Automated Design of Near Optimal Auctions for Realistic Scenarios
[Extended Abstract]

Yuri Rabinovich*

Amir Ronen†

ABSTRACT

Automated construction of revenue maximizing auctions poses many challenges which are not addressed by classic auction theory. In this paper we describe a system that facilitates generic automatic construction of near optimal auctions for realistic scenarios. In order to test our system we execute it on several benchmark distributions. On all of these benchmarks, our system yields revenue higher than the English auction. On some of them, the gap is significant. The system gives initial insights into several issues that have not been explored so far, such as the price of fairness in single item auctions, the power of ascending auctions, etc. Many open questions are presented as well.

Keywords: auctions, mechanism design, systems.

1. INTRODUCTION

In recent years, the usage of auctions has become a prominent method of trade. The applications of auctions are very diverse and range from selling multi billion dollar companies, to trading small articles over the Internet. In economics auction theory is broad and studies any situation in which a selling or a procurement method is to be designed. Thus, auctions play a major role in both economics and electronic commerce. Moreover, they give rise to many fascinating problems on the border of economics and computation. Perhaps the most basic auction design problem is to design revenue maximizing mechanisms for selling one item[1].

The standard setup of optimal auction theory is as follows: there are n potential self-interested buyers termed agents. Each agent i has a *privately known* value v_i for winning the object. v_i represents the agent's maximum willingness to pay for the item. The agent's value is zero if it does not win the auction. Each agent selfishly tries to maximize its **own** utility, i.e., the difference between its value and its payment. An auction is any protocol which decides who wins the item (if at all) and for what price. Given a joint probability distribution ϕ on the values of the agents, the goal is to design an auction that maximizes the *expected revenue* of the seller.

Auction theory focuses on mechanisms that satisfy two major properties: *individual rationality* and *incentive compatibility.* Each of these properties has two major variants. Informally, *ex-post individual rationality* (abbreviated IR) means that only the winner pays for the item and that the winner's payment is bounded by its private value. For many applications, this property is of great importance if not mandatory. In this work we only consider auctions that satisfy this property. In the Bayesian variant of IR, non negative utility is guaranteed to the bidders only if all of them follow a given Bayesian equilibrium. This variant is problematic for many applications (e.g. [8, p. 400]) and is not considered here. *Incentive compatibility* (IC) means that each agent has a dominant strategy, i.e., a strategy which is always optimal for it. An auction satisfies *Bayesian incentive compatibility* if there exists a Bayesian equilibrium in the resulting game of incomplete information. An auction that satisfies both, IC and IR is called *valid.* The celebrated Vickrey auction is an example of a valid auction. Similarly, an auction that satisfies IR and Bayesian IC is called *Bayesian valid.*

The *expected revenue* of a valid auction equals the expected payment of the winner when the agents follow their dominant strategies and their values are drawn from the underlying distribution ϕ. An auction is called *optimal* if its revenue is maximal among all valid auctions. The definitions for Bayesian valid auctions are similar.

Let v_1 denote the maximal value $max(v_1, \ldots, v_2)$ and v_2 denote the second highest one. Note that any auction that satisfies IR (even only in the Bayesian sense), cannot extract a revenue of more than $\mathbf{E}[v_1]$. On the other hand, the standard Vickrey auction has an expected revenue of $\mathbf{E}[v_2]$.

Currently, the vast majority of auctions which are used in practice are variants of the English auction [12]. Such auctions are almost equivalent to either Vickrey auctions or to Vickrey auctions with reserve price, and are by no means optimal. Thus, automated auction design has the potential of significantly impacting the way that people are trading.

1.1 Related Work

Auction design is a major topic in economics. An introduction can be found in recent books by Klemperer[6] and Krishna[7]. Still, notoriously little is known about the design of revenue maximizing auctions, in particular when ex-post IR is required. A seminal paper of Myerson [9] characterizes the optimal valid auction when the agents' values are independent and the distribution of each agent's value

*Faculty of Industrial Engineering & Management, Technion, Israel. Email:byuric@gmail.com

†Faculty of Industrial Engineering & Management, Technion, Israel. The author was supported in part by grant number 969/06 from the Israel Science Foundation. Email:amirr@ie.technion.ac.il

[1]The design of minimal payment procurement mechanisms is similar and the methods described in this paper can be applied to it as well.

is continuous with full support and regular (i.e. the functions $\widetilde{v}_i(x) = x + \frac{1-F_i(x)}{f_i(x)}$ are non decreasing where F_i and f_i denote the cumulative distribution and the density of agent i's type). Little progress has been made for more general probability distributions. Myerson [9] also studies the case where individual rationality is only required in a Bayesian equilibrium (Bayesian IR). He characterizes the optimal auction for this case for general independent distributions. For non independent agents, Crémer and McLean [2] show that under certain conditions on the distribution, it is possible to extract a revenue equal to the expected highest value $\mathbf{E}[v_1]$. Unfortunately, for many applications, Bayesian IR is unacceptable (see e.g. [8, p. 400]). A paper by Neeman shows that English auctions can extract a large portion of the optimal revenue in many natural environments [10].

In computer science, the study of revenue maximization was pioneered by Goldberg, Hartline, and Wright [4] who considered the case of unlimited supply. Since then, most of the work on revenue maximization focuses on this case and provides impressive guarantees against the worst case. (An interesting hybrid approach can be found in [5].) There are fundamental differences between the problem that we consider and the unlimited supply case. In particular, it is impossible to provide any worst case guarantee in our setup and focusing on the average case, which is standard in economics, is more natural [14]. Moreover, the sampling techniques which proved as very powerful in the unlimited case, are hardly relevant to the one item setup.

Several papers by Conitzer and Sandholm study what they call automated mechanism design (e.g. [16]). In these works, the emphasis is on the algorithmic side of mechanism design problems, and on solving specific instances, rather than constructing generic mechanisms. Applications of automated mechanism design include dispute settlement with 2-bidders, Myerson optimal auction for IID distributions[9], optimal combinatorial auctions, and optimal Bayesian mechanisms for public goods problem. As far as we know, all these works deal with very small instances of the corresponding mechanism design problems. In addition, the corresponding type distributions are fully available to the designer. A comprehensive discussion of the emerging literature on automated mechanism design is outside the scope of this exposition. A good description can be found in [1].

1.1.1 KLA auctions

KLA is a generic method for the construction of near optimal valid or Bayesian valid mechanisms [13, 14]. In a nutshell, a KLA-auction is a two stage mechanism. In the first stage, all except the k-highest bidders are rejected. This stage can be implemented either by a revelation auction or by a standard English auction. Let $res = v_{k+1}$ be the highest value of a rejected agent and let $\tilde{\phi}$ be the conditional distribution over the high agents, given the values and identities of the rejected agents. In the second stage, a valid auction which is optimal among all valid auctions with reserve price r, is conducted on the high bidders only.

The KLA method has many virtues. For any distribution ϕ, its expected revenue is at least 1/2 of the revenue of the optimal mechanism. When the distribution is independent, the revenue is at least $\frac{k}{k+1}$ of the optimum. This property is approximately preserved as long as the agents are not strongly dependent in some sense (see [14]). The method can be used to get upper bounds on the optimal revenue by setting the auction's revenue to res whenever no agent wins the item. Finally, it is possible in the second stage of the auction, to optimize over a subfamily of valid mechanisms.

In this work we consider several variants of KLA mechanisms. The first stage of all of these mechanisms is identical. The difference is only in the family of mechanisms over which the second stage auction is chosen. More details can be found in the full version of this paper.

As we shall see, despite its computational efficiency, the implementation of the KLA method requires overcoming several major obstacles.

1.2 Our Contribution

In this paper we describe a system that facilitates generic automatic construction of near optimal auctions for realistic scenarios. We describe many novel challenges that stem from the design of such a system along with our solutions, report on experiments conducted in order to test the system's performance, and present many open questions.

Perhaps the most basic difference between this work and previous work on automated auction design is the focus on the design of the system as a whole. All previous works on automated auction design assume that a full description of the underlying distribution ϕ on the agents' valuation is available to the designer. This assumption is not reasonable for many applications, neither from a cognitive nor from a computational perspective. Therefore, our first step is to provide the designer with an interface by which it can conveniently and feasibly describe the economic environment to the system. Technically, the designer describes a sampleable distribution via an XML file. The system allows usage of sampleable stochastic variables that represent the main factor that determine the agent values, mathematical expressions over these variables, discrete ranges with rounding operations, etc. The system then generates an approximation of ϕ via sampling. In order to sample a type vector, the system samples the basic variables and applies the above expressions on them. This approach is far from being a mere engineering issue. Since such a description is possible only in an approximate fashion, it is not clear at all that results from auction theory will hold up. Moreover, it is not clear how to compute the auction from such an approximate description. At least empirically, we demonstrate that even on complex (Section 2 presents example of complex environment), realistic distributions, it is possible to overcome these obstacles and obtain near optimal revenues.

Another major difference between our work and previous work, is the scalability of our system and the generality of the distributions that we can handle. We achieve these properties by working with a set of samples instead of the whole distribution, by applying the KLA method of [13, 14, 15] (with approximate conditional distributions), and by using either a novel method of extending incentive compatible auctions from subsets of the type space, or by restricting ourselves to simpler families of ascending auctions. Interestingly, on most of our benchmarks, the revenues of the ascending auctions and of the general incentive compatible auctions were very close. We note that in practical application, we can push the scalability of the system even further by querying the user about the type distribution only after the low agents have been rejected.

In order to examine the system we conduct several experiments. We construct benchmark distributions that represent real life situations, and test the revenue of the system on them. The results are very encouraging. On all our benchmarks we beat the English auction, with and without a predefined reserve price. On some of our benchmarks, the gap between the English auction and the system is significant. Moreover, we beat the English auction, even when the system is limited to use simple ascending auctions. Typically, the revenue of our system is around the average between $\mathbf{E}[v_1]$ and $\mathbf{E}[v_2]$. On all our benchmarks, the system's revenue is within 90% of the upper bound on the revenue of the optimal valid mechanism (at least under the assumption that the distribution is well approximated by the sampling process). The welfare of our system is also high, ranging from 90% to 100%.

Our experiments give initial insights into several issues that were not investigated in the context of single item auctions. These include a comparison between the revenue of Bayesian and dominant strategy auctions, the price of fairness, and the power of ascending auctions.

Remark: Due to space constraints, a lot of material was omitted from this exposition. A full version of this paper is available online at *http://iew3.technion.ac.il/~amirr*.

2. EXAMPLE: TECHNOLOGY SELL

In order to illustrate the complexity of realistic auction design problems consider the following example. A hi-tech company that owns a unique technology is for sale. Five companies are interested in buying it. Suppose that the main factors that determine the worth of the company to each potential buyer are the expected contribution of the new technology to its profit, the expected increase in its operational costs, and the expected change in its stock value. All these parameters are likely to be stochastic and include both common and private components. In addition, no company can pay more than its budget. It is unlikely, from both cognitive and computational perspectives, that the designer will be able to write down the joint distribution of the agents' values. Yet, it is more reasonable to expect that the user will be able to estimate the distribution of each of the main parameters that generate it (e.g. the budgets or the operational costs). Needless to say, we do not expect neither the resulting distribution to have a closed formula nor the optimal auction to be easy to calculate.

The setup described above is one of our benchmarks. A detailed description of the benchmark and the system's performance on it can be found in the full version of this paper.

3. MAIN CHALLENGES

This section describes the main challenges that we faced along with the methods that we used in order to overcome them.

3.1 Describing the distribution

By and large, the economic literature on auction design implicitly assumes that a full description of the underlying distribution ϕ on the agents' valuations is available to the designer. While this assumption facilitates obtaining a lot of insight on auction design, it is impractical for many applications. First, it is not clear how the designer can extract such knowledge about the agents. Second, even if it can, the size of the distribution is exponential and hence it is not even clear how to describe it to the computer. For concreteness, if there are 10 agents, each with 20 possible values, there are already more than 10^{13} possible combinations of values. Moreover, in many applications, there are non trivial dependencies between the valuations and there is no apparent way of calculating the joint distribution (e.g. computing the probability of a combination $(v_1, \ldots, v_n)$ in our technology auction example). Thus, we seek for substitutions of direct work with the underlying distribution ϕ.

Our approach While it is not reasonable to expect the user to be capable of describing the joint distribution directly, we believe that it is plausible to expect it to describe how this distribution is *generated*. This means, that it will be able to describe the main factors that influence the valuations, and to estimate the distribution of each one of them. Thus, our first step is to provide the user with a convenient method of supplying such descriptions.

Technically, the user describes a sampleable distribution via an XML file. The system allows the usage of sampleable stochastic variables, mathematical expressions over these variables, discrete ranges with rounding operations, etc. For example, the value of an agent in the technology auction example is simply represented as the following expression:

$$v_i = \min(budget_i, max(resalePrice_i, commonProfit \cdot \alpha_i - commonCost \cdot \beta_i + \rho_i \cdot marketPrice_i))$$

where the parameters, both common and private, are either constants or stochastic variables.

The system then generates an approximation of ϕ via *sampling*. In order to sample of a type vector, the system samples the basic variables and applies the above operators on them. Due to computational reasons, the system stores the samples in a database and works directly with the set of samples.

Remarks and open questions When either a full description of the distribution or a set of samples are available, the system can work directly with them. In the future, we plan to incorporate statistical methods in order to approximate the distribution from partial descriptions. It is also possible to introduce sensitivity analysis into the system. Such analysis can be done either over the distribution as a whole or over parameters provided by the user (e.g. budget).

The size of the distribution ϕ is exponential in the number of bidders. Thus, any feasible approximation method, causes some information loss. It is possible to generate distributions in which this information loss causes significant revenue loss (see [14]). Yet we believe that such examples are artificial. Indeed, on all our benchmarks, the system extracts near optimal revenues. We leave a theoretical study of the revenue loss, e.g. as a function of the entropy of ϕ, to future work.

Finally, one can view the problem of choosing the optimal auction from a family of possible mechanisms as a learning problem. The number of samples required in order to learn which auctions to choose may vary between families. This may give an advantage to simple classes of mechanisms over the class of all possible valid mechanisms. As far as we know, this topic has not yet been studied in the context of single item auctions.

3.2 Approximating conditional distributions of the high agents

In the first stage, a *kla*-auction finds the types and the identities of the low agents. The auction then computes the conditional distribution on the types of the k highest agents. When ϕ is given, this conditional distribution can be calculated in a straightforward manner. However, when it is only possible to sample ϕ, an exponential number of samples may be required in order to produce samples in which the low types match the actual types of the low agents. In order to overcome this, we exploit only *partial information* from the types of the low agents. Let $res = v_{k+1}$ be the $k+1$'th highest value. We define

$$\tilde{\phi}(v_1, \ldots, v_k) = \mathbf{Pr}[(v_1, \ldots, v_k) \,|\, \forall i, v_i \geq res],$$

i.e., we use only the information that the values of the high agents are greater than r. We leave other methods of approximating the conditional distribution to future research.

The set of samples that corresponds to $\tilde{\phi}$ is obtained from the set of samples of ϕ by querying the database and normalizing. In the second stage, we work directly with this set of samples.

3.3 Computing optimal auctions

Perhaps the greatest challenge in implementing the KLA method was to overcome the immense complexity of the optimal auction problem. While this problem can be translated into an integer program (see details in the full version of this paper), the size of this program is exponential in the number of agents. Even for a very small number of agents, solving the program is usually infeasible. In order to overcome this we developed a novel method of solving the problem on a subspace of the set of all possible type vectors, and then extending the solution to a valid auction on the whole type space. The method is described in Section 4. Currently, our method works only for dominant strategy IC. Even with our method, we can only cope with the fractional version of the problem. We also compute other mechanisms such as the Vickrey auction which is optimal among all Vickrey auctions that use up to k reserve prices; one per buyer. We call this mechanism *k-reserve*. This auction is simple, deterministic, and can be implemented via a standard ascending auction. As we shall see, the revenue of this mechanism is often close to the optimal revenue. Thus, in such cases, it may be better to use the k-reserve mechanism. The system computes the expected revenues of all the second stage mechanisms as well as bounds on the revenue of the optimal valid and Bayesian valid mechanisms. The user can use these estimations in order to decide which auction format to choose.

4. WORKING WITH SAMPLES

As we noted, perhaps the main challenge in implementing the system was to solve the optimal auction problem. A huge gain is obtained from the KLA method since we only need to solve the problem for a small number k of agents. Still, the size of the type-space $\mathbb{S}$ plays a crucial role. In the full paper we describe a novel method of constructing valid mechanisms from samples, and by this, drastically reduce the complexity of the problem. Currently, our method works only for valid auctions. The Bayesian case is left as an intriguing open problem.

5. DESCRIPTION OF THE SYSTEM

This section describes the architecture of our system. The system is illustrated in Figure 1. The main steps of the system are as follows:

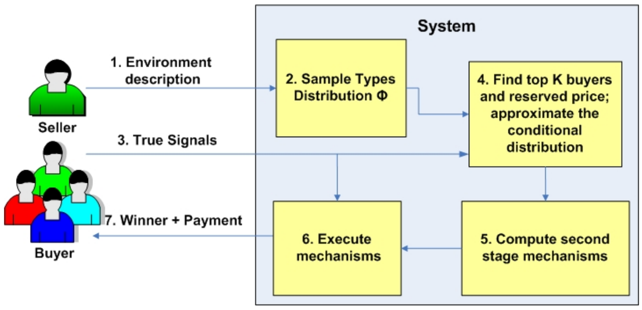

Figure 1: Main Steps

1. Getting the user's description of the environment.
2. Generating samples of the environment's distribution ϕ.
3. Getting the actual types signals from buyers.
4. Computing the inputs of the second stage auction: top k agents, reserve price v_{k+1}, and the conditional distribution $\tilde{\phi}$.
5. Computing the second-stage mechanisms.
6. Executing the second-stage mechanisms on the high agents and outputting the allocation and payment.

We now elaborate on each of the above steps.

5.1 Environment description

In our system the user describes the underlying environment via an XML file. This file contains information about the number of agents, constants, distributions of the various variables that determine the agents' values, value ranges, etc. We use type expressions in order to let the user express how the above variables determine the agents' private types. We support standard operations to build expressions '(', ')', '+', '-', '*' ,'/' ,$max(.)$, and $min(.)$ as well as complex expressions. The user can also load its own functions in order to create custom expressions via API interfaces. An example of such an XML file can be found in the full version of the paper.

In order to sample the type vector, the system samples each of the above variables, and then applies the above expressions for computing the agents' types. In order to ease on the usage of discrete distributions, we round the agents' types according to range classes provided by the user.

5.2 Approximation of the environment's distribution ϕ

After the system initializes itself according to the environment description, it approximates the underlying distribution by sampling the agents' type vector. The sampling process generates these type vectors by sampling the variables provided by the user and applying the type expressions on them. We store the set of samples in a database and work directly with it.

5.3 Getting the actual type signals

The buyers are requested to report their actual types to the system. Since our mechanisms are valid (or Bayesian valid), it is the agents' interest to report their actual types. In the experiments, we simulate the actual types of the agents by further samples of the environment description. We stress that we use additional samples and not the samples that were used to prepare ϕ.

5.4 Computing the input of the second stage

After constructing ϕ and getting the agents' values, the system computes the input of the second stage of the auction. It sorts the agents according to their values and finds out the ones with the k highest values. The $(k+1)$'th value, v_{k+1}, will be used as a reserve price in the second stage. The computation of the conditional distribution $\tilde{\phi}$, is described in Section 3.2.

5.5 Computing the second stage mechanisms

In the second stage of the auction, the system constructs several mechanisms: KLA which is approximately optimal among all valid auctions, k-reserve which is optimal among all the Vickrey auctions that use k reserve prices, k-fair which is optimal among all Vickrey auctions with a single reserve price, and Bayesian which is optimal among all Bayesian valid auctions. The details of these mechanisms can be found in the full version of this paper.

In the second stage, the system already knows that the types of all the high agents are above v_{k+1}. Every agent also has a maximal type v_i^{max} in the set of samples $\tilde{\phi}$. We thus solve the optimization problem that corresponds to each of the above mechanisms, only for type vectors in which the type of each agent i is between v_{k+1} and v_i^{max}.

Along with each of the above mechanisms, the system computes an estimation of its expected revenue. The system also computes some other intermediate mechanisms and upper bounds on the revenues of the optimal valid and Bayesian valid auctions. In reality, the system can use these estimations in order to determine which auction format to use in the second stage.

5.6 Executing the second stage mechanisms on the actual type vector

After computing the second stage mechanisms, the final stage of the system is to execute these mechanisms on the actual type vector of the high agents. The result of each of these mechanisms is an allocation and payment. In the experimental setup, we add the payment and the welfare to the statistics of each mechanism. In reality, the system would only choose one of the above auctions and execute it on the high agents.

6. EXPERIMENTS

In order to test our system we defined benchmark distributions that represent various realistic situations. In all of the experiments we approximate ϕ via sampling, then, in a loop, we sample a new actual type vector from the user's description, compute the inputs to the second stage, execute the various second stage auctions on the sampled signal, and take statistics. The reported results refer to an average of about 100 runs of each mechanism. Due to space constraints we only report the results of five benchmarks. We also constructed several other distributions that yielded similar phenomena.

A detailed description of the benchmark distributions and the results of the experiments can be found in the full version of this paper.

6.1 Outcomes and conclusions

The performance of our system.

As we noted, the vast majority of the auctions that are currently used in practice are English (with or without predefined reserve prices). An English auction without a reserve price yields an expected revenue of $\mathbf{E}[v_2]$ (up to discretization effects). On all our benchmarks, even the optimal predefined reserve price never improved the expected revenue of the English auction by more than 2%. Thus we used $\mathbf{E}[v_2]$ as a baseline for comparison.

Figure 2 shows the average full surpluses $\mathbf{E}[v_1]$ and the average revenues of our system. The advantage of the system over the English auction ranged from 41% in the coupling benchmark to 10% in the Art case. On all the benchmarks, the systems's revenue is around the average of $\mathbf{E}[v_1]$ and $\mathbf{E}[v_2]$. When there is a significant gap between the two highest bidders, the advantage of the system over English auctions is considerable. The welfare of the system ranged from 90% to 100%. Our system outputs an upper bound on the revenue of the optimal valid auction (at least under the assumption that the sampling process yields a sufficient approximation of $\tilde{\phi}$). On all our benchmarks the system yielded a revenue within 90% to 100% of this upper bound. In cases where the extension method was invoked, there was an additional loss of 0% to 15% in the KLA mechanism, even when the size of the type space was much larger then the feasible size.

Ascending auctions.

The mechanism computed by the KLA method can be non intuitive and probabilistic. Often, it is desired to use simpler and more intuitive auctions. k-reserve is an example of such an auction. It is equivalent to an English auction except that different reserve prices can be put on different agents. Such a mechanism can be implemented either as a revelation mechanism or as an ascending auction.

On all the benchmarks except coupling, the revenues of both auctions where close to each other. Thus, in many cases, it may be better to use ascending auctions. In the coupling example, we got a much higher revenue by setting k to 2. This is non surprising since the coupling is between pairs. In the future we plan to add the possibility to wait until other agents will drop out or at least to optimize over all k'-reserve auctions where $k' \leq k$.

The price of fairness.

Both KLA and k-reserve auctions treat different agents differently. In various settings, such unfair treatment may be unlawful or a source of disputes. There are several natural definitions of fairness in auctions (see, e.g., [3]). Perhaps the strongest is to choose the reserve price in advance and then conduct a Vickrey auction. In this case, both, the allocation and the payment are symmetric functions. A weaker but still reasonable notion of fairness is envy freeness. In this case, either no agent wins or the highest agent wins and its payment is at least v_2. In such a case, no agent can benefit from "switching" the outcome with another agent. The expected revenue of any envy free auction, as well as of any English auction with a predefined reserve price, is bounded by the expected revenue of a k-fair auction with $k = 1$. At least on all our benchmarks, the revenue of this auction never exceeded $\mathbf{E}[v_2] + 2.5\%$. We thus conclude that the price of fairness in single item auctions is often significant.

Dominant VS Bayesian auctions.

The set of the Bayesian valid auctions is a strict superset of the set of valid auctions. Indeed we saw an advantage typically between 3% to 5% to the Bayesian model. Since the concept of dominant strategies is much stronger than Bayesian equilibrium, we are not sure that such margins justify abandon the advantages of a dominant strategy implementation.

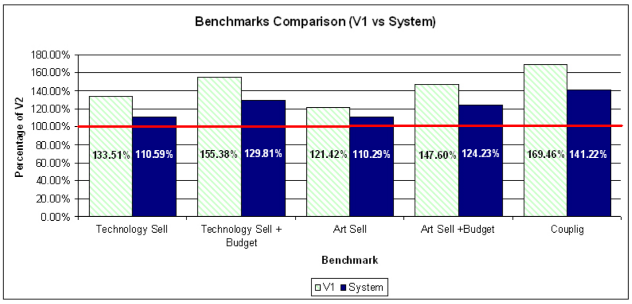

Figure 2: Revenue comparison

7. FUTURE RESEARCH

In this paper we demonstrated the huge potential but also the complexity of automated auction design. Many open issues stem from this work. This section illustrates some of them.

This work focuses on single item auctions in the private value model. In the future we would like to extend the system to support multi-attribute procurement auctions[11], multi-unit auctions, and perhaps even combinatorial auctions. Other extensions include interdependent valuations, and the relaxation of the IR and IC properties. All these extensions require solving non trivial theoretical problems as well as overcoming implementation obstacles. Extensions which are easier to implement include the introduction of risk attitudes for the seller and the buyers, incorporating sensitivity analysis, optimizing according to multiple criteria (e.g. efficiency and revenue), etc. Even in its current form, the system can support the management of ascending auctions. We would like to extend it to support other non-revelation mechanisms such as first price auctions.

The optimal auction problem is extremely difficult and poses many non trivial algorithmic challenges. Other computational questions are related to our extension method and to methods of approximating the underlying distribution from partial descriptions. Open questions which are related to the connection between our setup and computational learning theory were presented within the body of the paper.

Acknowledgements. We thank Inbal Ronen for her comments on an earlier draft of this paper and Alex Rogers for an helpful discussion.

8. REFERENCES

[1] Vincent Conitzer. *Computational Aspects of Preference Aggregation*. PhD thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, 2006.

[2] J. Crémer and R. McLean. Full extraction of the surplus in bayesian and dominant strategy auctions. *Econometrica*, 56(6):1247–57, 1988.

[3] A. Goldberg and J. Hartline. Envy-free auction for digital goods. In *Proceedings of the 4th ACM Conference on Electronic Commerce. (EC'03)*, 2003.

[4] A. Goldberg, J. Hartline, and A. Wright. Competitive auctions and digital goods. In *12th Annual ACM-SIAM Symposium on Discrete Algorithms*, January 2001.

[5] B. Kleinberg. A multiple-choice secretary problem with applications to online auctions. In *Proceedings of the 16th ACM-SIAM Symposium on Discrete Algorithms (SODA 2005)*, 2005.

[6] P. Klemperer. *The Economic Theory of Auctions*. Edward Elgar Publishing, 2000.

[7] V. Krishna. *Auction Theory*. Elsevier Press, 2006.

[8] P. McAfee and P. Reny. Correlated information and mechanism design. *Econometrica*, 60:395–421, 1992.

[9] R. B. Myerson. Optimal auction design. *Mathematics of operational research*, 6:58–73, 1981.

[10] Z. Neeman. The effectiveness of english auctions. *Games and Economic Behavior*, 43:214–238, 2003.

[11] D. Parkes and J. Kalagnanam. Iterative Multiattribute Vickrey Auctions. *Management Science*, 2004. Special Issue on Electronic Markets.

[12] D. Lucking Riley. Auctions on the internet: What's being auctioned, and how? *Journal of Industrial Economics*, 48(3):227–252, 2000.

[13] A. Ronen. On approximating optimal auctions. In *The Third ACM Conference on Electronic Commerce (EC01)*, pages 11–17. ACM, 2001.

[14] A. Ronen and A. Saberi. On the hardness of optimal auctions (extended abstract). In *The 43rd Annual IEEE Symposium on the Foundations of Computer Science (FOCS 2002)*, pages 396 – 405. IEEE, 2002.

[15] A. Ronen and A. Saberi. Approximating optimal auctions. Manuscript, 2006.

[16] T. Sandholm. Automated mechanism design: A new application area for search algorithms. In *Proceedings of the International Conference on Principles and Practice of Constraint Programming (CP)*, 2003.