Using Sampling To Compute Bayes-Nash Equilibrium In Auction Games

[CPSC 532A Course Project] *

Roman Holenstein
Department of Computer Science
University of British Columbia
Vancouver, BC
romanh@cs.ubc.ca

ABSTRACT
The use of sampling is investigated for computing equilibrium bidding strategies in auctions. An algorithm is proposed that requires minimal assumptions on the agents. In this paper we concentrate on asymmetric auctions with independent bidder valuations, however the approach is extendable to other scenarios, for example having bidders with different risk attitudes. Results are presented and the performance of the algorithm is discussed.

General Terms
Economics, theory

Keywords
Game theory, auctions, sampling, Monte Carlo

1. INTRODUCTION
Auctions have been a very popular mechanism for the trading of goods or allocation of resources. There are many different types of auctions, e.g. English, Dutch, or Japanese. The different types are designed to optimise some measure(s), such as welfare or efficiency [6, 12]. The buyers (agents) bid according to a strategy that maximises their utility.

In order to analyse this setting we use game theory. In that framework, an auction can be represented as a single-shot Bayesian game[1] One problem is to find the equilibrium strategy of the agents. In an imperfect information game, such as auctions, one such equilibrium is the Bayes-Nash equilibrium (BNE). Computing a BNE is generally very difficult. In order to find a solution the setting is often constrained by making simplifying assumptions.

Tools are available to help solving these problems, such as Gambit[2] and Gala [5]. These are able to solve finite extensive and normal form games, however, they are not designed to handle the general or more complex cases.

Various methods have been developed to find, or at least approximate, the BNEs. Sometimes the problem can be represented as a linear program (e.g. simple 2 person games). The game may be simplified by representing the problem in a more compact form [1, 2, 13]. Sampling has also been used. Cai et al. employ Monte Carlo in sequential auctions to sample the valuation space of the other agents and then use Gambit to solve the resulting complete information game [2]. Walsh et al. solves for a BNE using a payoff matrix of heuristic strategy profiles, where the average payoffs (ex-ante) are approximated by sampling from the competing buyers' valuations.

Simple auctions can be solved analytically, or by using a tool like Gambit. However, by relaxing some simplifying assumptions the problem of finding BNEs can easily become unmanageable for these methods. For example:

- introduce an entry cost
- asymmetric auction (different valuation distribution functions for agents)
- different risk attitudes for different bidders (e.g. constant absolute risk aversion such as log with constant factor or different bases)
- having k repeated auctions

A lot of research has been done on auctions, most assume symmetric information. In practice, agents often draw their valuations from different distributions. This is the case for

*This project was for the course *Multiagent Systems* taught by Kevin Leyton-Brown. Further details on the course are available at `www.cs.ubc.ca/∼kevinlb/teaching/cs532a%20-%202005`

[1]Sequential auctions can be represented as extensive form games [2].

[2]Gambit is a library of game theory software and tools for the construction and analysis of finite extensive and normal form games. See http://econweb.tamu.edu/gambit/.

example in contract bidding. Each of the bidders has a different opportunity cost for completing the project. In recent years some work has been done on relaxing that assumption [3, 8, 9, 10]. Many studies focus on expected revenue [4, 11]. Campoo et al. [3] and Riley et al. [8, 9, 10] formalise methods to characterise the equilibrium bidding strategies and revenue of the asymmetric auction. Often some (simplifying) assumptions are made, such as limit the number of bidders to two or assume translation invariant bid functions [7].

2. PROBLEM

For this study we focus on sealed-bid first-price auctions with bidders having asymmetric information about the good for sale (asymmetric auction). The agents do not know each other's exact valuation (type), but know the distributions from which they are drawn. We also assume that the valuations are independent and the type space is continuous. The valuation probability distribution is allowed to differ between the agents (asymmetry) and no restriction is placed on the functional form of the distributions.

In a sealed-bid auction, each agent simultaneously submits a single bid for for the item. We assume that all bidders participate in the auction, have quasi-linear utility and are risk-neutral. The goal of this project was to develop a sampling based approach for finding best-response strategy profiles or a good approximations to the BNEs.

3. APPROACH

Sampling is a relatively new approach to solve BNEs in auction games. A few articles are available, such as work by Cai et al.[2], which concentrates on sequential multi-unit auctions. Cai, however, relies on Gambit to solve parts of the problem, which can be a bottleneck when considering larger problems.

The approach taken in this project is based on Monte Carlo sampling and simulated annealing. This algorithm is designed to find a myopic best response. Provided that it converges, the solution found will be a Bayes-Nash equilibrium. The pseudo-code is given in Algorithm 1. First the strategies are initialised to truthful bidding. A different initial strategy profile may be chosen, for example one that is closer to the desired equilibrium. Then each agent in turn attempts to improve his expected utility for a sample chosen from its valuation space. Given his type, the agent now computes a best response to the other agents' current strategies and then updates his bidding strategy accordingly. The details on the computation of the best response and the method for updating the strategy profile are described in the next sections.

3.1 Computing a Best Response

Finding a best response requires computing the expected utility for each agent given the other agents' strategy, which is then to be maximised. The utility computation involves evaluating an integral of the form

$$\mathcal{E}(u_i(b, s_i, s_{-i})) = \int u_i(b, s_i, s_{-i})\, P(v_{-i}|v_i)\, dv_{-i} \qquad (1)$$

where u is the utility, v is the valuation, and s is the strategy. The subscripts i and $-i$ denote agent i and the other agent(s), respectively. P is the joint probability of the other agents' valuations given v_i (type). Note that for this project we have assumed that the valuations are independent, thus $P(v_{-i}|v_i) = P(v_{-i})$. This integral may not be analytically solvable, e.g. in the case of complicated valuation distribution functions for the agents.

Algorithm 1: Pseudo code of sampling-based algorithm.

```
set t = 0 and temperature T_0 = 1;
initialise strategy of each agent to truthful bidding;
while strategies changing (by more than some threshold)
do
    forall agents i do
        sample a valuation from i's type space;
        compute best response given other agents' current
        strategy;
        update i's strategy profile;
        t = t + 1;
    end
    set T_{t+1} according to cooling schedule
end
```

The maximisation of the expected utility is solved using simulated annealing. The pseudo-code is shown in Algorithm 2. We sample from the action space, i.e. bids, using a proposal distribution q and evaluate the expected utility for that choice. The fitness of that sample is evaluated by comparing the utilities. The samples are generated using a Markov-Chain symmetric random walk using a Gaussian distribution function q centered at the last accepted sample. The simulated annealing is implemented as a continuous process in the main loop shown in Algorithm 1. A decreasing exponential is used as a cooling schedule. As the temperature drops, the chance of choosing bids that generate lesser utility decreases. Care must be taken not to lower the temperature too fast, as the system may get stuck in a local minima. However, if the cooling rate is too slow, then the algorithm may take a long time to converge, or possibly not converge at all, depending on the equilibrium property. The optimal cooling schedule depends on the specific problem to be solved.

Algorithm 2: Pseudo code for computing best response.

initialise b^0;
compute expected utility: $u^0 = \mathcal{E}[u_i(b^0, v_i, v_{-i})]$;
for *k = 0 to N-1* **do**
  Sample $z \sim \mathcal{U}_{[0,1]}$;
  Sample $b^* \sim q(b^*|b^{(k)})$;
  compute expected utility $u^* = \mathcal{E}[u_i(b^*, v_i, v_{-i})]$;
  if $z < \mathcal{A}(b^k, b^*) = min\left\{1, \left(\frac{u^*}{u^k}\right)^{\frac{1}{T_t}}\right\}$ **then**
    $b^{(k+1)} = b^*$;
    $u^{(k+1)} = u^*$;
  else
    $b^{(k+1)} = b^k$;
    $u^{(k+1)} = u^k$;
  end
end

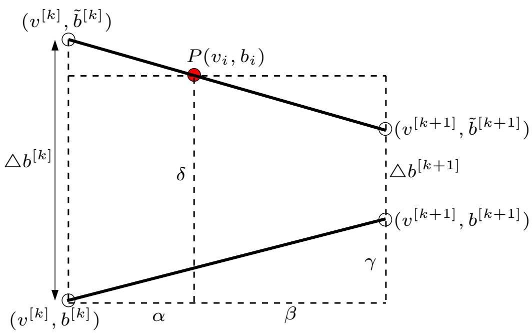

Figure 1: Updating of PWL function.

3.2 Updating the Bidding Strategy

The bidding strategy is represented as a piecewise linear (PWL) function. After finding a best response b_i for a valuation v_i, we want this function to pass through that point. This is illustrated by the diagram shown in Figure 1. The solid (red) point P at (b_i, v_i) represents new best response. The points superscripted by $[k]$ and $[k+1]$ denote subsequent nodes in the PWL. The lower solid thick line is the current profile, and the upper line is the updated one, passing through P. We have chosen the bid adjustments $\triangle b^{[k]}$ and $\triangle b^{[k+1]}$ to be in proportion to how close the new point is to the node, i.e. $\alpha \triangle b^{[k]} = \beta \triangle b^{[k+1]}$. In order to damp any oscillatory behaviour, we scale the adjustment by a constant factor $\lambda \in [0,1]$, so the new node points $\hat{b}^{[k]}$ and $\hat{b}^{[k+1]}$ are:

$$\hat{b}^{[k]} = b^{[k]} + \lambda \triangle b^{[k]} \quad (2)$$

$$\hat{b}^{[k+1]} = b^{[k+1]} + \lambda \triangle b^{[k+1]} \quad (3)$$

4. RESULTS

In order to test our technique we compute the BNE of a symmetric auction for which the equilibrium can be computed. Next a simple asymmetric auction that can be solved analytically serves as a second benchmark.

The algorithm described above was implemented in C++[3] and the computations were performed on a 2.4 GHz Pentium 4 running Linux 2.4. A typical calculation takes on the order of a few minutes to half an hour to finish, depending on the desired accuracy, resolution and number of agents.

4.1 Symmetric Auction

We first consider a case with two agents, both having uniform valuation distributions in the range $[0,1]$. The Bayes-Nash equilibrium is for both to bid half their valuation. In general, if there are N participating agents, each bids the expected second price assuming his bid is the highest:

$$b_i = \frac{n-1}{n} v_i \quad (4)$$

The result for the two agents is shown in Figure 2 and took approximately 3 minutes to compute. 1000 main cycles were

[3]The source code is available at
`www.cs.ubc.ca/~romanh/courses/cpsc532A/auction/`

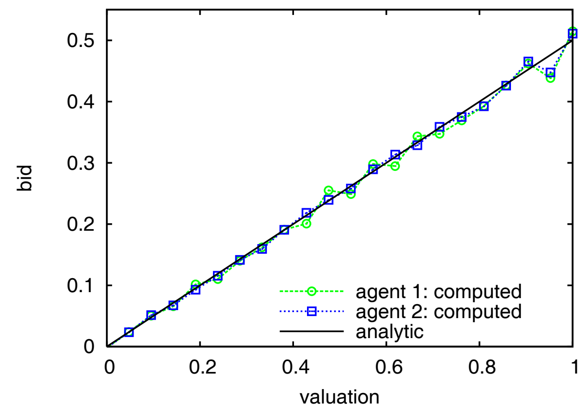

Figure 2: The computed and analytic equilibrium strategies for 2 players with valuations uniformly drawn from $[0,1]$. Both agents have the same strategy profile.

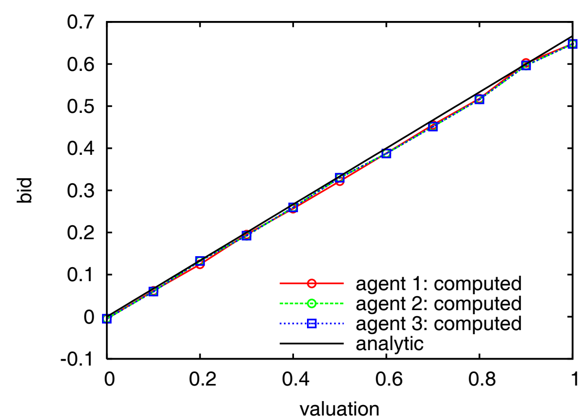

Figure 3: The computed and analytic equilibrium strategies for 3 players with valuations uniformly drawn from $[0,1]$.

required and we used 3000 samples for each utility computation. The best response at each cycle was selected from 50 bid samples. The temperature was decreased exponentially with a half-life of 100 main cycles. The obtained result agrees reasonably well with the true BNE. The results for the three agent auction, using the same valuation distribution, is shown in Figure 3. The result shows good agreement with the analytical strategy profile.

4.2 Asymmetric Auction

For the asymmetric auction, our test case has two agents with valuations drawn from uniform distributions $\mathcal{U}_i$ that are overlapping but with different bounds. We chose $\mathcal{U}_1 = \mathcal{U}_{[0,4/3]}$ and $\mathcal{U}_1 = \mathcal{U}_{[0,4/5]}$. The equilibrium in this case is for the

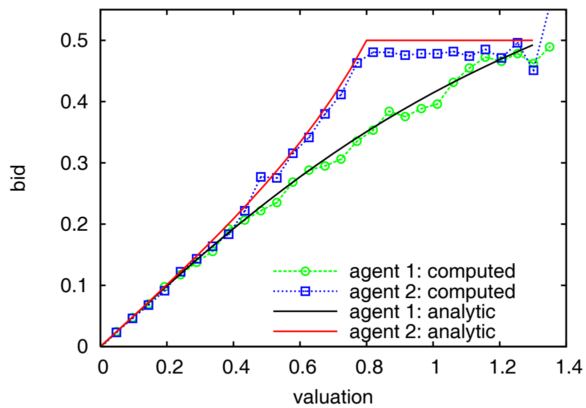

Figure 4: Results for asymmetric auction using valuations drawn from $\mathcal{U}_{[0,4/3]}$ and $\mathcal{U}_{[0,4/5]}$

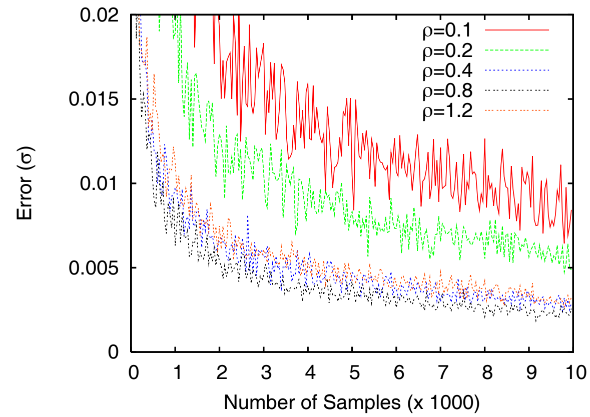

Figure 5: The standard deviation is plotted vs. the number of samples used to compute the utility. The rate of convergence depends on the shape of the proposal distribution, in this case the width of the gaussian. The agents' valuations were drawn from a Gaussian distribution with mean of 2 and a standard deviation of 0.5.

agents to bid (see Appendix A)[4]:

$$\begin{aligned} b_1(v_1) &= \tfrac{1}{v_1}(\sqrt{1+v_1^2}-1) \\ b_2(v_2) &= \tfrac{1}{v_2}(1-\sqrt{1-v_2^2}) \end{aligned} \qquad (5)$$

The computed result after 5000 iterations is shown inFigure 5. The result is less convincing than in the symmetric case. Some of the data points do not line up with the equilibrium, particularly at the plateau in agent 2's bid function. However, the general shape of the profile is found. It is possible that better convergence could be achieved by using more iterations, or the speed of convergence may be improved by optimising the parameters for the sampling.

4.3 Convergence

A large portion of the computational cost is computing the expected utility for a given valuation and bid. Thus it is important that the integral is computed with as few samples as possible. A Gaussian proposal distribution is used for generating the samples, unless the target distribution is uniform, in which case it is sampled directly. The choice of proposal distribution can significantly affect the convergence of the utility computation. This is illustrated in Figure 5. The standard deviation decreases with the number of samples used to compute the utility. The rate of convergence depends strongly on the width of the Gaussian proposal distribution. The standard deviation σ of the proposal distribution is set to $\rho(\overline{v}-\underline{v})/2$, where $\underline{v}$ and $\overline{v}-\underline{v}$ define an approximate lower and upper bound on the agents' valuations, i.e. the range in which the valuations fall with close to 100% probability. The error drops sharply over the first 1000 samples and does not significantly improve after about 3000 samples. For the cases presented in this study we used 3000 samples to compute the utility. The computation was done using single precision (32 bits). A comparison with using double precision gave virtually identical results.

5. SUMMARY

We have proposed a method for finding Bayes-Nash equilibria. The technique uses Monte Carlo sampling and simulated annealing. We were able to approximate the Bayes-Nash equilibrium for a few select example cases. The results agree reasonably well with the known analytic solutions. The current implementation requires on the order of a few minutes up to about one half hour to generate a result of the accuracy presented in this paper. Higher accuracy can be obtained by using more samples at the cost of increased running time.

The approach makes very few assumptions on the agents' valuation distributions. Required are some functional form and approximate bounds on the domain, in order to choose an appropriate proposal distribution for sampling. In this study we have only considered very simple cases and the emphasis was to demonstrate functionality of the algorithm. A possible extension on this work could be to apply this technique to other auctions that drop some of the simplifying assumptions outlined in Section 1. If the algorithm is applied to large auctions with many agents, it may take a long time to reach an equilibrium. However, the algorithm appears to be easily parallelizable, which would allow it to scale well with the number of bidders.

[4]This was an assignment problem in an Economics course taught by John Rust at the University of Maryland.

6. REFERENCES

[1] N. Bhat and K. Leyton-Brown. Computing Nash equilibria of action-graph games. In *Proceedings of the 20th Annual Conference on Uncertainty in Artificial Intelligence (UAI-04)*, pages 35–42, Arlington, Virginia, 2004. AUAI Press.

[2] G. Cai and P. R. Wurman. Monte carlo approximation in incomplete information, sequential auction games. *Decision Support Systems*, 39:153–168, 2005.

[3] S. Campo, I. Perrigne, and Q. Vuong. Asymmetry in first-price auctions with affiliated private values. Technical Report 13, French Institute for Agronomy Research (INRA), Economics Laboratory in Toulouse

(ESR Toulouse), 2000. available at http://ideas.repec.org/p/rea/inrawp/13.html.

[4] E. Cantillon. The effect of bidders' asymmetries of expected revenue in auctions. Technical Report 13, Harvard Business School, Boston (MA), 2004. available at http://www.people.hbs.edu/ecantillon/.

[5] D. Koller and A. Pfeffer. Generating and solving imperfect information games. In *IJCAI*, pages 1185–1193, 1995.

[6] V. Krishna. *Auction Theory*. Academic Press, Elsevier Science, San Diego, CA, 2002.

[7] M. C. Laskowski and R. L. Slonim. An asymptotic solution for sealed bid common-value auctions with bidders having asymmetric information. *Games and Economic Behavior*, 28:238–255, 1999.

[8] H. Li and J. G. Riley. Auction choice. Technical report, UCLA, Business Economics, Los Angeles (CA), 1999. available at http://www.econ.ucla.edu/riley/research/ach_a12.pdf.

[9] E. Maskin and J. Riley. Asymmetric auctions. *Review of Economic Studies*, 67(3):413–438, 2000. available at http://ideas.repec.org/a/bla/restud/v67y2000i3p413-38.html.

[10] E. Maskin and J. Riley. Equilibrium in sealed high bid auctions. *Review of Economic Studies*, 67(3):439–454, 2000. available at http://ideas.repec.org/a/bla/restud/v67y2000i3p438-54.html.

[11] P. Pezanis-Christou. On the impact of low-balling: Experimental results in asymmetric auctions. *International Journal of Game Theory*, 31(1):69–89, 2002. available at http://ideas.repec.org/a/spr/jogath/v31y2002i1p69-89.html.

[12] Y. Shoham and K. Leyton-Brown. *Multi Agent Systems*. unpublished, Vancouver, draft edition, 2004.

[13] W. Zhu and P. R. Wurman. Structural leverage and fictitious play in sequential auctions. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, pages 385–391, Menlo Park, Calif., 2002. AAAI Press.

APPENDIX

A. EQUILIBRIUM IN AN ASYMMETRIC AUCTION

We want to show that the equilibrium bid functions for two agents with valuations drawn from $\mathcal{U}_1 = \mathcal{U}_{[0,4/3]}$ and $\mathcal{U}_1 = \mathcal{U}_{[0,4/5]}$ are given by Eqn. 5. The equilibrium bid function is such that it maximises the utility, so we have

$$\begin{aligned} b_1(v) &= \arg\max_{b_1} (v - b_1) P(b_2(v_2) \leq b_1) && (6) \\ &= \arg\max_{b_1} (v - b_1) P(v_2 \leq b_2^{-1}(b_1) && (7) \\ &= \arg\max_{b_1} (v - b_1) \frac{5}{4} b_2^{-1}(b_1) && (8) \end{aligned}$$

where b_2^{-1} is the inverse bid function of agent 2. Similarly for agent 2, we get

$$b_2 = \arg\max_{b_2} (v - b_2) \frac{3}{4} b_1^{-1}(b_2) \qquad (9)$$

The inverse bid functions are

$$\begin{aligned} b_1^{-1}(b) &= \frac{2b}{1 - b^2} && (10) \\ b_2^{-1}(b) &= \frac{2b}{1 + b^2} && (11) \end{aligned}$$

We now substitute the inverse bid function into Eqn. 8. The maximum bid function is found by differentiating with respect to the bid and equating to zero:

$$\begin{aligned} 0 &= \frac{\partial}{\partial b_1} \left[(b_1 - v) \frac{5}{4} \frac{2b_1}{1 + b_1^2} \right] && (12) \\ 0 &= v b_1^2 - 2b_1 - v && (13) \\ b_1 &= \frac{1}{v_1} \left(\sqrt{1 + v_1^2} - 1 \right) && (14) \end{aligned}$$

And similar for b_2. Indeed we find that the best response is given by Eqn. 5.