Towards Peer-To-Peer Double Auctioning

Zoran Despotovic*, Jean-Claude Usunier†, Karl Aberer*

*Distributed Information Systems Laboratory
Institute of Core Computing Sciences
Swiss Federal Institute of Technology
Lausanne, Switzerland
email: zoran.despotovic@epfl.ch, karl.aberer@epfl.ch

†Institute of International Management
Graduate School of Business
University of Lausanne, Switzerland
email: jean-claude.usunier@hec.unil.ch

Abstract

P2P systems constitute nowadays an increasingly important part of the online world that needs or will soon need all kinds of e-commerce services and applications that are normally available today on client-server platforms. When designing new e-commerce solutions in P2P particular attention must be paid to masking autonomy of the peers and the lack of any central authorities as two most important problems. In this paper we explore possibilities to bring e-commerce into P2P and propose a double auctioning mechanism that does not rely on the existence of central authorities, auctioneer in particular, and is amenable to implementation in P2P environments. The mechanism has good economic properties such as, for example, fast convergence towards efficient trading through intuitive and simple bidding strategies.

1. Introduction

In spite of the effort that recording industries are making to protect their copyrights and disable MP3 songs distribution in P2P networks, P2P is here to stay. On the one hand, the client-server architecture, currently prevalent architectural solution, is being pushed very close towards its limits with respect to bandwidth consumption and scalability problems. On the other hand, network communities show nowadays a strong desire for autonomy and independence of any authorities and this idea will not be easily abandoned in the future. These are in our opinion two most important driving forces that bring us to the conclusion that P2P is not just a technological innovation of the day. The fact that almost $1 bln has been invested in P2P companies in the last two years goes well in line with this statement.

At present, P2P systems are mainly used in simple (mostly music) file sharing scenarios. But, the time when the potential of the P2P architecture, including distributed computing and P2P e-commerce for instance, will be fully exploited is not far away. This paper presents our attempt to bring e-commerce on P2P platforms. In particular, possibilities for making double auctioning markets on top of P2P are explored. We identify some desirable properties that a P2P-suitable double auctioning mechanism should have as well as possible problems related to a straightforward P2P implementation of the Continuous Double Auction (CDA). Based on the developed ideas we propose a new trading mechanism that is easily “P2P-able”. We also check its economic characteristics and show that a simple reasoning is sufficient to enable economic agents to quickly discover the equilibrium price and the way towards efficient trading.

The paper is organized as follows. Section 2 motivates the need for P2P computing and presents a short overview of the technology. Section 3 identifies the main problems that a P2P marketplace design would normally imply and motivates the way we proceed. In Section 4 we review main results of the double auction theory. Special attention is paid to the CDA as our trading is essentially an adaptation of the CDA to P2P environments. Section 5 defines our trading mechanism while empirical results related to its economic properties are presented in Section 6. Section 7 concludes the paper and outlines the future work.

0-7695-2056-1/04 $17.00 (C) 2004 IEEE

2. P2P: a technology overview

Peer-to-peer computing is certainly not a new idea. The Internet itself started as a P2P system. Originating in an academic environment with a relatively small number of users, in which cooperation was a goal and a value, it was easy for such a network to meet the needs of its users. But, the subsequent commercialization of the Internet as well as the availability of inexpensive high performance computing equipment brought an easy opportunity for literally everyone around the Globe to come online. Thus, under these new circumstances, new architectural solutions were needed to meet growing business (e-commerce) needs and circumvent serious security hazards. The World Wide Web and client-server architecture perfectly met such needs. It became easy for a company to set up a web site and provide various services to its clients. eBay, currently the largest auctioning site world wide, is a good example. What eBay does is essentially providing an online marketplace (in the form of a web service) in which its 49.7 million (the figure was announced on eBay's web site) registered users can compete in an auction-like manner to buy or sell the most diverse goods. But, when playing with so large numbers the client-server architecture has certain serious limitations - resources are concentrated on a small number of nodes and network bandwidth becomes a nightmare under such circumstances. As well, providing continuous and reliable access becomes a serious problem. These reasons, among many others, are bringing the idea of P2P computing alive again.

Peer-to-peer computing is most commonly defined as sharing of computer resources (disk storage, processing power, exchange of information, etc.) by direct communication between computing systems. Thus, it takes advantage of already existing computing resources allowing the whole group of computers to more effectively make use of their collective power. Every participating node in a P2P system can act as both a client and a server (and at the same time).

The most important design requirements for a P2P environment can be split into the following two confronted groups. On the one hand, the system must be decentralized in the sense that there must be no central authority that would present a single point of failure and that the global behaviour of the system should emerge from local interactions among the peers only. On the other hand, there is the requirement for high performances of the system in terms of low search latency, message bandwidth and storage costs.

Most of currently available P2P solutions (Gnutella [1], CHORD [12], P-Grid [2], just to mention a few) fulfill these requirements to a high degree. Though this degree varies from system to system, the main difference among P2P systems is in the way this is achieved. P-Grid, for example, builds a distributed binary search tree on the hashed content files. By means of an appropriate randomized algorithm, the tree is built only from local interactions among the peers. In any case, the result is highly efficient, logarithmic search. Robustness of the tree against peer or network failures is achieved by a suitable replication of the tree structure. On the other hand, Gnutella operates without any auxiliary indexing structure. A search request is flooded through the network until a peer having the file searched for replies back. It was shown that Gnutella networks have small-world properties, which keeps search latency at low levels.

3. P2P and markets: common problems

The lack of central trusted authorities and the autonomy of the peers present the two major obstacles one faces when designing double auctioning markets in P2P. Let us describe this assertion in more detail.

Namely, when designing a market in general its designer's primary goals are that:

1. the market is efficient (in the sense that no further gains from the trades are possible) and that
2. the economic agents are really willing to participate in the market (in the sense that there is enough trust that they will fulfil their promised actions and that the quality of their services will be as claimed).

To achieve the first goal the trading rules of the mechanism must be set in a proper way (we touch this problem in the next section). As for the second one, branding, formal contracting or litigation are established ways for achieving it. But, in a P2P environment, neither of these two goals can be achieved by straightforwardly applying the already established solutions.

With respect to the first goal, the problem we face is that all known double auctioning mechanisms base their proper functioning (efficiency in particular) on, at least, some form of centralization. As decentralization is an important design requirement for a P2P environment it is clear that a P2P-suitable trading mechanism must be free of any central components. A new mechanism that fulfils this requirement will be described throughout the rest of the paper.

As far as the second goal is concerned, the problem we face stems from the expectation that most of the transactions in a P2P environment might be at small stakes so that the mentioned assurance mechanisms are simply inefficient. To this end, we are assuming existence of social ways to achieve this goal such as trust and reputation management mechanisms. A number of decentralized reputation management schemes, appropriate for P2P networks, arrived in recent years ([3] and [15] for instance). Their functioning and related details are out of the scope of this paper. The exact influence of these works on the resulting market will be studied in the future.

4. Double auction theory overview

In this section we review main results of the double auction theory. It is not our intention here to give a comprehensive overview. Rather, the description that follows is tailored to suit the understanding of the rest of the paper. Particular attention is devoted to the continuous double auction as this trading mechanism bears some similarity to our approach.

An auction is usually defined as a market institution with an explicit set of rules determining resource allocation and prices on the basis of bids from the market participants [8]. In one-sided auctions there is one seller and many buyers willing to buy the auctioned item (the case of one buyer and many sellers is technically fully equivalent to this case), while in double-sided auctions there are many buyers and many sellers submitting bids simultaneously. The (double) auctioning problem has two sides. From the perspective of the auction participants, the problem is how to bid optimally and extract as much gain from the trade as possible given the valuation of the good the participant is about to trade and her knowledge of the trading rules as well as her beliefs about the valuations of the other participants. From the point of view of the auction designer, the problem is to design the auction rules in such a way that buyers (sellers) who value the goods the most (least) receive the goods (money) in the end so that no further gains from the trade are possible. The trading mechanism efficiency, defined as the sum of the gains of the auction participants who trade according to the rules and submitted bids, relates to this idea in an obvious way.

In a double auction, because the bidders' strategic behaviour must be modelled on both sides of the market, game-theoretic analysis of the double auction problem becomes substantially harder and analytic results are not so numerous as in the case of one-sided auctions. (A notable exception is Myerson-Satthertwaite impossibility result [9] claiming that the only way to get truthful bidding as the optimal strategy in a fully efficient and individually rational trading mechanism is to provide outside subsidies.) The rules governing the clearing policy of the market as well as the determination of the (pairs of) traders among buyers and sellers as well as the transaction price(s) play the most important role in double auctions classification. With respect to how frequently the auction market is cleared a distinction is made between continuous auctions (the continuous double auction, CDA, being the most widespread representative), that match buyers and sellers and clear the market immediately upon arrival of compatible bids, and synchronous variants (often referred to as call markets), that collect bids over a prespecified time interval and clear the market only once upon the expiration of this interval. The transaction prices determination policy makes a further distinction among double auction types by grouping them into uniform-price auctions, in which all traders trade at the same price, and multiple-price auctions, in which different pairs of traders trade at different prices.

[10] provides an analysis of a call market modelled as k-double auction. In a k-double auction ($k \in [0, 1]$) bidders submit sealed bids, then the trading price is computed and the market is cleared. The trading price is computed in the following way: denote by M and N numbers of buyers and sellers and a and b the M-th and $(M+1)$-st lowest bid (among both buy and sell bids). Then the price is determined as $p = ka + (1-k)b$. If m and n denote the number of buy offers above the price and the number of sell offers below it, then, for $c = min(m, n)$, the set of traders consists of c highest buy offers and c lowest sell offers.[1] For $k = 0$ this mechanism is incentive compatible for sellers, while for $k = 1$ it is incentive compatible for buyers ([14]). But, for any $k \in (0, 1)$ is not incentive compatible neither for buyers nor for sellers. Therefore, in this case truthful bidding is not an equilibrium but, as shown in [10], when numbers of buyers and sellers, M and N respectively, grow at the same rate (the ratio M/N being bounded both from above and away from zero), then all equilibria are within $O(1/M)$ of the truthful bidding and expected inefficiency of any equilibrium is $O(1/M^2)$.

As we will see in Section 5, the trading mechanism we propose, though *not* being synchronous, bears some similarity with the 0.5-double auction under the assumptions that the numbers of buyers and sellers are large, bidding is highly intensive and it develops in a networked environment. This similarity constitutes an important idea we pursued when defining the mechanism.

4.1. The continuous double auction

The continuous double auction is certainly one of the most dominant market institutions (it is used in the markets such as NASDAQ and NYSE). Its definition is rather simple, it consists of the following rules [7]:

- Bidding develops in prespecified periods;
- At any time during a bidding period any buyer (seller) may submit an offer to buy (sell) a certain good at the submitted price whereby this offer is observed simultaneously by all other buyers and sellers;

---

[1] It can be easily verified that $a \neq b$ implies $m = n$. Thus, in this case all buyers bidding above c trade with all sellers bidding below c.

- If a buyer's (seller's) bid is acceptable to a seller (buyer) then a trade is executed between them and all other buyers and sellers get informed about this trade.

Besides these rules, several other rules are common to the CDAs found in practice. The spread-improvement rule, as the most widespread one, implies that buy bids (sell asks) can be submitted only if they are higher (lower) than the currently highest bid (lowest ask).

As noted above, double auctions are generally very hard to analyse game-theoretically. The equilibria of the sealed-bid double auctions cannot be expressed in a closed form for even small number of players. If we know that with the CDA time dimension has to be also included then we see why all results about it are based on simulations and experiments. The most intriguing empirical result regarding the CDA, that all experiments with human subjects show ([11] presents a seminal work in this area), is that the trading prices converge very quickly to the price computed as the intersection point of the *true* demand and supply curves (in the literature often referred to as the competitive equilibrium (CE) price). To be exact, after only several trading periods the trading prices become very close to this price and the trading becomes almost fully efficient. Keeping the above definition of the CDA in mind the following important conclusion should be obvious: no central institution and no global knowledge of the market conditions is needed to have a market with efficient allocations of the traded resources; they can be achieved by local interactions of the market participants each of whom possesses only partial knowledge of the conditions. This assertion is of particular importance in a P2P environment where, as a design requirement, such central institutions cannot exist.

In the last decade the CDA became an important subject of research in artificial intelligence. In particular, many models of bidding software agents that can replace human traders were developed. Among the most prominent ones is [5] that introduces a simple learning mechanism based on reinforcement learning and shows that a market consisting of the agents equiped with this learning mechanism retains the characteristics of experimental markets studied in [11], i.e. a fast convergence of the prices towards the equilibrium price and high efficiency. The learning strategy we use in Section 6.2 to test the quality of our mechanism is essentially a variation of this strategy.

[7] developed a model in which all market participants follow an intuitive strategy - upon observing the trading history they form a subjective belief function that gives the probability that any given bid or ask will be accepted and then choose one that maximizes their expected gains. It was shown that the CE price is reached after 5-6 trading periods only. [6] experimented with a market consisting of both software agents and human traders. Thus their environment was not uniform in the sense that different market participants used different strategies and not all of them were humans. Their experiments showed that the software agents achieved larger gains than their human counterparts. They also observed persistent "far-from-equilibrium trading".

5. Open electronic bargaining system (OEBS) - a P2P suitable trading mechanism

We now turn our attention towards some specifics of the double auctioning in P2P environments and defining our trading mechanism. The idea is to derive the definition by outlining some desirable properties that a P2P suitable trading mechanism should have and possible problems related to a straightforward implementation of the CDA in a P2P environment. (We do not discuss here all aspects of the mechanism and the corresponding bargaining market. See [13] for a full description.)

Generally speaking, a P2P suitable trading mechanism must be free of any central authorities (auctioneer in particular) and it must be possible to scalably distribute it among the peers. The synchronous mechanisms (call markets) are highly inappropriate for P2P implementation. The tasks of collecting bids, forming the apparent demand and supply curves, computing the trading price and clearing the market make the job of the auctioneer in the call markets and devising a deception free distributed implementation of these tasks is not easy, if not impossible. On the other hand, the auctioneer's role is much less emphasized in the CDA and, apart from certain monitoring tasks such as assuring that offered prices are taken without any further bargaining between the traders, it can be subdued essentially to collecting the bids and informing the traders about them.[2] Clearly, in a P2P network this task can be easily implemented simply by having the bids broadcasted (or multicasted, to be precise) so that all the interested traders get immediately aware of them. For this reason the CDA may appear at first glance as a good candidate for implementation in P2P environments. However, there are certain problems occurring if we try to straightforwardly transfer the CDA into a P2P environment.

First, all mentioned CDA experiments and simulations work with centralized versions of the CDA in which all traders get informed about all relevant events (new shouts, trades, etc.) *at the same time*. None of them shows that the CDA retains its good behavior when different auction participants get informed about these events at different time instants as expected in a networked environment owing to its inherent latencies. This has to be checked before claiming that the CDA is appropriate for use in a computer network. In particular, this has to be checked in a P2P setting

---

[2] We use the notion of auctioneer here as a symbol to denote any aspect of centralization. It does not really matter whether the mentioned tasks are realized by one person or by some other means, say, by having all the traders gathered at one place and shouting their bids and offers.

with its characteristic distributions of the latencies between the nodes.

Second, consider the following scenario in which a seller s broadcasts a price p at time t. Due to different network latencies this information will reach potential buyers at different times. Knowing this and expecting that the seller s will not make the deal with the very first replier, every buyer with the valuation above p has incentive not to take the price p but to offer slightly more to the seller s. Even doing so those buyers' surpluses can be still positive. On the other hand, expecting this, the seller is tempted to wait for a period of time and take the best counteroffer received in this period instead of taking the first one of them. While this sort of (possibly) degenerate behavior is not present in the existing CDA implementations due to their inherent forms of centralization (say, it is eliminated by contractual agreements between the participants and the auction organizer which monitors the bidding process and makes sure that all the participants conform to the rules), it cannot be ruled out in a P2P implementation of the CDA. Simply, two neighboring nodes can engage in the described scenario and their deal cannot be even observed by the rest of the traders. Whether this behavior is really degenerate or not, or whether it harms the good properties of the CDA or not, is an open question and has to be checked by introducing such a possibility in the mechanism definition itself.

Third, the perfect continuity of the CDA market (the full sequentiality of the bid arrivals) cannot be achieved in a networked environment. To be exact, if the rate at which the bidders submit their bids (the bidding intensity) and the network latency are high then the following situation can be quite common: m buy and n sell bids ($m, n > 1$) are submitted within a short time interval; they "meet on the wire" and the corresponding m buyers and n sellers end up with identical sets of bids. Then it becomes unclear who trades with whom and at which price. The original price setting rule of the CDA does not anticipate this, apparently conflicting, situation. For this reason, we opt to a different price setting rule in our mechanism that overcomes this problem. At the heart of our price setting rule, that is inseparable from the rule determining who trades with whom, is the idea that the market can be cleared by local interactions of its, possibly self-interested, participants even in the case of the conflict we just mentioned. This can be achieved if the trading prices are set to the middle points between matched bids and asks. Then, every buyer has incentive to trade with the announcer of the lowest sell ask that the buyer observed. Similarly, any seller would want to trade with the announcer of the highest observed bid. It is easy to see that now self-interested (utility maximizing) behavior of the market participants leads to the highest buy bid being matched with the lowest sell ask, the second highest buy bid with the second lowest sell ask and so on, so that market can be cleared. This is illustrated in Figure 1. The bids are denoted by the solid lines while the dashed lines denote the asks. If all the involved traders know this picture then it is clear that buyer 1 must trade with seller 7 (the price will be set to $\frac{x_1+x_2}{2}$) and buyer 2 with seller 5 (the price is $\frac{x_2+x_5}{2}$) while the other traders cannot trade.

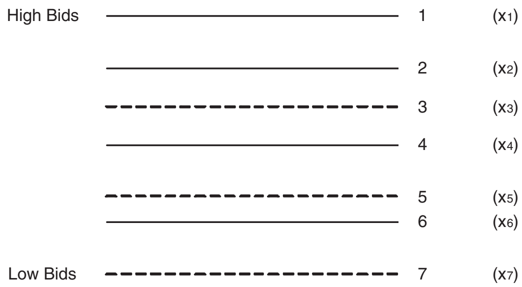

Figure 1. A trading scenario example

Thus, this reasoning brings us to the following mechanism definition.

The Mechanism Definition Rules:

- Bids can be submitted asynchronously, at any point in time; there is no time constraint.
- Any observed bid can be replied with a counteroffer or simply ignored. In the case the counteroffer is accepted by the bid originator then it is assumed that a deal is made between two involved parties and that the trading price is set to the middle point between the two offered values.

6. OEBS - simulation results

In this section we describe our empirical findings about the economic characteristics of our trading mechanism. More specifically, we present the preliminary results of our simulations that show how the two most important parameters reflecting the economic quality of the mechanism, its efficiency and the trading prices achieved in the resulting market, behave in a number of settings that approximate the real world P2P environments. The section is structured in the following way. The simulation settings and the details of the simulated environments are described in Section 6.1. Section 6.2 presents the bidding strategy used by the peers. The results of the simulations are given in Section 6.3.

6.1. The simulation settings

The core mechanism definition rules were given in the previous section. But, to provide a clear picture of what was and what was not modelled in the simulations we must further clarify our assumptions related to the underlying P2P environments as well as the ways of applying the rules in them (the bidding protocol).

First, as far as the underlying P2P network is concerned the only parameter we consider in these simulations are the latencies among the peer nodes in the network. To obtain as realistic latencies as possible we start from the power-law network topology (generated according to [4]), assign a latency to any branch in the graph and then calculate the latencies between any two nodes by applying the all-pairs shortest path algorithm.[3] As a result, we obtain the latency matrix that is used as the input parameter representing the network.

The fact that we do not take into consideration any other parameter of the underlying network, and the TTL (time-to-live) value in particular, means that any bid, when broadcasted to the network in Gnutella-like manner, will eventually reach all other peers interested in receiving it. Whether this will be the case or not depends also on whether intermediate peers are honest enough to forward the bid or not. Further, upon a bid reception any receiving peer can reply with a message containing its own counteroffer, but now, unlike the bid submission, the offer is sent to the original bid sender only (its address is available in its bid containing message). With respect to this, we are assuming that these replies reach their destinations with certainty and are not observed by other auction participants.

Finally, our last assumption regarding the bidding protocol is that when a reply to a bid arrives at the bid originator it is up to the originator to accept or reject the counteroffer. We are assuming in the simulations that in either cases an appropriate message is sent back to the reply issuer. (Of course, it is up to the receiving peer's strategy to decide which message will be sent). On the other hand, the reply issuer is obliged to wait for the answer or the timeout expiration before sending another reply to a bid that it might receive while waiting or submitting its own bid during this time period. It is interesting to observe that deviations from these behaviors present a good example of a situation in which trust and reputation management can help greatly. Roughly, the idea is that certain misbehaviors are recorded by the peers and this information is made available to the rest of the network so that the future business of the misbehaving entities is negatively influenced. We refer again to [3] and [15], for instance, for realizations of this idea in a decentralized fashion. However, we do not model trust and reputation possibilities in our simulations. The exact influence of concrete P2P trust management solutions on the mechanism quality will be studied in the future.

6.2. ZIP-modified bidding strategy

Common to all our experiments is the bidding strategy used by the peers. It is a variation of the Zero-Intelligence-Plus (ZIP) learning strategy introduced in [5] modified to fit our trading rules.

[5] considered machine learning techniques that could allow autonomous software agents to perform in a human like manner in the context of the CDA. The Zero-Intelligence-Plus (ZIP) bidding strategy was introduced as the very minimal learning mechanism needed to achieve this goal and it was shown that the performance of the software agents equiped with this strategy ("ZIP traders") is nearly as good as that of the human agents presented in [11]. As discussed in Section 4.1, this good performance essentially implies fast discovery of the equilibrium price and efficient trading.

The main idea of the ZIP strategy is that the traders maintain adjustable shout prices (or equivalently, profit margins, expressed as fractions of their limit prices), that are updated on the basis of observed shouts made by the other agents and whether these shouts were accepted or not. So called delta learning is used to compute the adjustment levels. Thus, when a price $q(t)$ is shouted, provided an update needs to be made (see below), then every trader i updates his shout price according to the formula:

$$p_i(t+1) = p_i(t) + \Gamma_i(t), \tag{1}$$

where

$$\Gamma_i(t) = \gamma_i \Gamma_i(t-1) + (1-\gamma_i)\beta_i(q(t-1) - p_i(t-1)) \tag{2}$$

Here β_i and γ_i are trader i's learning rate and momentum coefficients and, by definition, $\Gamma_i(0) = 0$. The rationale behind this learning method can be explained as follows. Assuming that the momentum coefficients $\gamma_i = 0$ and that the target price $q(t)$ remain constant then it can be easily seen that $p_i(t)$ converges to $q(t)$ at a speed determined by the learning rate coefficient β_i. The meaning of the momentum coefficients is also very simple. If, for example, the price was increased in each of several last updates and then the shout price at time t indicates that a decrease is needed it might be better not to decrease the price immediately but rather to decrease the rate of increase. By introducing the momentum coefficients, that make the entire history of updates matter when computing the new shout price, exactly this is achieved.

The last important detail of the ZIP strategy is when exactly updates are made. So, let $p_i(t)$ and $q(t)$ be the trader i's learned value of the price and the shout made at time t. Assume as well that trader i is a seller. Then, seller i updates its learned price when one of the following three conditions is met (the conditions for the buyers' updates can be specified in like manner):

---

[3] The latency of a branch is taken to be the maximum value of the latencies of its corresponding nodes, which are in turn inversely proportional to their degrees.

- The shout was an offer, it was not accepted, and $p_i(t) \geq q(t)$. In this case the seller should lower his shout price. Indeed, provided the momentum coefficient $\gamma_i = 0$, we can see from (1) and (2) that $p_i(t+1) \leq p_i(t)$.

- The shout was accepted (no matter it was a bid or an ask) and $p_i(t) \leq q(t)$. The seller should now increase his shout price. This is exactly what (1) and (2) do in this case ($p_i(t+1) \geq p_i(t)$).

- The shout was accepted, it was a bid, and $p_i(t) \geq q(t)$. In this case we have again $p_i(t+1) \leq p_i(t)$.

There are several problems with an eventual straightforward application of the above learning method to our trading mechanism. First, because our market is fully decentralized we cannot assume that the previously achieved trading prices are known to the bidders before they make their own shouts. Put another way, when a trade is made between a buyer and a seller we cannot assume that they will communicate their price to the rest of the bidders. Thus, on observing a bid the traders must form a belief whether that bid will be accepted or not and, if they believe that the observed bid will result in a trade, then they must guess at what price the trade will be made. To fix these difficulties we apply the following reasoning.

We are assuming that all the traders apply the above described technique to learn both buy bids, $B(\cdot)$, and sell asks, $S(\cdot)$. Precisely, if $B_i(t)$ and $S_i(t)$ denote the learned values of bids and asks for trader i at time t and if this trader observes a bid $Bid(t)$ (or alternatively, an ask $Ask(t)$) then he updates his learned value $B_i(t)$ (or $S_i(t)$) according to the formulas (1) and (2) with $p_i()$ and $q(t)$ replaced with $B_i()$ and $Bid(t)$ (or with $S_i()$ and $Ask(t)$). Now, we assume that trader i conjectures that the bid (ask) will result in a trade if its value is higher (lower) than the currently learned value of the sell asks (buy bids), or formally if $Bid(t) > S_i(t)$ ($Ask(t) < B_i(t)$), and that the price of the trade will be $q_i(t) = \frac{Bid(t)+S_i(t)}{2}$ ($q_i(t) = \frac{Ask(t)+B_i(t)}{2}$).

The assumed trading price $q_i(t)$ is then used to update the learned price and compute $p_i(t+1)$ just as in (1) and (2) with $q(t)$ replaced by $q_i(t)$. This value, $p_i(t+1)$, is trader i's conjecture about the price that will be achieved at time $t+1$. Therefore, the shout price of trader i at time $t+1$ becomes $Bid_i(t+1) = 2p_i(t+1) - S_i(t+1)$ if trader i is a buyer, or $Ask_i(t+1) = 2p_i(t+1) - B_i(t+1)$ if he is a seller.

As for the question when exactly the specified updates of $p_i()$ are made, the same rules as those given above for the case of ZIP traders are used.

	Average Number of Bids Sent per Trader				
Number of bidders	1	2	3	4	5
50	47.6	26.7	13.7	7.9	6.8
100	37.2	15.5	9.1	8.8	7.5
150	34.7	31.3	14.9	8.1	6.7
300	34.5	10.2	7.1	6.1	6.0
500	30.1	10.0	6.5	5.6	5.2
750	32.3	10.1	7.2	6.2	5.3

Figure 2. Dependence of the average efficiency loss on the overall number of participating traders and the average numbers of bids they sent

6.3. Results

Figure 2 shows how the average efficiency loss, expressed as the percentage of the totally attainable efficiency, depends on the size of the market and the time spent on bidding. The size of the market is expressed in the number of participating traders and only symmetric markets (with equal numbers of buyers and sellers) are represented in this figure. The bidding time is expressed as the average number of bids sent per trader. All the results assume that the traders were entitled to sell and buy only one unit each and, as mentioned, that they all use the strategy described in the previous section.

Let us comment on these results. The very first conclusion we can draw by inspecting the figure is that the market is not fully efficient but the efficiency loss falls quickly within 6–7%. We believe that this loss of few percents is a tolerable price to pay for having a fully decentralized market. Also, we have to add here that these numbers are not final; they are achieved for a specific bidding strategy used by the traders and specific (ranges of) values of the strategy related parameters. There might exist alternative strategies or different parameter values of the current strategy that would cut down the losses even more.

Another important conclusion is that the market efficiency increases as the number of the participating traders grows. This is particularly important for P2P environments as the number of the market participants quite frequently tends to be very high.

Worth mentioning is also that, in average, the traders need to submit only several bids in order to realize the trades. This means that the mechanism and the specific strategic behavior shown here incur a short time having to be spent on bidding and, consequently, lower transaction costs for the traders.

Another question that would be interesting to discuss is what would happen if we had the traders entitled to trade many units and whether the trading efficiency would become better. (This sort of experiment was carried out in [5] and [11].) We actually experimented with this kind of setting but did not observe any considerable improvement. An intuitive explanation for this is that the trading bids stabilize at points far away from the equilibrium price, which, along with a random determination of the trading pairs caused by the network latencies makes it impossible to have a convergence of the realized prices towards the equilibrium price and a long-run improvement as observed in the mentioned works.

	Total Number of Traders				
Fraction of Buyers	100	150	300	500	750
1/2	7.5	6.7	6.0	5.2	5.3
1/3	18.1	18.2	17.9	17.7	17.7
1/4	31.4	29.8	30.1	29.6	29.3
1/5	39.8	39.6	39.3	40.1	39.1

Figure 3. Average efficiency loss for asymmetric markets

Figure 3 shows the losses in the case of asymmetric markets (the asymmetry level is given as the fraction of the buyers in the overall trader population). As can be seen from the figure the market retains an acceptable efficiency until the population of the sellers becomes twice the size of the buyers. For higher asymmetries the mechanism becomes unacceptably inefficient when the described strategy is used.

7. Conclusion

In this paper we explored possibilities for making C2C marketplaces in P2P environments. We outlined some desirable properties that a P2P suitable double auctioning mechanism should have and discussed some potential problems that the straightforward implementation of the continuous double auction in a P2P setting might have. Based on these we proposed a new trading machanism that does not bear similar difficulties. The preliminary simulations that we carried out show that the mechanism retains many good economic behavior as that of the CDA. However, the setting considered in the simulations only approximate typical P2P environments and an investigation of the mechanism performance in a more realistic P2P setting constitutes the most important part of the future work.

As well, the study of the exact influence of decentralized trust and reputation management schemes on economic quality of the mechanism and the entire resulting market is one of the most important considerations for the future.

References

[1] Gnutella homepage. http://gnutella.wego.com.

[2] K. Aberer. P-grid: A self-organizing access structure for p2p information systems. In *Proceedings of the Sixth International Conference on Cooperative Information Systems (CoopIS 2001)*, Trento, Italy, September 5-7 2001.

[3] K. Aberer and Z. Despotovic. Managing trust in a peer-2-peer information system. In *Proceedings of the Ninth International Conference on Information and Knowledge Management*, Atlanta, Georgia, November 2001.

[4] A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[5] D. Cliff. Minimal-intelligence agents for bargaining behaviors in market-based environments. Technical report hp-97-91, Hewlett Packard Laboratories, Bristol, England, 1997.

[6] R. Das, J. E. Hanson, J. O. Kephart, and G. Tesauro. Agent-human interactions in the continuous double auction. In *Proceedings of IJCAI-01*, Seattle, USA, August 2001.

[7] S. Gjerstad and J. Dickhaut. Price formation in double auctions. *Games and Economic Behavior*, 22:1–29, 1998.

[8] R. P. McAfee and J. McMillan. Auctions and bidding. *Journal of Economic Literature*, 25:699–738, 1987.

[9] R. B. Myerson and M. A. Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 29:265–281, 1983.

[10] A. Rustichini, M. Satterthwaite, and S. Williams. Convergence to efficiency in a simple market with incomplete information. *Econometrica*, 62:1041–1063, 1994.

[11] V. L. Smith. An experimental study of competitive market behavior. *Journal of Political Economy*, 70:111–137, 1962.

[12] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proceedings of the 2001 ACM SIGCOMM Conference*, pages 149–160, 2001.

[13] J.-C. Usunier. An open electronic bargaining system. Working paper, IUMI, 2002.

[14] P. R. Wurman, W. E. Walsh, and M. P. Wellman. Flexible double auctions for electronic commerce. *Decision Support Systems*, 24:17–27, 1998.

[15] L. Xiong and L. Liu. Peertrust: A trust mechanisam for an open peer-to-peer information system. Technical report git-cc-02-09, Georgia Institute of Technology, 2002.