Algorithm Design for Agents Which Participate in Multiple Simultaneous Auctions

Chris Preist
Trusted E-Services Laboratory
HP Laboratories Bristol
HPL-2000-88
10th July, 2000*

E-mail: cwp@hplb.hpl.hp.com

auctions, eServices, agents, negotiation

In this paper, we discuss the design of algorithms for agents to use when participating in multiple simultaneous English auctions, aiming to purchase multiple goods. Firstly, we present a coordination algorithm, which ensures the agent places appropriate bids in the different auctions to buy exactly the right number of goods. Secondly, we combine this with an algorithm to determine what maximum bid an agent should place in an auction which is about to terminate. This algorithm combines a belief-based model of the auctions with a utility analysis. This analysis is to trade off the certain outcome of the terminating auction against the possible outcomes of the remaining auctions, and hence place appropriate bids in each.

* Internal Accession Date Only
Approved for External Publication

Introduction

Electronic commerce [1] is having a revolutionary effect on business. It is changing the way businesses interact with consumers, as well as the way they interact with each other. Electronic interactions are increasing the efficiency of purchasing, and are allowing increased reach across a global market.

E-commerce is not a static field, but is constantly evolving to discover new and more effective ways of supporting business. Initially, e-commerce involved the use of EDI and intranets to set up long-term relationships between suppliers and purchasers. This increased the efficiency and speed of purchasing, but resulted in lock-in in the relationship. Both suppliers and purchasers had to invest significantly up-front in the relationship, so were not easily able to move their business elsewhere. The technological relationship between the parties was a friction factor, preventing free competition in the longer term [2]. Often, the relationship was (and still is) beneficial to both parties. However, the lock-in effect also meant that when the relationship became less beneficial to one party, they couldn't easily move elsewhere.

The second phase of e-commerce aimed to address this problem. With the increasing availability of the web, a more open e-commerce environment is developing, allowing businesses to trade more flexibly with each other. Some of this openness is achieved by competition between web portals, while some competition occurs within a single web portal, acting as a marketplace for buyers and sellers to meet. Some of the efficiencies of EDI can now be achieved in a more open environment, where relationships no longer need to be long-term.

However, there is a benefit of the EDI approach that is often lost in this new phase. Price negotiation was carried out in advance in the EDI world, so purchasing can be entirely automated. When a manufacturing planning and forecast system identifies the need for a purchase, it can initiate it automatically without any human involvement, increasing speed and efficiency. In phase two, each purchase may involve interaction with a new supplier, and so may involve new negotiation of terms. As a result of this, many of these purchases can't be made automatically, and instead require human interaction, mediated by the web.

The third phase of electronic commerce is just beginning. It aims to address this issue, allowing automated business interactions to take place in a fluid environment. Technology will no longer be a friction factor to changing supplier or customer. Long-term relationships will still play an important role, but they will persist because of the choice of both parties rather than technological lock-in. The key building blocks of this new world, e-services, will be able to interact dynamically with each other to create short-term or long-term trusted trading relationships to satisfy the needs of different business partners. Many technologies are involved in this development – distributed systems, encryption and PKI, XML and associated business ontologies, economic analysis and game theory, to name just a few. As automation and distribution are central to the vision, agent technology provides a fundamental role in this.

In Section 2, we discuss the role of auctions in electronic commerce, present the problem of multiple auction participation and give an overview of tools used to support participation in auctions currently. In section 3, we present a basic coordination algorithm used to bid effectively in multiple auctions which terminate simultaneously. In section 4, we extend this algorithm by providing a belief-based learning component, and use utility analysis to determine

whether it is worth making a purchase in an auction about to terminate. In section 5, we present related work, and in section 6 we conclude with future directions for this work.

2 Auctions in Electronic Commerce

In phase two of the e-commerce revolution, auctions have become increasingly important both for business to business transactions and for consumer purchasing. Many different auction designs are possible. Game theory [3] can be applied to study how best to design them for specific situations [4]. Among the most popular designs are English, Dutch and Vickrey auctions. In an English auction, a seller offers a good for sale and buyers bid the price they are willing to pay. Each bid announced must be greater than the previous bid, and the item is sold to the highest bidder. In a Dutch auction, the process runs in reverse. The seller announces a proposed price, and buyers can accept it if they choose. As time progresses, the seller decreases the proposed price until a buyer accepts. In a Vickrey auction, bidders place their bids in a sealed envelope and submit them to a trusted third party. At a certain time, the party opens the envelopes, and the good is sold to the highest bidder at the second highest price. Under certain conditions, these three auction formats can be shown to produce an identical revenue for the seller.

As a result of this explosion in popularity, more and more companies are offering auction sites. Because of this, if you want to purchase a particular good, there are often many auction sites which are offering it. If you really want to get the best price, you must monitor all of these auctions using your web browser, and place bids appropriately. Care must be taken, to ensure you don't make more than one purchase! If there are a large number of auctions, this can be quite a daunting task, requiring your undivided attention for a period of time. Furthermore, if you wish to purchase more than one item, (as is often the case in B to B trading,) it becomes almost impossible.

As a result of this, auctions are beginning to offer support tools to make your job easier. Search tools such as Auction Beagle (http://www.auctionbeagle.com/) and Auction Octopus (http://www.auctionoctopus.com/) allow you to locate and monitor auctions selling specific goods, thus eliminating the need to have 20 browsers open at once. Auction Rover (http://www.auctionrover.com/) provides price trend information on popular items, to aid you in deciding what maximum price to bid. These help, but still leave a large amount of work for you to do. Particularly in the B-to-B context, this can result in an unacceptable overhead. To eliminate this, it is necessary to design automated agents able to carry out the task on your behalf.

Some sites running English auctions, such as Auction Sales (http://www.auction-sales.com/), do offer a simple bidding agent. This agent resides on the auction site, and bids on your behalf. You enter the maximum you are willing to pay, and it places the lowest possible bid on the web site (Either the reservation price, or if bidding has already started, it bids just above the current highest bid.) If all bidders in an auction use such an agent, the auction becomes a Vickrey style auction instead of an English auction. (The sale is made at second price plus the minimum bid increment.)

However, such agents are not able to participate in multiple auctions, either on the same web site or across different ones. As a result, once you use the agent, you are committed to making a purchase on the site if you can. Locally, you may pay the lowest price you can to win the

auction. However, from a global perspective, that particular auction is unlikely to have been the best place to make a purchase.
These bidding agents have an additional disadvantage. To use them, you must reveal the highest price you are willing to pay to the auction site. This gives the auction site information that could be used to cheat you. Furthermore, as auction sites often receive percentage commission on sales made, they also have an incentive to cheat you. To do this, they would take note of the highest price you are willing to pay, and enter a mythical bid in the auction (a 'shill') just under this price. Your agent would then place your maximum bid, and the seller has made a sale to you at the best possible price (assuming you are the highest bidder.) The auctioneer would therefore collect their maximum possible commission.[1]
Hence, neither the auction search facilities nor the existing bidding agents provided by current technology are adequate to meet the needs of a B-to-B auction trader. In this paper, we propose a solution to this problem. We present algorithms that can be used by an agent to participate in multiple auctions on behalf of a trader, and can lead to optimal or near-optimal purchase decisions being made.
The agent aims to purchase one or more identical goods on behalf of its user. It can participate in many auctions for this good, and coordinates bids across them to hold the lowest bids. As auctions progress, and it is outbid, it may bid in the same auction or choose to place a bid in a different auction. The algorithm consists of two parts. Firstly, it has a coordination component, which ensures it has the lowest leading bids possible to purchase the appropriate number of goods. Secondly, it has a belief-based learning and utility analysis component, to determine if it should deliberately 'lose' an auction in the hope of doing better in another auction later.

3 The Coordination Algorithm

The basic algorithm uses the coordination aspect only. If all auctions terminate simultaneously, it will place optimal bids, and make the required purchase at the lowest price. However, if some auctions terminate later than others, and new auctions may come into being, then its behaviour can be non-optimal, and needs to be supported by belief-based and economic techniques, to be discussed in the next section.
An *auction house* may run one or more *auctions* for a given good. Each auction a_i offers $n(a_i)$ goods for sale. Auctions are assumed to be English auctions in format, with *bidders* placing bids at the price they are currently willing to pay for the good. A bidder may place more than one bid in a given auction. The $n(a_i)$ goods offered in the auction are sold to the bidders making the $n(a_i)$ highest bids, for the price they bid. In case of two equal bids, the item goes to the earliest bidder. Hence the auction is *discriminatory* – some buyers will pay more than others for the same good. Different auctions impose different rules covering how a bid may be entered or retracted. For the purposes of this paper, we assume that a buyer may not retract a bid, and a buyer may enter a bid provided it is at least a certain minimal increment δ above the $n(a_i)$th highest bid.
We will define the algorithm, and simultaneously present an example to illustrate the definitions. The illustrative paragraphs, provided for clarification, are indented.
Our agent participates in many auctions selling similar goods, spread out between many auction houses. It wishes to purchase m goods in these auctions, and is given a valuation of v on

[1] While this is possible, I make no claims that it actually happens.

each good by its user. To do this, it monitors the set of auctions currently progressing. For each auction a_i, it observes the n_i highest bids. In other words, it observes the values of the bids which, if the auction terminated immediately, would result in a successful purchase. Let these bids be labelled $\{b_1^i,..,b_{ni}^i\}$, where b_1^i is the highest bid in the auction, and b_{ni}^i is the lowest bid which would currently succeed. We refer to these as the currently *active* bids. To represent the reservation price r, we assume that the seller places n_i bids of value r-δ, where δ is the minimum bid increment.

Let us assume our agent is attempting to purchase 6 discount PCs of a given specification. There are 3 auction houses, each running one auction to sell PCs of this specification. Auction a_1 is selling 4 PCs, auction a_2 is selling 3 PCs and auction a_3 is selling 2 PCs. Assume all auctions have a minimum bid increment of 1.

Auction a_1 currently has the following bids registered (Starred bids are by our agent);

180 180 160** 150 110** 110 100 100

As the auction is for 4 items, the agent observes the 4 highest bids, $\{b_1^1=180, b_2^1 = 180, b_3^1 = 160, b_4^1 = 150\}$

Auction a_2 has the following bids registered;

160** 155 150 100 100

The agent observes the bids $\{b_1^2=160, b_2^2 = 150, b_3^2 = 150\}$

Finally, auction a_3 has just opened, and has no bids. It has a reservation price of 100. The agent represents this by observing the fictitious bids $\{b_1^3=99, b_2^3 = 99\}$ by the seller.

Our agent holds 2 active bids, and so needs to place bids to gain an additional 4.

Let L be the number of currently active bids that are held by our agent. (Initially, L will be zero.) To ensure it makes m purchases, it needs to make new bids that result in it having an additional (m-L) active bids. As we shall see, this may require it to make more than (m-L) bids, as it may need to outbid itself.

If the agent is to hold j active bids in auction a_i, it must place bids that beat the lowest j of the currently active bids. We define the *beatable-j list* for auction a_i to be the ordered set of these bids – namely bids $\{b_{ni-j+1}^i,..,b_{ni}^i\}$. To beat the bids in this list, the agent must place j bids of value $b_{ni-j+1}^i+\delta$ where δ is the minimum bidding increment. The *incremental cost* to the agent of placing these bids, if successful, above the cost that it would have incurred in auction a_i previously, is $j*b_{ni-j+1}^i+\delta$ - {sum of previous bids in a_i}. The beatable-0 list of any auction is defined to be the empty set, and has incremental cost of zero. Obviously, an auction for q goods has no beatable-j lists for j>q.

In the above example, the beatable-1list of auction a_1 is {150}, with incremental cost 151. The beatable-2 list is {160, 150}, with incremental cost 162 (because the agent already holds the bid at 160). Similarly the beatable-3 list is {180, 160, 150} with incremental cost 383, and the beatable-4 list is {180, 180, 160, 150} with an incremental cost of 564.

The agent now constructs potential *bid sets*. A bid set is a set of beatable-j lists that satisfies the following criteria;

1. The set contains exactly one beatable-j list from each auction. (In the terminology of Reiter and the model-based diagnosis community, the set constructed is a *hitting set* of the sets of all beatable-j lists associated with each auction.)
2. The beatable-j lists contain, in total, exactly (m-L) bids made by parties other than our agent.

In other words, each bid set represents one possible way of placing bids to ensure that our agent will gain an additional (m-L) active bids, and therefore will hold exactly m active bids. We define the incremental cost of each of these bid sets to be the sum of the incremental costs of the beatable-j lists in it.

> Returning to our example, our agent needs to find bid sets containing exactly 4 bids made by parties other than it. An example bid set satisfying the above criteria would be;
> [{180, 160,150}, {150}, {99}]
> This set is made from the beatable-3 list of auction a_1, and the beatable-1 lists of auctions a_2 and a_3. Its incremental cost is 634.

The agent must generate the bid set with the lowest incremental cost. In addition, it must avoid generating bid sets that contain a bid equal to or greater than its valuation of the good, v. Various algorithms can be used to do this. The simplest is to generate all possible bid sets, filter out those containing bids greater than v, and select the one with lowest cost. However, this is clearly inefficient, and alternative more intelligent search techniques would be better. Discussion of such algorithms is beyond the scope of this paper.
If there is more than one bid set with identically lowest cost, the agent chooses one arbitrarily. If no such bid sets exist, the agent relaxes condition 2 and finds the smallest i such that at least one bid set exists which contains (m-L-i) bids made by parties other than the agent. Given this i, the agent chooses the bid set with the lowest incremental cost.
Having generated the bid set with the lowest cost, the agent places bids in each auction. For each beatable-j list $\{b_{ni-j+1}^{i},..,b_{ni}^{i}\}$ in the bid set, the agent places j bids of value $b_{ni-j+1}^{i}+\delta$ in the corresponding auction a_i.

> In our example, the bid set with lowest cost is [{150} , {150}, {99,99}]
> This set has cost 502. The agent therefore places one bid of 151 in each of auctions a_1 and a_2, and two bids of 100 in auction a_3, and thereby gains 6 active bids.

The agent continues to monitor the auction, and repeats its analysis if other parties place new bids. In this way, the agent ensures it maintains m active bids at the least possible cost to itself, unless doing so requires it to place bids above its valuation of the good. Providing all auctions terminate simultaneously, this will result in it buying the goods at the best price possible, given the competition in each auction.

4 Dealing with Different Auction Deadlines

Now, we consider the case where auctions terminate at different times. In such a situation, the algorithm above will not necessarily behave optimally. Imagine a situation where an auction starts every half hour, and lasts for an hour. The agent would always monitor two auctions, one which is nearer closing than the other. Inevitably, bids will be higher in the auction which is nearing completion. Hence the agent would switch bidding to the newer auction, and withdraw from the auction about to close. If this continued, the agent would never make a purchase, but would simply switch bids to a new auction every half hour.

The agent needs a mechanism for determining whether to remain in an auction which is about to close, even when there are other auctions with lower current bid prices. To do this, it must be able to make a trade-off in terms of expected value between the relative certainty of remaining in an auction about to close, against the risk of participating in a newer auction. The newer auction may result in a lower purchase, or may result in a far higher purchase price above the agent's valuation of the good. In this section, we propose a mechanism for doing this.

The mechanism we use combines belief-based learning with utility theory. The agent uses belief-based learning to build a model of the spread of valuations held by participants in different auction houses. Then, based on its beliefs about these valuations, it calculates the utility of likely participation in persisting auctions, and compares this with the certain outcome in the terminating auction. If the terminating auction has a higher utility, it remains a participant and makes the purchase. If the remaining auctions have higher expected utility, it withdraws from the terminating auction and continues participation elsewhere.

4.1 The belief-based learning mechanism

The agent generates a model of the potential outcome of auctions by creating a model of each auction house. For a given auction house and a given type of good, it creates a belief function B(x,y) representing the probability that x bidders value the good with a valuation greater than y in a given auction for that good. It builds up this function by monitoring auctions for the good conducted by the auction house. Various possible belief learning techniques can be used to generate this function. For the purposes of this paper, we present a simple example based on that used in [21].

The belief function is defined iteratively – The initial beliefs after one auction $B_1(x,y)$ are defined, and the beliefs after the t+1th auction $B_{t+1}(x,y)$ are defined in terms of the beliefs $B_t(x,y)$ held prior to the auction.

$B_1(x,y)$ = 1 if x or more bidders have placed a bid of y or greater in the first auction observed, 0 otherwise.

$B_{t+1}(x,y)$ = $((t\ B_t(x,y) + 1) / (t + 1)$ if x or more bidders have placed a bid of y or greater in the t+1th auction observed,
= $t\ B_t(x,y) / (t + 1)$ otherwise

More sophisticated versions, for example giving more weight to recent auctions or using the statistical switching technique of [21], may be appropriate depending on the learning environment. However, discussion of these is beyond the scope of this paper.
Using this function, we can estimate the probability that a bid of a certain value will be successful in an auction by a given auction house. Consider an auction for n goods, in which our agent wishes to purchase one. The probability that a bid of x by our agent will be successful can be estimated to be 1 – B(n,x); i.e. 1 minus the probability that n other bidders are prepared to outbid our agent.
There is a flaw in this model, which must be taken into account if it is to be successful. Unlike a Vickrey or Dutch auction, an English auction reveals nothing about the valuations of successful bidders. In other words, if a bidder makes a successful bid of x, we cannot be sure how much higher they may have been willing to bid. To take account of this, it is necessary to add some kind of heuristic weighting to the belief function – we must increase the value of a successful bid by a certain amount, to reflect this possible willingness to bid higher. One possibility is to add a small random amount to each successful bid. In some domains, it may be possible to use econometric data to determine accurately the range that this should be drawn over, while in other domains it may be necessary to use a heuristic estimate.

4.2 Utility analysis of leaving an auction

We now consider how this belief function can be used to compare the expected payoff of an auction which is about to terminate with the less certain outcome of other auctions which terminate later. For the sake of clarity and brevity, we present the technique assuming our agent wishes to purchase a single good.
The expected payoff from the terminating auction is simple to calculate. Assuming our agent is holding an active bid x, or is able to place one at the last moment, then the payoff will be (v-x). If the agent is unable to place a bid because all active bids are beyond its valuation of the good, then payoff will be zero and the agent is forced to participate in other auctions.

> Again, we will introduce an example to illustrate the principles being presented. Assume our agent is purchasing one good from one of three auctions. Auction a_1 is nearing completion, while auctions a_2 and a_3 are continuing. Each auction is for 2 goods, and are run by separate auction houses. The active bids are as follows;
>
> Auction a_1: 140 130
> Auction a_2: 115 110
> Auction a_3: 100 95
>
> Our agent values the good at 200, so could continue bidding in auction a_1. However, should it, or should it switch to the other auctions where prices are lower?

The expected payoff of continuing to participate in the non-terminating auctions is more complex to calculate. To do this, we use the belief function to calculate the probability our agent will be able to make a purchase at various possible bid prices. Recall that, for a given bid price x, the probability our agent will make a successful bid in an auction run by a given auction

house is 1-B(n,x), where n is the number of goods being sold. Similarly, the probability that our agent will be able to make a successful bid at a lower price, x-1, is 1-B(n,x-1). Hence, the probability that our agent will succeed with a bid of x and no lower is B(n,x-1)-B(n,x). The utility of this outcome will be (v-x). Hence, we can calculate the expected utility of participating in a given auction as;

$$\sum_{x=1 \text{ to } v} [B(n,x-1)-B(n,x)](v-x)$$

Of course, as the auction may already be in progress, it is necessary to take into account the current active bids in that auction. The general belief function for the auction house is therefore adapted for this particular auction by setting B(n,x) = 1 if there are n active bids at or over x.

To return to our example, let us assume that our agent has built up the following belief function for the auction house running auction a_2;

Bid price x:	105	110	115	120	125	130	135	140	145	
B(2,x):		1	0.8	0.7	0.6	0.6	0.5	0.3	0.3	0

As there are already bids of 110 and 115, B(2,110) becomes set to 1 for this auction.

As our agent has a valuation of 200, we can calculate the expected utility of this auction to be;

$$\sum_{x=1 \text{ to } v} [B(n,x-1)-B(n,x)](v-x) = (0.3*85)+(0.1*80)+(0.1*70)+(0.2*65)+(0.3*55) = 63.7$$

Given an expected utility on the remaining auctions, the agent must decide whether to place higher bids in the auction which is about to terminate, or withdraw from it. If we assume that the agent is risk neutral, then it will be willing to bid up to a value where the actual utility of the terminating auction is the same as the highest expected utility among the remaining auctions. In other words, it is prepared to make a maximum bid b_{max} of;

$$b_{max} = v - \sum_{x=1 \text{ to } v} [B(n,x-1)-B(n,x)](v-x)$$

In our example, assuming that a_3 has a lower expected utility than a_2, our agent will be willing to bid up to (200-63.7) = 136.3 in auction a_1. Hence, it will place a bid of 135, and will withdraw if this is outbid. In this case, it will hope to make a better purchase in auctions a_2 or a_3.

In this way, the agent is able to make informed decisions about whether to continue bidding in an auction or to switch. If it is making multiple purchases, it may purchase some in the terminating auction, and choose to switch others to continuing auctions. Extensions of the algorithm to handle this case will be dealt with in a future paper.

5 Related Work

Research into automated negotiation has long been an important part of distributed AI and multi-agent systems. Initially it focused primarily on negotiation in collaborative problem solving, as a means towards improving coordination of multiple agents working together on a common task. Laasri, Lassri, Lander and Lesser [5] provide an overview of the pioneering work in this area. As electronic commerce became increasingly important, the work expanded to encompass situations with agents representing individuals or businesses with potentially conflicting interests. The contract net [6] provides an early architecture for the distribution of contracts and subcontracts to suppliers. It uses a form of distributed request-for-proposals. However, it does not discuss algorithms for determining what price to ask in a proposal. Jennings et.al. [7] use a more sophisticated negotiation protocol to allow the subcontracting of aspects of a business process to third parties. This is primarily treated as a one-to-one negotiation problem, and various heuristic algorithms for negotiation in this context are discussed in [8]. Vulkan and Jennings [9] recast the problem as a one-to-many negotiation, and provide an appropriate negotiation protocol to handle this. Other relevant work in one-to-one negotiation includes the game-theoretic approach of [10] and the logic-based argumentation approach of [11].

As much electronic commerce involves one-to-many or many-to-many negotiation, the work in the agent community has broadened to explore these cases too. The Michigan AuctionBot [12] provides an automated auction house for experimentation with bidding algorithms. The Spanish Fishmarket [13] provides a sophisticated platform and problem specifications for comparison of different bidding strategies in a Dutch auction, where a variety of lots are offered sequentially. The Kasbah system [14] featured agents involved in many-to-many negotiations to make purchases on behalf of their users. However, the algorithm used by the agents (a simple version of those in [8]) was more appropriate in one-to-one negotiation, and so gave rise to some counter-intuitive behaviours by the agents. [15] and [16] present adaptive agents able to effectively bid in many-to-many marketplaces. [17] demonstrates how these can be used to produce a market mechanism with desirable properties. Park et.al. [18][19] present a stochastic-based algorithm for use in the University of Michigan Digital Library, another many-to-many market.

Gjerstad et. al. [20] use a belief-based modeling approach to generating appropriate bids in a double auction. Their work is close in spirit to ours, in that it combines belief-based learning of individual agents bidding strategies with utility analysis. However, it is applied to a single double auction marketplace, and does not allow agents to bid in a variety of auctions. Vulkan et.al. [21] use a more sophisticated learning mechanism which combines belief-based learning with reinforcement learning. Again, the context for this is a single double auction marketplace. Unlike Gjerstad's approach, this focuses on learning the distribution of the equilibrium price. Finally, the work of Garcia et.al. [22] is clearly relevant. They consider the development of bidding strategies in the context of the Spanish fishmarket tournament. Agents compete in a sequence of Dutch auctions, and use a combination of utility modeling and fuzzy heuristics to generate their bidding strategy. Their work focuses on Dutch rather than English auctions, and on a sequence of auctions run by a single auction house rather than parallel auctions run by multiple auction houses. However, the insights they have developed may be applicable in our domain also. We hope to investigate this further in the future.

6 Conclusions and Future Work

By interleaving the application of two algorithms of the form described above, our agent can effectively participate in multiple English auctions. It will use the coordination algorithm to place lowest possible bids across auctions. It will use the bid withdrawal algorithm to determine when it is worth bidding higher in an auction which is about to terminate, as opposed to transferring to other auctions where the active bids are currently lower.
In this paper, we have sketched out the structure of appropriate algorithms to do this. The specifics of these algorithms, particularly the bid withdrawal mechanism, may need refinement and specialisation to operate in specific market applications. Furthermore, the richness of the model may be increased. Specifically;

- A more sophisticated belief mechanism could be used. This may be generic, or could be tailored to the specific dynamics of a particular market.
- The algorithm presented assumes that the buyer is risk-neutral. This should be generalised to allow the agent to adopt other risk attitudes as appropriate.
- The algorithm does not take into account time discounting or deadlines. Again, this should be generalised.
- The algorithm assumes that the agent values all goods equally. This should be generalised to allow the agent to receive a demand curve from its user.
- The algorithm assumes that the beliefs about auctions are accurate. By measuring the deviation of actual auctions from the predictions, it would be possible to give a measure of confidence in the belief. This could be used to moderate the agent's decision to switch auctions, taking into account the agent's attitude to risk.
- The algorithm only takes into account the expected payoff of existing auctions. It may be appropriate to also model the possibility that an auction house may bring new auctions into being, and the potential payoff of such auctions.
- Throughout this paper, we have assumed that all auctions are English in format. Research is required to generalise this to cover other forms of auction, such as Dutch, Vickrey and Double auctions.

We hope to address some of these issues in a future paper.

References

1. R. Kalakota and A. Whinston. Frontiers of Electronic Commerce.. Addison-Wesley 1996.
2. C. Shapiro and H. Varian. Information Rules – A Strategic Guide to the Network Economy. Harvard business school press, 1999.
3. K. Binmore Fun and Games.. D. Heath and Co. 1992.
4. Paul Klemperer, "Auction theory: a guide to the literature" J. of Econ. Surveys, Vol. 13, No. 3, July 1999, pp. 227-286
5. B.Laasri, H.Laasri, S. Lander and V. Lesser. A Generic Model for Intelligent Negotiating Agents. International Journal of Intelligent and Cooperative Information Systems, 1(2) 1992 pp 291-317.

6. R.G. Smith . The contract net protocol: high-level communication and control in a distributed problem solver. IEEE Trans. Comput., 29, 1104-1113, 1980.
7. N.R. Jennings, P. Faratin, M.J. Johnson, P.O. O'Brien and M.E. Wiegand. Using Intelligent Agents to Manage Business Processes. In Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM-96), 345-360, April 1996
8. P. Faratin, C. Sierra and N. Jennings. Negotiation decision functions for autonomous agents. Robotics and Autonomous Systems 24(3-4),1998, pp159-182.
9. N.Vulkan and N.Jennings. Efficient Mechanisms for the Supply of Services in Multi-Agent Environments, Proceedings of the 1st International Conference on the Internet, Computing and Economics, ACM Press 1998.
10. J. Rosenschein and G. Zlotkin. Rules of Encounter. MIT Press, 1994.
11. S. Parsons, C.Sierra and N.Jennings. Agents that reason and negotiate by arguing.
12. P.Wurman, M.Wellman and W.Walsh. The Michigan Internet AuctionBot: A Configurable Auction Server for Human and Software Agents. In Proc. Second Conference on Autonomous Agents, 1998.
13. J. Rodriquez-Aguilar, P. Noriega, C. Sierra and J.Padget. Fm96.5: A Java-based e,ectronic auction house. Proc. Second International Conference on the Practical Application if Intelligent Agents and Multi-Agent Systems, 1997, pp207-224.
14. A. Chavez, D.Dreilinger, R.Guttman and P.Maes. A Real-Life Experiment in creating an Agent Marketplace. Proc. Second International Conference on the Practical Application if Intelligent Agents and Multi-Agent Systems, 1997.
15. Cliff, D. and Bruten, J. Less than Human: Simple adaptive trading agents for CDA markets. Proceedings of the 1998 Symposium on Computation in Economics, Finance, and Engineering: Economic Systems
16. Preist, C. and van Tol, M. Adaptive Agents in a Persistent Shout Double Auction. Proceedings of the 1st International Conference on the Internet, Computing and Economics, ACM Press 1998.
17. C.Preist. Commodity Trading using an Agent-Based iterated Double Auction. Proc. Third Conference on Autonomous Agents, 1999.
18. S. Park, E.Durfee and W.Birmingham. Emergent Properties of a Market-Based Digital Library with Strategic Agents. Proc. International Conference on Multi Agent Systems, 1998.
19. S. Park, E.Durfee and W.Birmingham. An Adaptive Agent Bidding Strategy based on Stochastic Modelling. Proc. Third Conference on Autonomous Agents, 1999.
20. Gjerstad, S. and Dickhaut, J. Price formation in double auctions. Games and Economic Behaviour, 22(1), pp1-29, 1998.
21. N. Vulkan and C. Preist. Automated Trading in Agents-based Markets for Communication Bandwidth. Proc. UKMAS (1999.
22. P. Garcia, E.Giminez, L. Godo and J. Rodriguez-Aguilar. Possibilistic-based design of bidding strategies in Electronic Auctions. Proc. 13th Biennial European Conference on Artificial Intelligence, 1998.