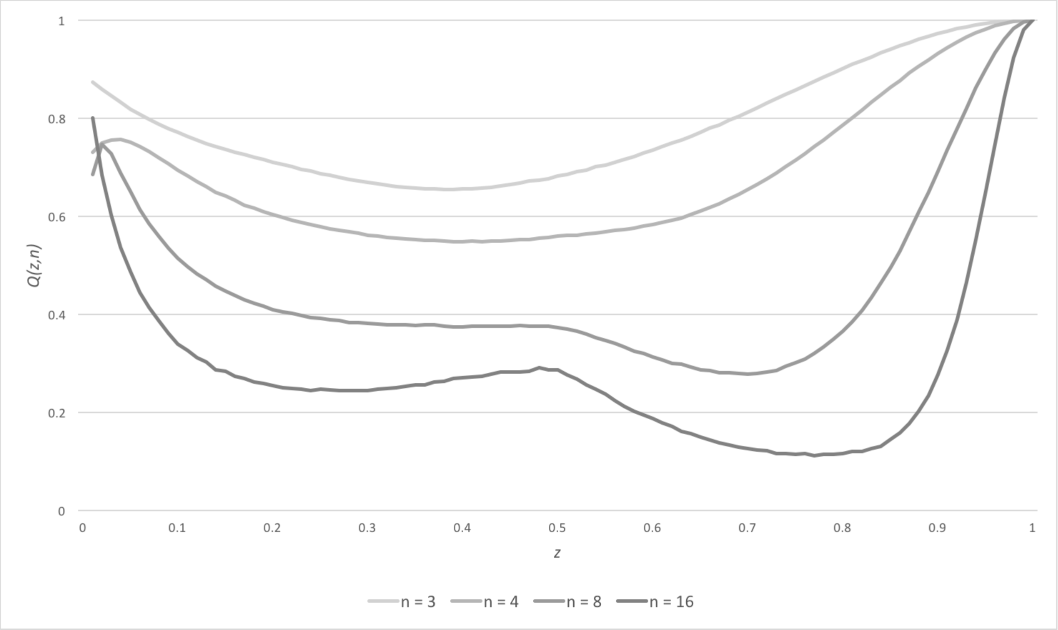
When the license size is close to 0, there is only a small probability that a firm's demand can be absorbed by another firm's ally. If this does indeed not happen, t^i_i is the same for all firms. Now, consider firm a and firm b. $v^a = t^a_a - t^a_b - \sum_{j \neq a,b}^{n} t^a_j$ and $w_b = t^b_b + t^a_b + \sum_{j \neq a,b}^{n} t^j_b$. t^a_a and t^b_b are irrelevant because they are equal. Now, as $\frac{\partial v^a}{\partial t^a_b} < 0$ and $\frac{\partial w_b}{\partial t^a_b} > 0$: if firm a incurs an externality from firm b winning, firm a's valuation increases, while welfare if firm b wins decreases. Therefore, a firm that incurs a multitude of externalities is both more likely to win and more likely to optimise welfare if it does. If only one firm incurs a large externality, that firm will most likely outbid all other firms and unless it causes small externalities to a large number of other firms, welfare will then be optimised. It is thus reasonably likely to be optimised in the standard sealed bid auctions when the auctioned license size is small.

As the license size increases, the probability that all firms will absorb the entire demand of their markets by winning also becomes larger. If all firms indeed do so, they will all value the license equally (at $v^i = (1 - d^i_0) + d^i_0 = 1$). Welfare if firm i wins will then be $w_i = (1 - d^i_0) + \sum_{j \neq i}^{n} -d^j_0 = 1 - d^1_0 - ... - d^n_0$ and is thus the same for all firms. Therefore, as z approaches 1, standard sealed bid auctions will almost always optimise welfare as well.

7.3 Monte Carlo Simulations

To see how the market model influences the auctions in (simulated) practice, some other Monte Carlo simulations are run. For the code used to run these simulations, please see Appendix B 'Python Code Hotelling Model'. In the simulation, the original locations of the firms are uniformly distributed on interval $[0, 1]$. The license size increases by 0.01

Figure 5: Function $Q(z,n)$ for standard sealed bid auctions when $0.01 \leq z \leq 1$ and $n \in \{3,4,8,16\}$



every hundredth of the total iterations done. For $n = 3$ and $n = 4$, 100 million iterations are done. For $n = 8$ and $n = 16$, respectively, 25 million and 10 million iterations are done.

The probability that welfare is optimised given the license size and number of firms participating in an auction is defined by function $Q(z,n)$. Figure 5 shows how the function develops as the license size increases for different numbers of firms in the standard sealed bid auctions. The graph shows that the standard sealed bid auctions indeed perform rather well in terms of efficiency when the license size z is close to either 0 or 1. This is especially important for larger n, where random allocation would only optimise welfare sporadically.

However, the figure also shows a number of unexpected results. Firstly, $Q(0.01,16) > Q(0.01,8)$, $Q(0.01,4)$. This is surprising because $Q(z,16)$ is lowest for any $z > 0.01$, except for when $z = 1$. This high likelihood of welfare optimisation is most likely caused by the probability that exactly one firm can absorb another firm's ally's demand (and vice versa) is higher for $n = 16$ than for $n = 8$ or $n = 4$, while the probability of two or more firms being able to do so is very low for any number of participating firms. This would also explain the small rise in $Q(z,4)$ when $0.01 \leq z \leq 0.04$ and $Q(z,8)$ when $0.01 \leq z \leq 0.02$.

Another notable observation is that, after a decrease in $Q(z,8)$ and $Q(z,16)$ as z increases, $Q(z,8)$ increases when $0.4 \leq z \leq 0.48$ and the same goes for $Q(z,16)$ when $0.3 \leq z \leq 0.48$. The cause of this is most likely that the difference in valuations between firms for $n = 8$ and $n = 16$ is largest for these license sizes. As shown in section 5.2, a

Figure 6: Revenue by License Size $n = 3$

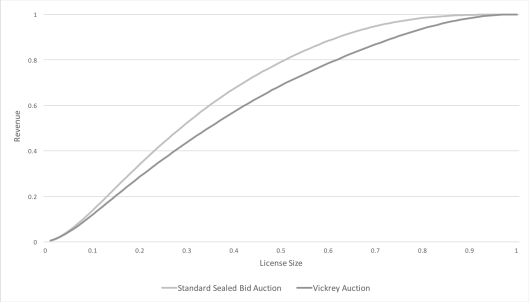


Figure 7: Revenue by License Size $n = 4$

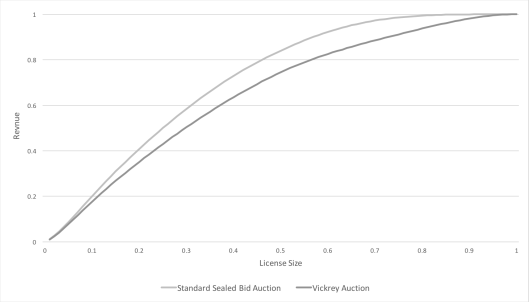


Figure 8: Revenue by License Size $n = 8$

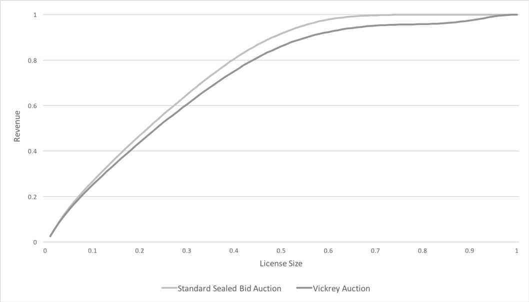
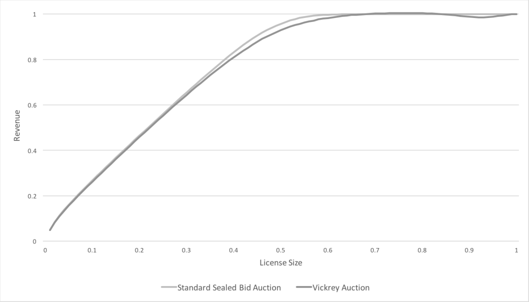


Figure 9: Revenue by License Size for $n = 16$



larger difference between the winner's valuation and that of the other firms means that welfare is more likely to be optimised.

Figures 6 through 9 show how revenue for the Vickrey auction and standard sealed bid auctions[6] develops as the auctioned license size increases for different numbers of participating firms. Clearly, the average revenue given a license size increases as the number of firms participating in the auction increases for both the Vickrey auction and the standard sealed bid auctions.

The standard sealed bid auctions perform better than the Vickrey auction in terms of revenue in almost all situations. However, as the number of participating firms increases, the revenue of the Vickrey auction does get closer. When $n = 16$, the Vickrey auction even outperforms the standard sealed bid auction in terms of revenue when $0.68 \leq z \leq 0.83$, where its revenue actually exceeds 1. As stated, it is very well possible for a single Vickrey auction's revenue to exceed 1 even for lower numbers of participating firms and smaller license sizes. It just so happens to be that for this small interval, the frequency of this happening and the average magnitude of it outweigh the times the Vickrey auction fails to yield a revenue of 1.

---

[6] The second price sealed bid auction is again considered in simulation as revenue is approximately equivalent to the first price sealed bid auction, but takes less time to calculate.

8 Conclusion & Discussion

The results from this paper show that for a small number of participants, the standard sealed bid auctions generally perform fairly well in terms of efficiency and yield higher revenue than the Vickrey auction. In general, the standard sealed bid auctions shine especially when the firms' payoffs of getting the good themselves are much higher than the externalities they incur of others winning. The standard sealed bid auctions perform worse when the opposite is true: the externalities incurred of others winning are high compared to the firms' personal payoffs.

For a low number of auction participants, the Vickrey auction's revenue is often much lower than it is in the standard sealed bid auctions. Generally, the Vickrey auction should thus be used only when efficiency is the main goal or when the number of participating firms is high. Only when externalities are divided such that firms care a lot about one firm winning and not at all about another could it be wise to choose the Vickrey auction over a standard sealed bid auction for revenue in other cases.

For a large number of participants, the Vickrey auction quickly gains an edge over the standard sealed bid auctions. Where the standard sealed bid auctions start failing to allocate the good efficiently more and more often, the Vickrey auction (of course) does it without fail. Moreover, while both the standard sealed bid auctions' and the Vickrey auction's revenue increases as the number of participating firms does, this effect is much stronger for the Vickrey auction. The reason for this difference is that externalities affect the auctions' revenue differently. In the standard sealed bid auctions, firms incorporate externalities in their valuation through a weighted function of the sum of the externalities. The average value of that function will not change as the number of firms increases, because the weight placed on one externality decreases proportionally. In the Vickrey auction, the effect of externalities is very different. A firm communicates the unweighted externality to the auctioneer and may have to pay the entirety of that externality if it is necessary to change the auction's outcome. At a low number of participating firms, the probability of this happening is small. However, as the number of firms participating in the auction increases, the probability of a number of firms having to pay the auctioneer the (full) amount (or part) of their externality also increases. This difference holds true regardless of whether the externalities are negative or positive.

Despite the attractive features of the Vickrey auction, one should be wary of its weaknesses. In the case of allocative externalities, the sensitivity to collusion is especially worrisome. Moreover, the Vickrey auction's revenue generally has a higher variance than that of the standard sealed bid auctions.

In the context of the market model discussed, the standard sealed bid auctions perform fairly well in terms of efficiency when the license size is small or large. On top of that, the standard sealed bid auctions almost always perform better than the Vickrey auction when it comes to revenue, even for larger numbers of participants. The reason for this is the fact that high personal payoffs and large externalities are mutually exclusive to some degree. In the standard sealed bid auctions, the auctioneer's revenue depends on only one firm; the firm with the highest valuation. While smaller externalities do decrease a firm's valuation, it is only (a small) part of the function. The Vickrey auction, however, depends mostly on externalities for its high revenue. As large externalities become rarer, the Vickrey auction's revenue decreases rapidly.

What these results show is that an auction's underlying market model combined with what is being sold may be of great influence on an auction's performance and it is thus paramount that this is taken into account when evaluating any auction.

All in all, not one of the auctions discussed in this paper is perfect for heterogeneous allocative externalities. If strong rivalries exist, an open ascending bid auction may be a good alternative. In such an auction, firms have more information on who is most likely to win. Therefore, it could be that rivalries that lead to inefficiency in the standard sealed bid auctions do not do so in the English auction. However, the root cause of the inefficiency found in the standard sealed bid auctions still remains: the auctioneer does not have perfect information on the externalities a specific firm causes. Therefore, there seems to be no way to attain efficiency without allowing firms to communicate the externalities they incur for each competitor winning. Finding a mechanism that incorporates this, but avoids the collusion pitfall may be a worthwhile avenue of future research. Another option is to do empirical research on bidder behaviour. This could provide additional insights in the effect of, for example, risk aversion or the complexity of the auction. A third option would be to further explore the effect of market dynamics by extending the market model used in this paper. For example, by replacing z with z^m, so that the license size may differ per market or by having allies not benefit from the license fully, but just in part.

References

Acemoglu, D. and Ozdaglar, A. (2009). Networks lectures 19-21: Incomplete information: Bayesian nash equilibria, auctions and introduction to social learning.

Ausubel, L. M. et al. (1999). A generalized vickrey auction. *Econometrica*.

Ausubel, L. M., Milgrom, P., et al. (2006). The lovely but lonely vickrey auction. *Combinatorial auctions*, 17:22–26.

Clarke, E. H. (1971). Multipart pricing of public goods. *Public choice*, 11(1):17–33.

Das Varma, G. (2002). Standard auctions with identity-dependent externalities. *RAND Journal of Economics*, pages 689–708.

Google (2017). About the ad auction. Retrieved 30 June 2017, from `https://support.google.com/adsense/answer/160525`.

Groves, T. (1973). Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631.

Haouideg, H. (2016). Patent auctions. Retrieved 30 June 2017, from `https://www.patentauction.com/`.

Jehiel, P. and Moldovanu, B. (2005). Allocative and informational externalities in auctions and related mechanisms.

Jehiel, P., Moldovanu, B., and Stacchetti, E. (1999). Multidimensional mechanism design for auctions with externalities. *Journal of economic theory*, 85(2):258–293.

MacKie-Mason, J. K. and Varian, H. R. (1994). Generalized vickrey auctions.

McAfee, R. P. and McMillan, J. (1987). Auctions and bidding. *Journal of economic literature*, 25(2):699–738.

Morgan, J. (2001). Efficiency in auctions: Theory and practice. *Journal of International Money and Finance*, 20(6):809–838.

Parkes, D. C. and Ungar, L. H. (2001). *Iterative combinatorial auctions: Achieving economic and computational efficiency*. University of Pennsylvania.

Pfanner, E. (2014). Google and samsung sign broad cross-licensing agreement. *The New York Times*.

Robinson, M. S. (1985). Collusion and the choice of auction. *The RAND Journal of Economics*, pages 141–145.

Vickrey, W. (1961). Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16(1):8–37.

Appendices

A. Python Code Uniform Parameter Distribution

```python
#Author: Bart de Koning
#This code can be found online at http://bkdekoning.com/mcg.txt
import random
import time
from scipy.integrate import quad
from fractions import Fraction

x = 0
a = 0 #Lower bound t^i_i
b = 1 #Upper bound t^i_i
c = -1 #Lower bound t^i_j
d = 0 #Upper bound t^i_j
n = 3
y = 1000 #Number of iterations
wins0 = 0
optimise0 = 0
totrealisedwelfarenonorm = 0
totvaluation = 0
totsecondvaluation = 0
totfirstpricerevenue = 0
totmaxwelfarenonorm = 0
totvcgrevenue = 0
welfareoptimiser0 = 0
interimvarfpsba = 0
interimvarspsba = 0
interimvarvcg = 0
maxvcg = 0
maxfpsba = 0
maxspsba = 0
#Uncomment below lines if you want to input as user.
#a = input('What would you like the lower bound of personal payoff to be? ')
#b = input('What would you like the upper bound of personal payoff to be? ')
#c = input('What would you like the lower bound of externality to be? ')
#d = input('What would you like the upper bound of externality to be? ')
n = input('How many firms are in te auction? ')
y = input('How many iterations would you like to run? ')
fracvar = Fraction(1,y-1)
def genRandSelf(): #Will generate payoff for firm if it wins the good itself.
        randself = random.uniform(a,b) #Change this if you want to change the distribution of t^i_i
        return randself
def genRandOther():
        randother = random.uniform(c,d) #Will generate payoff for firm if another firm wins.
        return randother #Change this if you want to change the distribution of t^i_j
start_time = time.time()
while x < y:
        i = 0
        payoff = {}
        while i < n: #Number of firms in the game.
                j = 0
                while j < n:
                        keypayoff = int(str(i)+str(00)+str(j)) #Create key for payoff, starts at t^i_0: payoff firm i when 0 wins then goes up to t^i_(n-1) - Note that firm 0 is also a firm.
                        if j == i:
                                valuepayoff = genRandSelf() #Generate payoff if the firm itself gets the good (t^i_i).
                        if j != i:
                                valuepayoff = genRandOther() #Generate payoff if firm j gets the good (t^i_j).
                        j += 1
                        payoff[keypayoff]=valuepayoff #Assign payoff values to the payoff dictionary.
```

```
            i += 1
        i = 0
        valuation = {}
        while i < n:
            keyvaluation = i
            j = 0
            valuevaluation = 0
            while j < n:
                if j == i:
                    valuevaluation += payoff[int(str(i)+str(00)+str(j))] #Add payoff if firm i wins itself to valuation. The str(00) is to avoid problems if n>10.
                if j !=i:
                    valuevaluation -= payoff[int(str(i)+str(00)+str(j))]/(n-1) #Add 1/n-1 times the payoff firm i gets when firm j wins to the valuation.
                j += 1
            valuation[keyvaluation]=valuevaluation #This gives valuation of a firm in the sealed bid auction.
            i += 1
        i = 0
        welfare = {}
        while i < n:
            keywelfare = i
            j = 0
            valuewelfare = 0
            while j < n:
                valuewelfare += payoff[int(str(j)+str(00)+str(i))] #Calculate total welfare if firm i wins.
                j += 1
            welfare[keywelfare]=valuewelfare
            i += 1
        #Firm 0 will be considered, it does not matter because every firm is symmetric.
        winner = max(valuation, key=valuation.get)
        welfareoptimiser = max(welfare, key=welfare.get)
        if winner == 0:
            wins0 += 1
            totrealisedwelfarenonorm += welfare[0]
            totmaxwelfarenonorm += welfare[welfareoptimiser]
            if winner == welfareoptimiser:
                optimise0 += 1 #Shows how often the winner optimises welfare in first or second price sealed bid auction.
            totvaluation += valuation[0] #This variable will show te sum of all firm 0's valuations so that the sealed bid auctions can be compared to this in terms of revenue.
            j = 0
            secondmaxval = 0
            while j < (n-1):
                if valuation[j+1] > secondmaxval:
                    secondmaxval = valuation[j+1] #Find the second highest valuation (revenue for auctioneer in second price sealed bid auction)
                j += 1
                if secondmaxval > maxspsba:
                    maxspsba = secondmaxval
            totsecondvaluation += secondmaxval
            if a == 0 and b == 1 and c == -1 and d == 0 and n == 3:
                varspsba = (secondmaxval-1)**2
            if a == 0 and b == 1 and c == -1 and d == 0 and n == 4:
                varspsba = (secondmaxval-1.10684507763)**2
            if a == 0 and b == 1 and c == -1 and d == 0 and n == 8:
                varspsba = (secondmaxval-1.28617911859)**2
            if a == 0 and b == 1 and c == -1 and d == 0 and n == 16:
                varspsba = (secondmaxval-1.391406106)**2
            if a == 0 and b == 1 and c == -1 and d == 0:
                if n == 3 or n == 4 or n == 8 or n == 16:
                    interimvarspsba += varspsba
                    totvarspsba = fracvar*interimvarspsba
            if a == 0 and b == 1 and c == -1 and d == 0 and n == 3: #Because the first price sealed bid auction depends on the distribution of variables, this is only calculated for when parameters are
```

```
            distributed uniformly on range 0,1 and -0,1 - It is theoretically possible to do this for more distributions. My advice would be to first calculate the strategy then plug it in here.
            #Revenue in first price sealed bid auction is calculated here using the first price sealed bid auction symmetric bidding strategy described in the thesis.
            frac1 = Fraction(1,6)
            frac2 = Fraction(13,3)
            if valuation[0] == 0:
                firstpricebid = 0
            if valuation[0] > 0:
                def integrand0(y):
                    return y*(n-1)*(((2)*y**3)/3)**(n-2)*(2*y**2)
                integrandresult0, err = quad(integrand0, 0, min(0.5, valuation[0]))
            if valuation[0] > 0.5:
                def integrand1(y):
                    return y*(n-1)*((-(2)*y**3)/3+2*y**2-y+(frac1))**(n-2)*(-2*y**2+4*y-1)
                integrandresult1, err = quad(integrand1, 0.5, min(1.5, valuation[0]))
            if valuation[0] > 1.5:
                def integrand2(y):
                    return y*(n-1)*((2*y**3)/3-4*y**2+8*y-frac2)**(n-2)*(2*y**2-8*y+8)
                integrandresult2, err = quad(integrand2, 1.5, min(2, valuation[0]))
            if valuation[0] > 0 and valuation[0] < 0.5:
                nonintegrandresult = ((2*valuation[0]**3)/3)**(n-1)
                firstpricebid = integrandresult0/nonintegrandresult
            if valuation[0] > 0.5 and valuation[0] <= 1.5:
                nonintegrandresult = ((-(2)*valuation[0]**3)/3+2*valuation[0]**2-valuation[0]+(frac1))**(n-1)
                firstpricebid = (integrandresult0+integrandresult1)/nonintegrandresult
            if valuation[0] > 1.5 and valuation[0] <= 2:
                nonintegrandresult = ((2*valuation[0]**3)/3-4*valuation[0]**2+8*valuation[0]-(frac2))**(n-1)
                firstpricebid = (integrandresult0+integrandresult1+integrandresult2)/nonintegrandresult
            if firstpricebid > maxfpsba:
                maxfpsba = firstpricebid
            totfirstpricerevenue += firstpricebid
            varfpsba = (firstpricebid-1)**2
            interimvarfpsba += varfpsba
            totvarfpsba = fracvar*interimvarfpsba
    #Vickrey Analysis
    if welfareoptimiser == 0: #If welfareoptimiser == 0, firm 0 would win in the VCG auction.
        welfareoptimiser0 += 1
        currentsit = {}
        i = 0
        vcgrevenue = 0
        while i < n:
            keycurrentsit = i
            currentsit[keycurrentsit] = 0
            j = 0
            while j < n:
                if j != i:
                    payoffversus = payoff[int(str(j)+str(00)+str(0))]
                    currentsit[keycurrentsit] += payoffversus
                j += 1
            welfarenoi = {}
            j = 0
            while j < n:
                if j != i:
                    keywelfarenoi = j
                    valuewelfarenoi = 0
                    k = 0
                    while k < n:
```

```
                        if k != i:
                            valuewelfarenoi += payoff[int(str(k)+str(00)+str(j))]
                        k += 1
                    welfarenoi[keywelfarenoi]=valuewelfarenoi
                j += 1
            welfareoptimiserwithouti = max(welfarenoi, key=welfarenoi.get) #Find who welfare optimiser would have been, had firm 0 not existed
            vcgrevenue += max((welfarenoi[welfareoptimiserwithouti]-currentsit[keycurrentsit]),0) #Calculate VCG revenue, not necessary to go below 0 as long as t^i_i>0.
            i += 1
        if vcgrevenue > 0:
            totvcgrevenue += vcgrevenue #If it's not bigger than 0, auctioneer can choose to give the good away for free. Firm will want that for sure.
        if vcgrevenue > maxvcg:
            maxvcg = vcgrevenue
        if a == 0 and b == 1 and c == -1 and d == 0 and n == 3:
            varvcg = (vcgrevenue-0.8497)**2
        if a == 0 and b == 1 and c == -1 and d == 0 and n == 4:
            varvcg = (vcgrevenue-0.9223550392395)**2
        if a == 0 and b == 1 and c == -1 and d == 0 and n == 8:
            varvcg = (vcgrevenue-1.15388886611)**2
        if a == 0 and b == 1 and c == -1 and d == 0 and n == 16:
            varvcg = (vcgrevenue-1.52327541084)**2
        if a == 0 and b == 1 and c == -1 and d == 0:
            if n == 3 or n == 4 or n == 8 or n == 16:
                interimvarvcg += varvcg
                totvarvcg = fracvar*interimvarvcg
    x += 1
    if x % 10000 == 0:
        print x,"out of",y,"iterations done."
        print("--- %s seconds ---" % (time.time() - start_time))
numberfile = "results"+str(n)+".txt"
f = open(numberfile, 'a')
print "This analysis is done for n =",n,"and",x,"iterations."
f.write("Firms"+"        "+str(n))
f.write('\n')
f.write("Iterations"+"  "+str(x))
f.write('\n')
print "Firm 0 won",wins0,"times."
f.write("Sealed Bid Wins 0"+"   "+str(wins0))
f.write('\n')
print "In a first price- or second price sealed bid auction, a firm is expected to win", (float(wins0)/x)*100, "% of the time."
f.write("Expected Win Percentage Sealed Bid"+"  "+str((float(wins0)/x)*100))
f.write('\n')
print "If a firm wins in the first price- or second price sealed bid auction, welfare is optimised", (float(optimise0)/wins0)*100, "% of the time."
f.write("Sealed Bid Welfare Optimisation"+"     "+str((float(optimise0)/wins0)*100))
f.write('\n')
print "The maximum possible average welfare was", totmaxwelfarenonorm/wins0
f.write("Maximum Possible Average Welfare"+"    "+str(totmaxwelfarenonorm/wins0))
f.write('\n')
print "The average welfare realised is", totrealisedwelfarenonorm/wins0
f.write("Average Realised Welfare"+"    "+str(totrealisedwelfarenonorm/wins0))
f.write('\n')
print "Average revenue in the second price sealed bid auction is", (totsecondvaluation/wins0)
f.write("Average Revenue SPSBA"+"        "+str((totsecondvaluation/wins0)))
f.write('\n')
print "Average revenue as percentage of max valuation in second price sealed bid auction is", (totsecondvaluation/totvaluation)*100,"%."
f.write("Revenue SPSBA as % Max Valuation" + "  " +str((totsecondvaluation/totvaluation)*100))
f.write('\n')
print maxspsba,"was the maximum revenue in the second price sealed bid auction auction!"
f.write("Max Revenue SPSBA" + " " +str(maxspsba))
```

```
f.write('\n')
if a == 0 and b == 1 and c == -1 and d == 0 and n == 3:
        print "Average revenue in the first price sealed bid auction is", (totfirstpricerevenue/wins0)
        print "Average revenue as percentage of max valuation in first price sealed bid auction is", (totfirstpricerevenue/totvaluation)*100,"%."
        print maxfpsba,"was the maximum revenue in the first price sealed bid auction auction!"
        f.write("Max Revenue FPSBA" + " " +str(maxfpsba))
        f.write('\n')
print maxvcg,"was the maximum revenue in the VCG auction!"
f.write("Max Revenue VCG" + "    " + str(maxvcg))
f.write('\n')
print "Average VCG revenue is",(totvcgrevenue/welfareoptimiser0)
f.write("Average Revenue VCG" + "        " +str(totvcgrevenue/welfareoptimiser0))
f.write('\n')
f.write('End of this simulation!')
f.write('\n')
numberfilevar = "var"+str(n)+".txt"
g = open(numberfilevar, 'a')
if a == 0 and b == 1 and c == -1 and d == 0:
        if n == 3:
                print "The variance in the first price sealed bid auction is", totvarfpsba
                g.write("Variance FPSBA" + "    " +str(totvarfpsba))
                g.write('\n')
        if n == 3 or n == 4 or n == 8 or n == 16:
                print "The variance in the second price sealed bid auction is", totvarspsba
                g.write("Variance SPSBA" + "    " +str(totvarspsba))
                g.write('\n')
                print "Variance in the VCG auction is", totvarvcg
                g.write("Variance VCG" + "      " +str(totvarvcg))
                g.write('\n')
g.write('End of this simulation!')
g.write('\n')
```

B. Python Code Hotelling Model

```python
#Author: Bart de Koning
#This code can be found online at http://bkdekoning.com/hot.txt
import random
from fractions import Fraction
import time

def genRand():
        rand = random.uniform(0,1)
        return rand

x = 0
y = 100000
n = 3
winnerwelfareoptimiser = 0
license = 0.5
totssbrevenue = 0
totvcgrevenue = 0
n = input('How many firms are in te auction? ')
y = input('How many iterations would you like to run? ')

start_time = time.time()
while x < y:
        if x % (0.01*y) == 0: #Increase license size every 0.01*yth iteration. Every license size from 0.01 to 1 will thus be simulated equally often.
                if license != 0:
                        print("--- %s seconds ---" % (time.time() - start_time))
                        print "For license size",license,",",float((100*float(winnerwelfareoptimiser))/(0.01*y)),"% of the time, the winner is the welfare optimiser."
                        print "Average revenue standard sealed bid auction is", totssbrevenue/(0.01*y),". Average revenue Vickrey auction is", totvcgrevenue/(0.01*y),"."
                        numberfile = "HotellingModel"+str(n)+".txt"
                        f = open(numberfile, 'a')
                        f.write( "License size" + "      " + str(license))
                        f.write('        ')
                        f.write( "% of time Winner is Welfare Optimiser" + "     " + str(float((100*float(winnerwelfareoptimiser))/(0.01*y))))
                        f.write('        ')
                        f.write( "Average Revenue SPSBA" + "     " + str(totssbrevenue/(0.01*y)))
                        f.write('        ')
                        f.write( "Average Revenue Vickrey" + "   " + str(totvcgrevenue/(0.01*y)))
                        f.write('\n')
                license += 0.01
                winnerwelfareoptimiser = 0
                totssbrevenue = 0
                totvcgrevenue = 0
                maxvcg = 0
        location = {}
        location[str(0)] = 0
        location[str(1)] = 1
        basepayoff = {}
        leftneighbour = {}
        rightneighbour = {}
        valuation = {}
        welfare = {}
        currentsit = {}
        i = 0
        maxval = 0
        secondmaxval = 0
        while i < n: #Number of firms in the game.
                j = 0
                while j < n:
                        valuelocation = genRand()
                        keylocation = (str(j)+str(00)+str(i)) #Location of firm j in market i
                        location[keylocation] = valuelocation
                        j += 1
```

```
        distanceleft = abs(location[str(i)+str(00)+str(i)]) #These variables will define demand for firm i
        distanceright = abs(location[str(i)+str(00)+str(i)] - 1)
        j = 0
        leftneighbour[i] = "None"
        rightneighbour[i] = "None"
        while j < n:
            diff = location[str(i)+str(00)+str(i)]-location[str(j)+str(00)+str(i)]
            if diff > 0 and abs(diff) < distanceleft:
                distanceleft = abs(diff) #This is the distance to the firm's left neighbour
                leftneighbour[i] = j
            if diff < 0 and abs(diff) < distanceright:
                distanceright = abs(diff)
                rightneighbour[i] = j
            j += 1
        j = 0
        basepayoff[i] = distanceleft/2+distanceright/2 #A firm gets at least 0.5*distanceleft+0.5*distranceright and then possible some more if it has no neighbours on one side
        if distanceleft == abs(location[str(i)+str(00)+str(i)]):
            basepayoff[i] += distanceleft/2
        if distanceright == abs(location[str(i)+str(00)+str(i)] - 1):
            basepayoff[i] += distanceright/2
        i += 1
    i = 0
    locationleft = {} #Will be used to define location of a firm after it has won the license
    locationright = {}
    payoff = {}
    while i < n:
        locationleft[str(i)+str(00)+str(i)] = max((location[str(i)+str(00)+str(i)]-license),0)
        locationright[str(i)+str(00)+str(i)] = min((location[str(i)+str(00)+str(i)]+license),1)
        distanceleft = abs(locationleft[str(i)+str(00)+str(i)])
        distanceright = abs(locationright[str(i)+str(00)+str(i)] - 1)
        j = 0
        while j < n:
            if location[str(j)+str(00)+str(i)] < locationleft[str(i)+str(00)+str(i)]: #Check if there are any neighbours
                diffleft = locationleft[str(i)+str(00)+str(i)] - location[str(j)+str(00)+str(i)]
                if abs(diffleft) < distanceleft:
                    distanceleft = abs(diffleft)
            if location[str(j)+str(00)+str(i)] > locationright[str(i)+str(00)+str(i)]:
                diffright = locationright[str(i)+str(00)+str(i)] - location[str(j)+str(00)+str(i)]
                if abs(diffright) < distanceright:
                    distanceright = abs(diffright)
            j += 1
        payoff[str(i)+str(00)+str(i)] = distanceleft/2+distanceright/2+(locationright[str(i)+str(00)+str(i)]-locationleft[str(i)+str(00)+str(i)]) #Payoff after license has been won
        valuation[i] = payoff[str(i)+str(00)+str(i)]-basepayoff[i]
        welfare[i] = payoff[str(i)+str(00)+str(i)]-basepayoff[i]
        i += 1
    i = 0
    while i < n:
        j = 0
        while j < n: #Find payoff if firm i does not win
            if abs(location[str(i)+str(00)+str(i)]-location[str(j)+str(00)+str(i)]) <= license and j != i:
                payoff[str(i)+str(00)+str(j)] = 0
            if abs(location[str(i)+str(00)+str(i)]-location[str(j)+str(00)+str(i)]) >= license :
                if leftneighbour[i] == j or rightneighbour[i] == j:
                    payoff[str(i)+str(00)+str(j)] = basepayoff[i]-license/2
            if abs(location[str(i)+str(00)+str(i)]-location[str(j)+str(00)+str(i)]) >= license and leftneighbour[i] != j and
```

```
                        rightneighbour[i] != j:
                payoff[str(i)+str(00)+str(j)] = basepayoff[i]
            if j != i:
                valuation[i] -= (payoff[str(i)+str(00)+str(j)]-basepayoff[i])/(n-1)
                welfare[j] += (payoff[str(i)+str(00)+str(j)]-basepayoff[i])
            j += 1
        i += 1
    i = 0
    while i < n: #Check what the winner has to pay in second price sealed bid auction
        if valuation[i] > secondmaxval:
            secondmaxval = valuation[i]
            if valuation[i] > maxval:
                secondmaxval = maxval
                maxval = valuation[i]
        i += 1
    totssbrevenue += secondmaxval
    i = 0
    winner = max(valuation, key=valuation.get)
    welfareoptimiser = max(welfare, key=welfare.get)
    if welfare[winner] <= welfare[welfareoptimiser]+0.000001 and welfare[winner] >= welfare[welfareoptimiser]-0.000001: #Account for inaccuracy calculation
        winnerwelfareoptimiser += 1
    vcgrevenue = 0
    while i < n: #Vickrey Revenue Analysis
        keycurrentsit = i
        currentsit[keycurrentsit] = 0
        j = 0
        while j < n:
            if j != i:
                payoffversus = payoff[str(str(j)+str(00)+str(welfareoptimiser))]
                currentsit[keycurrentsit] += payoffversus
            j += 1
        welfarenoi = {}
        j = 0
        while j < n:
            if j != i:
                keywelfarenoi = j
                valuewelfarenoi = 0
                k = 0
                while k < n:
                    if k != i:
                        valuewelfarenoi += payoff[str(str(k)+str(00)+str(j))]
                    k += 1
                welfarenoi[keywelfarenoi]=valuewelfarenoi
            j += 1
        welfareoptimiserwithouti = max(welfarenoi, key=welfarenoi.get)
        vcgrevenue += max((welfarenoi[welfareoptimiserwithouti]-currentsit[keycurrentsit]),0)
        i += 1
    totvcgrevenue += vcgrevenue
    x += 1
print "For license size",license,",",float((100*float(winnerwelfareoptimiser))/(0.01*y)),"% of the time, the winner is the welfare optimiser."
print "Average revenue standard sealed bid auction is",totssbrevenue/(0.01*y),". Average revenue Vickrey auction is", totvcgrevenue/(0.01*y),"."
numberfile = "HotellingModel"+str(n)+".txt"
f = open(numberfile, 'a')
f.write( "License size" + "      " + str(license))
f.write('        ')
f.write( "% of time Winner is Welfare Optimiser" + "     " + str(float((100*float(winnerwelfareoptimiser))/(0.01*y))))
f.write('        ')
f.write( "Average Revenue SPSBA" + "     " + str(totssbrevenue/(0.01*y)))
f.write('        ')
f.write( "Average Revenue Vickrey" + "  " + str(totvcgrevenue/(0.01*y)))
f.write('\n')
```

C. Distribution of b^i

Convolution will be used to find the distribution of b^i under the assumption of uniformly distributed parameters. In doing so, the distribution of t_i^i can be used in its original form. However, firms do not fully incorporate every externality into their valuation function. Therefore, $s_j^i = -\frac{1}{n-1} t_j^i$ is defined. The probability density functions of t_i^i and s_j^i are as follows.

$$f_{t_i^i}(t_i^i) = \begin{cases} 1 & \text{if } 0 \leq t_i^i \leq 1 \\ 0 & \text{otherwise} \end{cases} \qquad f_{s_j^i}(s_j^i) = \begin{cases} 2 & \text{if } 0 \leq s_j^i \leq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

From these distribution functions, it is possible to define the probability density function of the effect of the externalities (t_{j1}^i & t_{j2}^i) on firm i's valuation defined as $f_{e_v}(e_v)$.

$$f_{e_v}(e_v) = \int_{-\infty}^{\infty} f_{s_{j1}^i}(e_v - s_{j2}^i) f s_{j2}^i(s_{j2}^i) ds_{j2}^i$$

$f s_{j2}^i(s_{j2}^i) = 0$ unless $0 \leq s_{j2}^i \leq 0.5$ and then $f s_{j2}^i(s_{j2}^i) = 2$

$$f_{e_v}(e_v) = \int_{0}^{0.5} 2 f_{s_{j1}^i}(e_v - s_{j2}^i) ds_{j2}^i$$

$f_{s_{j1}^i}(e_v - s_{j2}^i) = 0$ unless $0 \leq e_v - s_{j2}^i \leq 0.5 \rightarrow e_v - 0.5 \leq s_{j2}^i \leq e_v$ and then $f_{s_{j1}^i}(e_v - s_{j2}^i) = 2$.

$$f_{e_v}(e_v) = \int_{0}^{e_v} 4 = 4e_v \text{ for } 0 \leq e_v \leq 0.5$$

$$f_{e_v}(e_v) = \int_{e_v - 0.5}^{0.5} 4 = 4 - 4e_v \text{ for } 0.5 \leq e_v \leq 1$$

$$f_{e_v}(e_v) = \begin{cases} 4e_v & \text{if } 0 \leq e_v \leq 0.5 \\ 4 - 4e_v & \text{if } 0.5 \leq e_v \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

Now, it is possible to get $f_{v^i}(v^i)$ through convolution of f_{e_v} and $f_{t_i^i}$.

$$f_{v^i}(v^i) = \int_{-\infty}^{\infty} f_{e_v}(v^i - t_i^i) f t_i^i(t_i^i) dt_i^i$$

$$f_{v^i}(v^i) = \int_0^1 f_{e_v}(v^i - t_i^i) dt_i^i$$

$f_{e_v}(v^i - t_i^i) = 0$ unless $0 \leq v^i - t_i^i \leq 1$, then $f_{e_v}(v^i - t_i^i) = 4(v^i - t_i^i)$ for $0 \leq v^i - t_i^i \leq 0.5 \rightarrow v^i - 0.5 \leq t_i^i \leq v^i$ and $f_{e_v}(v^i - t_i^i) = 4 - 4(v^i - t_i)$ for $0.5 \leq v^i - t_i^i \leq 1 \rightarrow v^i - 1 \leq t_i^i \leq v^i - 0.5$. This means that three cases arise.

Case 1: $0 \leq v^i \leq 0.5 \rightarrow 0 \leq t_i^i \leq v^i$ for $4(v^i - t_i^i)$

$$f_{v^i}(v^i) = \int_0^{v^i} 4(v^i - t_i^i) dt_i^i = 2{v^i}^2$$

Case 2.1: $0.5 \leq v^i \leq 1.5 \rightarrow v^i - 0.5 \leq t_i^i \leq v^i$ for $4(v^i - t_i^i)$

Case 2.2: $0.5 \leq v^i \leq 1.5 \rightarrow 0 \leq t_i^i \leq v^i - 0.5$ for $4 - 4(v^i - t_i^i)$

$$f_{v^i}(v^i) = \int_{v^i - 0.5}^{v^i} 4(v^i - t_i^i) dt_i^i + \int_0^{v^i - 0.5} 4 - 4(v^i - t_i^i) dt_i^i = -2{v^i}^2 + 4v^i - 1$$

Case 3: $1.5 \leq v^i \leq 2 \rightarrow v^i - 1 \leq t_i^i \leq 1$ for $4 - 4(v^i - t_i^i)$

$$\int_{v^i - 1}^{1} 4 - 4(v^i - t_i^i) dt_i^i = 2{v^i}^2 - 8v^i + 8$$

$$f_{v^i}(v^i) = \begin{cases} 2{v^i}^2 & \text{if } 0 \leq v^i \leq 0.5 \\ -2{v^i}^2 + 4v^i - 1 & \text{if } 0.5 \leq v^i \leq 1.5 \\ 2{v^i}^2 - 8v^i + 8 & \text{if } 1.5 \leq v^i \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

The cumulative distribution function for $f_{v^i}(v^i)$ can now be found through the integrands of the piecewise function.

$$\int_0^{v^i} 2{v^i}^2 = \frac{2}{3}{v^i}^3$$

$$\int_{0.5}^{v^i} -2{v^i}^2 + 4v^i - 1 = -\frac{2}{3}{v^i}^3 + 2{v^i}^2 - v^i + \frac{1}{12}$$

$$\int_{1.5}^{v^i} 2{v^i}^2 - 8v^i + 8 = \frac{2}{3}{v^i}^3 - 4{v^i}^2 + 8v^i - 5.25$$

$$F_{v^i}(v^i) = \begin{cases} 0 & \text{if } v^i \leq 0 \\ \frac{2}{3}{v^i}^3 & \text{if } 0 \leq v^i \leq 0.5 \\ -\frac{2}{3}{v^i}^3 + 2{v^i}^2 - v^i + \frac{2}{12} & \text{if } 0.5 \leq v^i \leq 1.5 \\ \frac{2}{3}{v^i}^3 - 4{v^i}^2 + 8v^i - \frac{13}{3} & \text{if } 1.5 \leq v^i \leq 2 \\ 1 & \text{if } v^i \geq 2 \end{cases}$$

This cumulative distribution function can subsequently be used to find the probability density function and cumulative distribution function of the maximum valuation of the firms apart from i. Because the distributions are symmetric $f_{v^i}(v^i) = f_{v^j}(v^j)$ and $F_{v^i}(v^i) = F_{v^j}(v^j)$.

$$F_{v_j^{max}}(v^j) = F_{v^j}(v^j)^{n-1}$$

$$f_{v_j^{max}}(v^j) = (n-1)F_{v^j}(v^j)^{n-2}f_{v^j}(v^j)$$

Thus, for $n = 3$, the equilibrium bidding strategy is defined as follows.

$$b^{i^*} = \begin{cases} 0 & \text{if } v^i \leq 0 \\ \frac{1}{\frac{2}{3}{v^i}^6} \int_0^{v^i} 2y\frac{2}{3}y^3 2y^2 dy & \text{if } 0 \leq v^i \leq 0.5 \\ \frac{1}{(-\frac{2}{3}{v^i}^3 + 2{v^i}^2 - v^i + \frac{2}{12})^2} \int_0^{v^i} 2y(-\frac{2}{3}y^3 + 2y^2 - y + \frac{2}{12})(-2y^2 + 4y - 1)dy & \text{if } 0.5 \leq v^i \leq 1.5 \\ \frac{1}{(\frac{2}{3}{v^i}^3 - 4{v^i}^2 + 8v^i - \frac{13}{3})^2} \int_0^{v^i} 2y(\frac{2}{3}y^3 - 4y^2 + 8y + \frac{13}{3})(2y^2 - 8y + 8)dy & \text{if } 1.5 \leq v^i \leq 2 \end{cases}$$