

# Automated Mechanism Design in Infinite Games of Incomplete Information: Framework and Applications

Yevgeniy Vorobeychik  
University of Michigan  
Computer Science & Engineering  
Ann Arbor, MI 48109-2121 USA  
yvorobey@umich.edu

Daniel M. Reeves  
Yahoo! Research  
45 W 18th St, 6th Floor  
New York, NY 10011-4609 USA  
dreeves@yahoo-inc.com

Michael P. Wellman  
University of Michigan  
Computer Science & Engineering  
Ann Arbor, MI 48109-2121 USA  
wellman@umich.edu

## Abstract

We present a functional framework for automated Bayesian and robust mechanism design based on a two-stage game model of strategic interaction between the designer and the mechanism participants, and apply it to several classes of two-player infinite games of incomplete information. Our approach yields optimal or nearly optimal mechanisms in three application domains using various objective functions. By comparing our results with known optimal mechanisms, and in some cases improving on the best known mechanisms, we show that ours is a promising approach to parametric design of indirect mechanisms.

## 1 Motivation

While the field of Mechanism Design has been quite successful within a wide range of academic disciplines, much of its progress came as a series of arduous theoretical efforts. In its practical applications, however, successes have often

been preceded by a series of setbacks, with the drama of auctioning radio spectrum licenses that unfolded in several countries providing a powerful example [Klemperer, 2004, McMillan, 1994]. The following quote by Klemperer [2004] is, perhaps, especially telling: “Most of the extensive auction literature... is of second-order importance for practical auction design.”

A difficulty in practical mechanism design that has been especially emphasized is the unique nature of most practical design problems. Often, this uniqueness is manifest in the idiosyncratic nature of objectives and constraints. For example, when the US government tried to set up a mechanism to sell the radio spectrum licenses, it identified among its objectives promotion of rapid deployment of new technologies. Additionally, it imposed a number of non-traditional constraints, such as ensuring that some licenses go to minority-owned and women-owned companies [McMillan, 1994].

Thus, a prime motivation for Conitzer and colleagues’ automated mechanism design work (for example, Conitzer and Sandholm [2003a, 2004], Sandholm et al. [2007]) was to produce a framework for solving mechanism design problems computationally given arbitrary objectives and constraints. While we are similarly motivated, we recognize and tackle an additional problem: reliance on direct truthful mechanisms. This reliance has at its core the Revelation Principle [Myerson, 1981], which states that any outcome that can be achieved by an arbitrary mechanism can also be achieved if we restrict the design space to mechanisms that induce truthful revelation of agent preferences. While theoretically sound and general in settings without non-traditional constraints, there have been criticisms of the principle on computational grounds, for example, those leveled by Conitzer and Sandholm [2003b]. It is also well recognized that if the design space is restricted in arbitrary ways, the revelation principle need not hold.<sup>1</sup> While the former set of criticisms can be addressed to some degree by multi-stage mechanisms that implement partial revelation of agent preferences in a series of steps (for example, an ascending auction), the latter criticisms are a property of the idiosyncratic constraints on the design problem at hand and offer a more difficult hurdle to overcome.

In this work, we introduce an approach to the design of indirect mechanisms<sup>2</sup> and apply it to several mechanism design problems. In many ways, our methods follow in the footsteps of the work on empirical mechanism design [Vorobeychik et al., 2006], although here we present a more systematic approach, albeit restricted to a particular class of infinite games of incomplete information. In practice, of course, we cannot possibly tackle an arbitrarily complex design space. Our simplification comes from assuming that the designer seeks to find the best setting for particular design parameters. In other words, we allow the

---

<sup>1</sup>As a simple example, imagine that the designer’s only choice is a first-price sealed-bid auction. Since this auction is not truthful, the revelation principle clearly fails in this restricted design space. Abstractly and generally, we can simply imagine eliminating truthful mechanisms from the design space.

<sup>2</sup>Indeed, several of our applications yield Bayes-Nash equilibrium strategies that do not correspond to any player type, although all equilibria we obtain are linear functions of the type parameter.

designer to search for a mechanism in some subset of an  $n$ -dimensional Euclidean space, rather than in an arbitrary function space, as would be required in a completely general setting.

In the following sections, we present our framework for automated mechanism design and test it out in several application domains. We specifically look at two settings: Bayesian and robust. In both settings, we assume that the designer knows the probability distribution over agent types. The difference, rather, is in the designer’s willingness to bear risk. In the Bayesian setting, the designer simply maximizes expected value of the objective function. In the robust setting, however, the designer is willing to take few chances, and is instead interested in maximizing relative to the worst outcome. Since it is impossible to guarantee computationally that a particular mechanism is robust with respect to every realization of agent types, we introduce the notion of *probably approximately robust mechanism design*, which instead aims to probabilistically ensure that very few type profiles can result in poor outcomes for the designer. Our results suggest that our approach has much promise: most of the designs that we discover automatically are nearly as good as or better than the best known hand-built designs in the literature.

## 2 Preliminaries

In this work we restrict our attention to *one-shot games of incomplete information*, denoted by  $[I, \{A_i\}, \{T_i\}, F(\cdot), \{u_i(a, t)\}]$ , where  $I$  refers to the set of players and  $m = |I|$  is the number of players.  $A_i$  is the set of actions available to player  $i \in I$ , and  $A_1, \dots, A_m$  is the joint action space.  $T_i$  is the set of types (private information) of player  $i$ , with  $T = T_1 \times \dots \times T_m$  representing the joint type space. Since we presume that a player knows his type prior to taking an action, but does not know types of others, we allow him to condition his action on own type. Thus, we define a strategy of a player  $i$  to be a function  $s_i : T_i \rightarrow \mathbb{R}$ , and will use  $s(t)$  to denote the vector  $(s_1(t_1), \dots, s_m(t_m))$ .  $F(\cdot)$  is the distribution over the joint type space.

It is often convenient to refer to a strategy of player  $i$  separately from that of the remaining players. To accommodate this, we use  $s_{-i}$  to denote the joint strategy of all players other than player  $i$ . Similarly,  $t_{-i}$  designates the joint type of all players other than  $i$ .

We define the payoff (utility) function of each player  $i$  by  $u_i : A \times T \rightarrow \mathbb{R}$ , where  $u_i(a_i, t_i, a_{-i}, t_{-i})$  indicates the payoff to player  $i$  with type  $t_i$  for playing action  $a_i \in A_i$  when the remaining players with joint types  $t_{-i}$  play  $a_{-i}$ . Given a strategy profile  $s \in S$ , the expected payoff of player  $i$  is  $\tilde{u}_i(s) = E_t[u_i(s(t), t)]$ .

Faced with such a game, we assume that the players would play optimally against each other. A profile of strategies that are mutually optimal is called a *Bayes-Nash equilibrium*.

**Definition 1.** A strategy profile  $s = (s_1, \dots, s_m)$  constitutes a Bayes Nash equilibrium of game  $[I, \{R_i\}, \{T_i\}, F(\cdot), \{u_i(r, t)\}]$  if for every  $i \in I$  and  $s'_i \in S_i$ ,  $\tilde{u}_i(s_i, s_{-i}) \geq \tilde{u}_i(s'_i, s_{-i})$ .

## 3 Automated Mechanism Design Framework

### 3.1 General Framework

We can model the strategic interactions between the designer of the mechanism and its participants as a two-stage game [Vorobeychik et al., 2006]. The designer moves first by selecting a value  $\theta$  from a set of allowable mechanism settings,  $\Theta$ . All the participant agents observe the mechanism parameter  $\theta$  and move simultaneously thereafter. For example, the designer could be deciding between a first-price and a second-price sealed-bid auction mechanism, with the presumption that after the choice has been made, the bidders will participate with full awareness of the auction rules.

Since the participants know the mechanism parameter, we define a game between them in the second stage as

$$\Gamma_\theta = [I, \{A_i\}, \{T_i\}, F(\cdot), \{u_i(a, t, \theta)\}].$$

We refer to  $\Gamma_\theta$  as the game *induced* by  $\theta$ . As is common in mechanism design literature, we will evaluate mechanisms with respect to a sample Bayes-Nash equilibrium,  $s(t, \theta)$ . We say that given an outcome of play  $r$ , the designer's goal is to maximize a welfare function  $W(r, t, \theta)$  with respect to the distribution of types. Thus, given that a Bayes-Nash equilibrium,  $s(t, \theta)$ , is the relevant outcome of play, the designer's problem is to maximize  $W(s(\theta), \theta) = E_t[W(s(t, \theta), t, \theta)]$ .<sup>3</sup>

Observe that if we *knew*  $s(t, \theta)$  as a function of  $\theta$ , the designer would simply be faced with an optimization problem. This insight is actually a consequence of the application of backwards induction, which would have us find  $s(t, \theta)$  first for every  $\theta$  and then compute an optimal mechanism with respect to these equilibria. If the design space were small, backwards induction applied to our model would thus yield an algorithm for optimal mechanism design. Indeed, if additionally the games  $\Gamma_\theta$  featured small sets of players, strategies, and types, we would say little more about the subject. Our goal, however, is to develop a mechanism design tool for settings in which these assumptions do not hold. Specifically, we presume that it is infeasible to obtain a solution of  $\Gamma_\theta$  for every  $\theta \in \Theta$ , either because the space of possible mechanisms is large, or because solving (or approximating solutions to)  $\Gamma_\theta$  is very computationally intensive. Additionally, we try to avoid making assumptions about the objective function or constraints on the design problem or the agent type distributions. We do restrict the games to two players with piecewise linear utility functions, but allow them to have infinite strategy and type sets.

In short, we propose the following high-level procedure for finding optimal mechanisms:

1. Select a candidate mechanism,  $\theta$ .
2. Find (approximate) solutions to  $\Gamma_\theta$ .

---

<sup>3</sup>Note the overloading of  $W(\cdot)$ .

3. Evaluate the objective and constraints given solutions to  $\Gamma_\theta$ .
4. Repeat this procedure for a specified number of steps.
5. Return an approximately optimal design based on the resulting optimization path.

**Bayesian Mechanism Design** In much of this paper we explore the problem of Bayesian Mechanism Design, in which the designer is presumed to have a belief about the distribution of agents' types. The designer in this framework is also required to know what distribution of agents' types is common knowledge to all the agents. This problem can be formulated in our framework (as we had already mentioned) by expressing the designer's optimization function as the expectation of his objective with respect to his belief about the agent joint type distribution, that is

$$W(s(\theta), \theta) = E_t[W(s(t, \theta), t, \theta)].$$

In this work, we assume that the designer has the same belief about agent types as the agents themselves, although this assumption is there purely for convenience.

**Probably Approximately Robust Mechanism Design** We address the problem of robust mechanism design by allusion to the analogous problem in the optimization literature, referred to as Robust Optimization [Ben-Tal and Nemirovski, 2002]. In Robust Optimization, uncertainty over parameters of an optimization program is accounted for by treating the uncertainty set essentially as an adversary which selects the worst outcome for the problem at hand. Thus, the solution to the robust program is one that gives the best outcome in the face of such an adversary.

Our analogy comes from treating the type space of agents as such an adversary. As a justification for such a pessimistic outlook, we can imagine that the designer is extremely averse to poor outcomes, perhaps envisioning a politician who is extremely worried about being reelected. While risk aversion can be treated formally in a Bayesian framework, such treatment requires the designer to be aware of his risk preferences. Robust treatment sidesteps this issue and may provide a useful approximation instead.

Formally, we can express the robust objective of the designer as

$$W(s(\theta), \theta) = \inf_{t \in T} W(s(t, \theta), t, \theta). \tag{1}$$

Note that this change is relatively minor and has no effect on the rest of the framework (replacing the expectation operator with the infimum). However, it entails a computationally infeasible problem of ensuring robustness for every joint type of a possibly infinite type space. To address this problem, we relax the pure robustness criterion to probabilistic robustness. Our relaxation is that the designer is not worried about the worst subset of outcomes of the type space

if that subset has very small measure. For example, if 0.0001% of types are extremely unfavorable, their appearance is deemed sufficiently unlikely not to worry the designer. Furthermore, we probabilistically ascertain that the worst outcome based on a finite number of samples from the type distribution is no better than a high proportion of the type space. We call this paradigm *probably approximately robust* mechanism design.

To formalize this, suppose that in every exploration step using our framework we take  $n$  samples from the type distribution,  $T^n = \{T_1, \dots, T_n\}$ , and then select the worst value of the objective over these  $n$  types:

$$\hat{W}(s(t, \theta), t, \theta) = \min_{t \in T^n} W(s(t, \theta), t, \theta).$$

We would like to select a sufficiently high number of samples  $n$ , in order to attain high enough confidence,  $1 - \alpha$ , that the best objective value that we obtain via  $L$  explorations using our framework is approximately robust. The following theorem gives us such an  $n$ .

**Theorem 1.** *Suppose we select the best design of  $L$  candidates, using  $n$  samples from the type distribution for each to estimate the value of  $\inf_{t \in T \setminus T_A} W(s(t, \theta), t, \theta)$ , where  $T_A$  is the set of types with value of  $W(s(t, \theta), t, \theta)$  below  $\hat{W}(s(t, \theta), t, \theta)$ . If we want to attain the confidence of at least  $1 - \alpha$  that the measure of  $T_A$  is at most  $p$ , we need*

$$n \geq \frac{\log(1 - (1 - \alpha)^{\frac{1}{L}})}{\log(1 - p)}$$

*samples.*

The proofs of this and other results can be found in the appendix.

### 3.2 Designer’s Optimization Problem

We begin by treating the designer’s problem as black-box optimization, where the black box produces a noisy evaluation of an input design parameter,  $\theta$ , with respect to the designer’s objective,  $W(s(\theta), \theta)$ , given the game-theoretic predictions of play. Once we frame the problem as a black-box (simulation) optimization problem, we can draw on a wealth of literature devoted to developing methods to approximate optimal solutions [Olafsson and Kim, 2002]. While we can in principle select a number of these, we have chosen simulated annealing, as it has proved quite effective for a great variety of simulation optimization problems in noisy settings, particularly in problems with many local optima [Corana et al., 1987, Fleischer, 1995, Siarry et al., 1997]. We opt for a relatively simple adaptive implementation of simulated annealing, with normally distributed random perturbations applied to the solution candidate in every iteration (respecting the constraint set  $\Theta$ ).

As an application of black-box optimization, the mechanism design problem in our formulation is just one of many problems that can be addressed with one of a selection of methods. What makes it special is the subproblem of evaluating

the objective function for a given mechanism choice, and the particular nature of typical mechanism design constraints.

### 3.3 Objective Evaluation

As implied by the backwards induction process, we must obtain the solutions (a Bayes-Nash equilibrium in our case) to a game induced by the design choice,  $\theta$ , in order to evaluate the objective function. In general, this is simply not possible to do, since Bayes-Nash equilibria may not even exist in an arbitrary game, nor is there a general-purpose tool to find them. However, there are a number of tools that can find or approximate solutions in specific settings. For example, GAMBIT [McKelvey et al., 2005] is a general-purpose toolbox of solvers that can find Nash equilibria in finite games, although it is often ineffective for even moderately sized games.

To the best of our knowledge, the only solver for infinite games of incomplete information was introduced by [Reeves and Wellman, 2004] (henceforth, RW). In fact, RW is a best response finder, which has successfully been used iteratively to obtain sample Bayes-Nash equilibria for a restricted class of infinite two-player games of incomplete information.

While RW is often effective in converging to a sample Bayes-Nash equilibrium, it does not do so always. Thus, we are presented with the first problem that makes automated mechanism design unique: how do we evaluate the objective if no solution can be obtained? There are a number of ways to approach this difficulty. For example, we can use a uniform distribution over pure strategy profiles in lieu of a Bayes-Nash equilibrium. However, this may not be possible, as the set of pure strategy profiles may be unbounded. Alternatively, since we are dealing with an iterative tool that will always (at least in principle) produce a best response to a given strategy profile, we can use the last best response in the non-converging finite series of iterations as the prediction of agent play. Finally, we may simply constrain the design to discard any choices for which our solver does not produce an answer. Here we employ the last alternative, which is the most conservative of the three.

Since the goal of automated mechanism design is to approximate solutions to design problems with arbitrary objectives and constraints and to handle games with arbitrary type distributions, we treat the probability distribution over player types as a black box from which we can sample joint player types. Thus, we use numerical integration (sample mean in our implementation) to evaluate the expectation of the objective with respect to player types, thereby introducing noise into objective evaluation.

### 3.4 Dealing with Constraints

Mechanism design can feature any of the following three classes of constraints: ex ante (constraints evaluated with respect to the joint distribution of types), ex interim (evaluated separately for each player and type with respect to the joint type distribution of other players), and ex post (evaluated for every joint

type profile). When the type space is infinite we of course cannot numerically evaluate any expression for every type. We therefore replace these constraints with probabilistic constraints that must hold for all but a set of types (or joint type profiles) of small measure. For example, an ex post individual rationality (IR) constraint would only have to hold for type profiles that can occur with probability greater than 95%.

Besides a computational justification, there is a practical justification for weakening the requirement that constraints be satisfied for every possible player type (or joint type profile), at least as far as individual rationality is concerned. When we introduce constraints that hold with high probability, we may be excluding a small measure set of types from participation (this is the case for the individual rationality constraints). But by excluding a small portion of types, the expected objective function will change very little, and, similarly, such a change will introduce little incentive for other types to deviate. Indeed, by excluding a subset of types with low valuations for an object, the designer may raise its expected revenue [Krishna, 2002].

Even when we weaken traditional constraints to their probabilistic equivalents, we still need a way to verify that such constraints hold by sampling from the type distribution. Since we can only take a finite number of samples, we will in fact only verify a probabilistic constraint with some level of confidence. The question we want to ask, then, is how many samples do we need in order to say with confidence  $1 - \alpha$  that a particular constraint holds for a type set with probability measure at least  $p$ ? That is the subject of the following theorem.

**Theorem 2.** *Let  $B$  denote a set on which a probabilistic constraint is violated, and suppose that we have a uniform prior over the interval  $[0, 1]$  on the probability measure of  $B$ . Then, we need at least  $\frac{\log \alpha}{\log p} - 1$  samples to verify with confidence  $1 - \alpha$  that the constraint holds on a set of types with measure at least  $p$ .*

In practice, however, this is not the end of the story for the ex interim constraints. The reason is that the ex interim constraint will take expectation with respect to the joint distribution of types of players other than the player  $i$  for which it is verified. Since we must evaluate this expectation numerically, we cannot escape the presence of noise in constraint evaluation. Furthermore, if we are trying to verify the constraint for many type realizations, it is quite likely that in at least one of these instances we will get unlucky and the numerical expectation will violate the constraint, even though the actual expectation does not. We circumvent this problem in two ways. First, we introduce a slight tolerance for a constraint, so that it will not fail due to small evaluation noise. Second, we split the set of types for which the constraint is verified into smaller groups, and throw away a small proportion of types in each group with the worst constraint evaluation result. For example, if we are trying to ascertain that ex interim individual rationality holds, we would throw away several types with the lowest estimated ex interim utility value.

One final general note on constraints: Since constraints are evaluated in our framework as a part of the objective function evaluation process, if a constraint

fails a value must still be returned for an objective function. Thus, we set the objective to negative infinity if any constraint fails.

We now describe three specific constraints that we consider in our applications.

**Equilibrium Convergence Constraint** The purpose of this constraint is to ensure that we have a reliable evaluation of any mechanism that we determine to be optimal. For this to be the case, we must ascertain that it is indeed evaluated with a true equilibrium (or near-equilibrium) strategy profile (given our assumption that a Bayes-Nash equilibrium is a relevant predictor of agent play). For example, best response dynamics using RW need not converge at all. We formally define this constraint as follows:

**Definition 2.** *Let  $s(t)$  be the last strategy profile produced in a sequence of solver iterations, and let  $s'(t)$  immediately precede  $s(t)$  in this sequence. Then the equilibrium convergence constraint is satisfied if for every joint type profile of players,*

$$|s(t) - s'(t)| < \delta$$

for some a priori fixed tolerance level  $\delta$ .

The problem that we cannot in practice evaluate this constraint for every joint type profile is resolved by making this constraint probabilistic, as described above. Thus, we define a *p-strong equilibrium convergence constraint*:

**Definition 3.** *Let  $s(t)$  be the last strategy profile produced in a sequence of solver iterations, and let  $s'(t)$  immediately precede  $s(t)$  in this sequence. Then the p-strong equilibrium convergence constraint is satisfied if for a set of type profiles  $t$  with probability measure no less than  $p$ ,*

$$|s(t) - s'(t)| < \delta$$

for some a priori fixed tolerance level  $\delta$ .

**Ex Interim Individual Rationality** This constraint (henceforth, Ex-Interim-IR) specifies that for every agent and for every possible agent's type, that agent's expected utility conditional on its type is greater than its opportunity cost of participating in the mechanism. Formally, it is defined as follows:

**Definition 4.** *The Ex-Interim-IR constraint is satisfied when for every agent  $i \in I$ , and for every type  $t_i \in T_i$ ,*

$$E_{t_{-i}} u_i(t, s(t) | t_i) \geq C_i(t_i),$$

where  $C_i(t_i)$  is the opportunity cost to agent  $i$  with type  $t_i$  of participating in the mechanism.

Again, in the automated mechanism design framework, we must change this to a probabilistic constraint as described above.

**Definition 5.** *The  $p$ -strong Ex-Interim-IR constraint is satisfied when for every agent  $i \in I$ , and for a set of types  $t_i \in T_i$  with probability measure no less than  $p$ ,*

$$E_{t_{-i}} u_i(t, s(t) | t_i) \geq C_i(t_i) - \delta,$$

where  $C_i(t_i)$  is the opportunity cost of agent  $i$  with type  $t_i$  of participating in the mechanism, and  $\delta$  is some a priori fixed tolerance level.

Commonly in the mechanism design literature the opportunity cost of participation,  $C_i(t_i)$ , is assumed to be zero but this assumption may not hold, for example, in an auction where not participating would be a give-away to competitors and entail negative utility (see Section 4.3).

**Minimum Revenue Constraint** The final constraint that we consider ensures that the designer will obtain some minimal amount of revenue (or bound its loss) in attaining a non-revenue-related objective.

**Definition 6.** *The minimum revenue constraint is satisfied if*

$$E_t k(s(t), t) \geq C,$$

where  $k(s(t), t)$  is the total payment made to the designer by agents with joint strategy  $s(t)$  and joint type profile  $t$ , and  $C$  is the lower bound on revenue.

## 4 Applications

In this section we present results from several applications of our automated mechanism design framework to specific two-player problems. One of these problems, finding auctions that yield maximum revenue to the designer, has been studied in a seminal paper by Myerson [1981] in a much more general setting than ours. Another one, which seeks to find auctions that maximize social welfare, has also been studied more generally. Additionally, in several instances we were able to derive optima analytically. For all of these we have a known benchmark to strive for. Others have no known optimal design.

An important consideration in any optimization routine is the choice of a starting point, as it will generally have important implications for the quality of results. This could be especially relevant in practical applications of automated mechanism design, for example, if it is used as a tool to enhance an already working mechanism through parametrized search. Thus, we would already have a reasonable starting point and optimization could be far more effective as a result. We explore this possibility in several of our applications, using a previously studied design as a starting point. Additionally, we apply our framework to every application with completely randomly seeded optimization runs, taking the best result of five randomly seeded runs in order to alleviate the problem posed by local optima. Furthermore, we enhance the optimization procedure by using a *guided* restart, that is, by running the optimization procedure once using the current best mechanism as a new starting point.

In all of our applications player types are independently distributed with uniform distribution on the unit interval. Finally, we used 50 samples from the type distribution to verify Ex-Interim-IR. This gives us 95% confidence that 94% of types lose no more than the opportunity cost plus our specified tolerance which we add to ensure that the presence of noise does not overconstrain the problem. It turns out that every application that we consider produces a mechanism that is individually rational for all types *with respect to the tolerance level that was set*.

#### 4.1 Shared-Good Auction (SGA)

Consider the problem of two people trying to decide between two options. Unless both players prefer the same option, no standard voting mechanism (with either straight votes or a ranking of the alternatives) can help with this problem. VCG is a nonstarter with no one to provide a subsidy and third-party payments are tantamount to a pure efficiency loss.

Instead we propose a simple auction: each player submits a bid and the player with the higher bid wins, paying some function of the bids to the loser in compensation. Reeves [2005] considered a special case of this auction and gave the example of two roommates using it to decide who should get the bigger bedroom and for how much more rent. We sometimes refer to this mechanism as an *un-sharing auction*—it allows one agent to sell its half of a good to the other joint owner (or pay the other to take on its half of a “bad”).

We define a space of mechanisms for this problem that are all budget balanced, individually rational, and (assuming monotone strategies) socially efficient. We then search the mechanism space for games that satisfy additional properties. The following is a payoff function defining a space of games parametrized by the function  $f$ .

$$u(t, a, t', a') = \begin{cases} t - f(a, a') & \text{if } a > a' \\ \frac{t - f(a, a') + f(a', a)}{2} & \text{if } a = a' \\ f(a', a) & \text{if } a < a', \end{cases} \quad (2)$$

where  $u()$  gives the utility for an agent who has a value  $t$  for winning and chooses to bid  $a$  against an agent who has value  $t'$  and bids  $a'$ . The  $t$ 's are the agents' types and the  $a$ 's their actions. Finally,  $f()$  is some function of the two bids.<sup>4</sup> In the tie-breaking case (which occurs with probability zero for many classes of strategies) the payoff is the average of the two other cases, i.e., the winner is chosen by the flip of a fair coin.

We now consider a restriction of the class of mechanisms defined above.

**Definition 7.**  $\text{SGA}(h, k)$  is the mechanism defined by Equation 2 with  $f(a, a') = ha + ka'$ .

<sup>4</sup> Reeves [2005] considered the case  $f(a, a') = a/2$ .

For example, in  $\text{SGA}(1/2, 0)$  the winner pays half its own bid to the loser; in  $\text{SGA}(0, 1)$  the winner pays the loser's bid to the loser. We now give Bayes-Nash equilibria for such games when types are uniform.

**Theorem 3.** *For  $h, k \geq 0$  and types  $U[A, B]$  with  $B \geq A + 1$  the following is a symmetric Bayes-Nash equilibrium of  $\text{SGA}(h, k)$ :*

$$s(t) = \frac{t}{3(h+k)} + \frac{hA+kB}{6(h+k)^2}$$

For the following discussion, we need to define the notion of truthfulness, or Bayes-Nash incentive compatibility.

**Definition 8 (BNIC).** *A mechanism is Bayes-Nash incentive compatible (truthful) if bidding  $s(t) = t$  constitutes a Bayes-Nash equilibrium of the game induced by the mechanism.*

The Revelation Principle [Mas-Colell et al., 1995] guarantees that for any mechanism, involving arbitrarily complicated sequences of messages between participants, there exists a BNIC mechanism that is equivalent in terms of how it maps preferences to outcomes. This is demonstrated by construction: consider a meta-mechanism that consists of the original mechanism with proxies inserted that take reported preferences from the agents and play a Nash equilibrium on the agents' behalf in the original game.

We can now characterize the truthful mechanisms in this space. According to Theorem 3,  $\text{SGA}(1/3, 0)$  is truthful for  $U[0, B]$  types. We now show that this is the *only* truthful design in this design space.

**Theorem 4.** *With  $U[0, B]$  types ( $B > 0$ ),  $\text{SGA}(h, k)$  is BNIC if and only if  $h = 1/3$  and  $k = 0$ .*

Of course, by the revelation principle, it is straightforward to construct a mechanism that is BNIC for any  $U[A, B]$  types. However, to be a proper auction [Krishna, 2002] the mechanism should not depend on the types of the participants. In other words, the mechanism should not be parametrized by  $A$  and  $B$ . With this restriction, the revelation principle fails to yield a BNIC mechanism for arbitrary uniform types. Below, we will show concrete examples of the failure of the revelation principle for several sensible designer objectives.

From now on we will restrict ourselves to the case of  $U[0, 1]$  types. Since  $\text{SGA}(1/3, 0)$  is the only truthful mechanism in our design space, we can directly compare the objective value obtained from this mechanism and the best indirect mechanism in the sections that follow.

#### 4.1.1 Bayesian Mechanism Design Problems

**Minimize Difference in Expected Utility** First, we consider as our objective *fairness*, or negative differences between the expected utility of winner and

loser. Alternatively, our goal is to minimize

$$\begin{aligned}
 & |E_{t,t'}[u(\text{winner}) - u(\text{loser})]| = \\
 & |E_{t,t'}[u(t, s(t), t', s(t'), k, h \mid a > a') - \\
 & \quad - u(t, s(t), t', s(t'), k, h \mid a < a')]|.
 \end{aligned} \tag{3}$$

We first use the equilibrium bid derived above to analytically characterize optimal mechanisms.

**Theorem 5.** *The objective value in (3) for SGA( $h, k$ ) is*

$$\frac{2h + k}{9(h + k)}.$$

*Furthermore, SGA(0,  $k$ ), for any  $k > 0$ , minimizes the objective, and the optimum is 1/9.*

By comparison, the objective value for the truthful mechanism, SGA(1/3, 0), is 2/9, twice as high as the minimum produced by an untruthful mechanism. Thus, the revelation principle does not hold for this objective function in our design space. We can use Theorem 5 to find that the objective value for SGA(1/2, 0), the mechanism described previously in literature [Reeves, 2005] is 2/9.

Now, to test our framework, we imagine we don't know about the above analytic derivations (including the derivation of the Bayes-Nash equilibrium) and run the automated mechanism design procedure in black-box mode. Table 1

Parameters	Initial Design	Approx. Optimal Design
$h$	0.5	0
$k$	0	1
objective	2/9	1/9
$h$	random	0
$k$	random	1
objective	N/A	1/9

Table 1: Design that approximately maximizes fairness (minimizes difference in expected utility between utility of winner and loser) when the optimization search starts at a fixed starting point, and the best mechanism from five random restarts.

presents results when we start the search at random values of  $h$  and  $k$  (taking the best outcome from 5 random restarts), and at the starting values of  $h = 0.5$  and  $k = 0$ . Since the objective function turns out to be fairly simple, it is not surprising that we obtain the optimal mechanism for specific and random starting points (indeed, the optimal design was produced from every random starting point we generated).

**Minimize Expected (Ex-Ante) Difference in Utility** Here we modify the objective function slightly as compared to the previous section, and instead aim to minimize the expected ex ante difference in utility:

$$\begin{aligned}
& E|u(\text{winner}) - u(\text{loser})| = \\
& E|u(t, s(t), t', s(t'), k, h | a > a') \\
& \quad - u(t, s(t), t', s(t'), k, h | a < a')|.
\end{aligned} \tag{4}$$

While the only difference from the previous section is the placement of the absolute value sign inside the expectation, this difference complicates the analytic derivation of the optimal design considerably. Therefore, we do not present the actual optimum design values.

Parameters	Initial Design	Approx. Optimal Design
$h$	0.5	0.49
$k$	0	1
objective	0.22	0.176
$h$	random	0.29
$k$	random	0.83
objective	N/A	0.176

Table 2: Design that approximately minimizes expected ex ante difference between utility of winner and loser when the optimization search starts at a random and a fixed starting points.

The results of application of our AMD framework are presented in Table 2. While the objective function in this example appears somewhat complex, it turns out (as we discovered through additional exploration) that there are many mechanisms that yield nearly optimal objective values.<sup>5</sup> Thus, both random restarts as well as a fixed starting point produced essentially the same near-optima. By comparison, the truthful design yields the objective value of about 0.22, which is considerably worse.

**Maximize Expected Utility of the Winner** Yet another objective in the shared-good-auction domain is to maximize the expected utility of the winner.<sup>6</sup> Formally, the designer is maximizing  $E[u(t, s(t), t', s(t'), k, h | a > a')]$ .

We first analytically derive the characterization of optimal mechanisms.

**Theorem 6.** *The problem is equivalent to finding  $h$  and  $k$  that maximize  $4/9 - \frac{k}{18(h+k)}$ . Thus,  $k = 0$  and  $h > 0$  maximize the objective, and the optimum is  $4/9$ .*

<sup>5</sup>Particularly, we carried out a far more intensive exploration of the search space given the analytic expression for the Bayes-Nash equilibrium to ascertain that the values reported are close to actual optima. Indeed, we failed to improve on these.

<sup>6</sup>For example, the designer may be interested in minimizing the amount of money which changes hands, which is, by construction, an equivalent problem.

Parameters	Initial Design	Approx. Optimal Design
$h$	0.5	0.21
$k$	0	0
objective	4/9	4/9
$h$	random	0.91
$k$	random	0.03
objective	N/A	0.443

Table 3: Design that approximately maximizes the winner’s expected utility.

Here again our results in Table 3 are optimal or very nearly optimal, unsurprisingly for this relatively simple application.

**Maximize Expected Utility of the Loser** Finally, we try to maximize the expected utility of the loser.<sup>7</sup> Formally, the designer is maximizing

$$E[u(t, s(t), t', s(t'), k, h | a < a')].$$

We again first analytically derive the optimal mechanism.

**Theorem 7.** *The problem is equivalent to finding  $h$  and  $k$  that maximize  $2/9 + \frac{k}{18(h+k)}$ . Thus,  $h = 0$  and  $k > 0$  maximize the objective, and the optimum is  $5/18$ .*

Parameters	Initial Design	Approx. Optimal Design
$h$	0.5	0
$k$	0	0.4
objective	2/9	5/18
$h$	random	0.13
$k$	random	1
objective	N/A	0.271

Table 4: Design that approximately maximizes the loser’s expected utility.

Table 4 shows the results of running AMD in black-box mode in this setting. We can observe that our results are again either actually optimal when the search used a fixed starting point, or close to optimal when random starting points were used. While this design problem is relatively easy and the answer can be analytically derived, the objective function is non-linear, which, along with the presence of noise, adds sufficient complexity to blind optimization to suggest that our success here is at least somewhat interesting.

<sup>7</sup>For example, the designer may be interested in maximizing the amount of money which changes hands, which is, by construction, an equivalent problem.

### 4.1.2 Robust Mechanism Design Problems

**Minimize Nearly-Maximal Difference in Utility** Here, we study the problem of probably approximately robust design for maximal difference in players' utility. The robust formulation of this problem is to minimize

$$\sup_{t, t' \in T} |u(t, s(t), t', s(t'), k, h | a > a') - u(t, s(t), t', s(t'), k, h | a < a')|.$$

**Theorem 8.** *The robust problem is equivalent to finding  $h, k$  that minimize*

$$\frac{h + 2k}{3(h + k)}.$$

*Thus,  $k = 0$  is optimal for any  $h > 0$ , and the optimal value is  $1/3$ .*

Parameters	Initial Design	Approx. Optimal Design
$h$	random	0.01
$k$	random	0
objective	N/A	1/3

Table 5: Design that approximately robustly minimizes the difference in utility.

As we can see from the results in Table 5, our mechanism produced via the automated framework is optimally robust, as the optimum corresponds to one of the robust designs in Theorem 8.

**Maximize Nearly-Minimal Utility of the Winner** The second problem in robust design we consider is maximization of minimum utility of the winner given the type distribution on unit support set. This problem can be more formally expressed as maximizing

$$\inf_{t, t' \in T} u(t, s(t), t', s(t'), k, h | a > a').$$

**Theorem 9.** *The problem is equivalent to finding  $h, k$  that minimize*

$$\frac{k}{6(h + k)}.$$

*Thus,  $k = 0$  is optimal for any  $h > 0$ , with the optimal value of  $0$ .*

Table 6 shows the results of optimizing this objective function using our automated mechanism design framework. As in the previous robust application of our framework, our design is optimally robust according to Theorem 9.

Parameters	Initial Design	Approx. Optimal Design
$h$	random	0.65
$k$	random	0
objective	N/A	0

Table 6: Design that approximately robustly maximizes the winner’s utility.

**Maximize Nearly-Minimal Utility of the Loser** The final robust design problem we will consider for the shared-good auction domain is that of robustly maximizing the utility of the loser. More formally, this is expressed as maximizing

$$\inf_{t, t' \in T} u(t, s(t), t', s(t'), k, h | a < a').$$

**Theorem 10.** *The problem is equivalent to finding  $h, k$  that maximize*

$$\frac{k}{6(h+k)}.$$

*Thus,  $h = 0$  is optimal for any  $k > 0$ , with the optimal value of  $1/6$ .*

Parameters	Initial Design	Approx. Optimal Design
$h$	random	0
$k$	random	0.21
objective	N/A	1/6

Table 7: Design that approximately robustly maximizes the loser’s utility.

According to Table 7, we again observe that our automated process arrived at the optimal robust mechanism, as described in Theorem 10.

Of the examples we considered so far, most turned out to be analytic, and only one we could only approach numerically. Nevertheless, even in the analytic cases, the objective function forms were not trivial, particularly from a blind optimization perspective. Furthermore, we must take into account that even the simple cases are somewhat complicated by the presence of noise, and thus we need not arrive at global optima even in the simplest of settings so long as the number of samples is not very large.

Having found success in the simple shared-good auction setting, we now turn our attention to a series of considerably more difficult problems.

## 4.2 Myerson Auctions

The seminal paper by Myerson [1981] presented a theoretical derivation of revenue maximizing auctions in a relatively general setting. Here, our aim is to find a mechanism with a nearly-optimal value of some given objective function, of which revenue is an example. However, we restrict ourselves to a considerably

less general setting than did Myerson, constraining our design space to that described by the parameters  $q, k_1, k_2, K_1, k_3, k_4, K_2$  in (5).

$$u(t, a, t', a') = \begin{cases} U_1 & \text{if } a > a' \\ 0.5(U_1 + U_2) & \text{if } a = a' \\ U_2 & \text{if } a < a', \end{cases} \quad (5)$$

where  $U_1 = qt - k_1a - k_2a' - K_1$  and  $U_2 = (1 - q)t - k_3a - k_4a' - K_2$ . We further constrain all the design parameters to be in the interval  $[0,1]$ . In standard terminology, our design space allows the designer to choose an allocation parameter,  $q$ , which determines the probability that the winner (i.e., agent with the winning bid) gets the good, and transfers, which we constrain to be linear in agents' bids.

While our automated mechanism design framework assures us that  $p$ -strong individual rationality will hold with the desired confidence, we can actually verify it by hand in this application. Furthermore, we can adjust the mechanism to account for lapses in individual rationality guarantees for subsets of agent types by giving to each agent the amount of the expected loss of the least fortunate type.<sup>8</sup> Similarly, if we do find a mechanism that is Ex-Interim-IR, we may still have an opportunity to increase expected revenue as long as the minimum expected gain of any type is strictly greater than zero.<sup>9</sup>

#### 4.2.1 Bayesian Mechanism Design Problems

**Maximize Revenue** In this section, we are interested in finding approximately revenue-maximizing designs in our constrained design space. Based on Myerson's feasibility constraints, we derive in the following theorem that an optimal incentive compatible mechanism in our design space yields revenue of  $1/3$  to the designer,<sup>10</sup> as compared to  $0.425$  in the general two-player case.<sup>11</sup>

**Lemma 11.** *The mechanism in the design space described by the parameters in equation 5 is BNIC and Ex-Interim-IR if and only if  $k_3 = k_4 = K_1 = K_2 = 0$  and  $q - k_1 - 0.5k_2 = 0.5$ .*

**Theorem 12.** *Optimal incentive compatible mechanism in our setting yields the revenue of  $1/3$ , which can be achieved by selecting  $k_1 \in [0, 0.5]$  and  $k_2 \in [0, 1]$ , respecting the constraint that  $k_1 + 0.5k_2 = 0.5$ .*

In addition to performing five restarts from random starting points, we repeated the simulated annealing procedure starting with the best design produced via the random restarts. This procedure yielded the design in Table 8.

<sup>8</sup>Observe that such constant transfers will not affect agent incentives.

<sup>9</sup>We do not explore computational techniques for either fixing individual rationality or exploiting strictly positive player surplus in this work. We imagine, however, that given a working design many approaches that perform extensive exploration of player surplus in the players' type spaces could be adequate.

<sup>10</sup>For example, Vickrey auction will yield this revenue.

<sup>11</sup>The optimal mechanism prescribed by Myerson is not implementable in our design space.

Parameters	Initial Design	Approx. Optimal Design
$q$	random	0.96
$k_1$	random	0.95
$k_2$	random	0.84
$K_1$	random	0.78
$k_3$	random	0.73
$k_4$	random	0
$K_2$	random	0.53
objective	N/A	0.3

Table 8: Design that approximately maximizes the designer’s revenue.

We used the RW solver to find a symmetric equilibrium of this design, under which the bids are  $s(t) = 0.72t - 0.73$ .<sup>12</sup> We now verify the Ex-Interim-IR and revenue properties of this design.

**Proposition 13.** *The design described in Table 8 is Ex-Interim-IR and yields the expected revenue of approximately 0.3. Furthermore, the designer could gain an additional 0.0058 in expected revenue without effect on incentives while maintaining the individual rationality constraint.*

We have already shown that the best known design, which is also the optimal incentive compatible mechanism in this setting, yields a revenue of 1/3 to the designer. Thus, our AMD framework produced a design near to the best known. It is an open question what the actual global optimum is.

**Maximize Welfare** It is well known that the Vickrey auction is welfare-optimal. Thus, we know that the welfare optimum is attainable in our design space. Before proceeding with search, however, we must make one observation. While we are interested in welfare, it would be inadvisable in general to completely ignore the designer’s revenue, since the designer is unlikely to be persuaded to run a mechanism at a disproportionate loss. To illustrate, take the same Vickrey auction, but afford each agent one billion dollars for participating. This mechanism is still welfare-optimal, but seems a senseless waste if optimality could be achieved without such spending (and, indeed, at some profit to the auctioneer). To remedy this problem, we use a minimum revenue constraint, ensuring that no mechanism that is too costly will be selected as optimal.

First, we present a general result that characterizes welfare-optimal mechanisms in our setting.

**Theorem 14.** *Welfare is maximized if either the equilibrium bid function is strictly increasing and  $q = 1$  or the equilibrium bid function is strictly decreasing and  $q = 0$ . Furthermore, the maximum expected welfare in our design space is  $2/3$ .*

<sup>12</sup>This equilibrium is approximate in the sense that we rounded the parameters to the nearest hundredth.

Thus, for example, both first- and second-price sealed bid auctions are welfare-optimizing (as is well known).

In Table 9 we present the result of our search for optimal design with 5 random restarts, followed by another run of simulated annealing that uses the best outcome of 5 restarts as the starting point. We verified using the RW

Parameters	Initial Design	Approx. Optimal Design
$q$	random	1
$k_1$	random	0.88
$k_2$	random	0.23
$K_1$	random	0.28
$k_3$	random	0.06
$k_4$	random	0.32
$K_2$	random	0
objective	N/A	2/3

Table 9: Design that approximately maximizes welfare.

solver that the bid function  $s(t) = 0.645t - 0.44$  is an equilibrium given this design. Since it is strictly increasing in  $t$ , we can conclude based on Theorem 14 that this design is welfare-optimal. We only need to verify then that both the minimum revenue and the individual rationality constraints hold.

**Proposition 15.** *The design described in Table 9 is Ex-Interim-IR, welfare optimal, and yields the revenue of approximately 0.2. Furthermore, the designer could gain an additional 0.128 in revenue (for a total of about 0.33) without affecting agent incentives or compromising individual rationality and optimality.*

It is interesting that this auction, besides being welfare-optimal, also yields a slightly higher revenue to the designer than our mechanism in the previous section if we implement the modification proposed in Proposition 15. Thus, there appears to be some synergy between optimal welfare and optimal revenue in our design setting.

#### 4.2.2 Robust Mechanism Design Problems

**Maximize Nearly-Minimal Revenue** Our robust objective in this section is to maximize minimal revenue to the designer over the entire joint type space. That is, the robust objective function is

$$\begin{aligned}
& \inf_{t,t' \in T | s(t) > s(t')} [k_1 s(t) + k_2 s(t') + k_3 s(t') + k_4 s(t)] + \\
& \inf_{t,t' \in T | s(t) < s(t')} [k_1 s(t') + k_2 s(t) + k_3 s(t) + k_4 s(t')] + \quad (6) \\
& \inf_{t,t' \in T | s(t) = s(t')} [(k_1 + k_2 + k_3 + k_4) s(t)] + K_1 + K_2.
\end{aligned}$$

Assuming symmetry, here is a simple result about a set of mechanisms that yields 0 for the objective in Equation 6.

**Theorem 16.** *Any auction with  $K_1 = K_2 = 0$  which induces equilibrium strategies in the form  $s(t) = mt$  with  $m > 0$  yields 0 as the value of the objective in Equation 6.*

Thus, both first-price and second-price sealed-bid auctions result in the value of 0 for the robust objective. Furthermore, by Lemma 11 it follows that the same is true for *any* BNIC and ex-post individually rational mechanism in our design space.

Since it is far from clear what the actual optimum for this problem or for its probably approximately robust equivalent is, we ran our automated framework to obtain an approximately optimal design. In Table 10 we show the approxi-

Parameters	Initial Design	Approx. Optimal Design
$q$	random	1
$k_1$	random	1
$k_2$	random	0.34
$K_1$	random	0.69
$k_3$	random	0
$k_4$	random	0
$K_2$	random	0
objective	N/A	0.0066

Table 10: Design that approximately robustly maximizes revenue.

mately optimal mechanism that results. We now verify its individual rationality and revenue properties.

**Proposition 17.** *The mechanism in Table 10 yields the value of 0.0066 for the robust objective. While it is not ex-interim individually rational, it can be made so by paying each agent a fixed 0.000022, resulting in the adjusted robust objective value above 0.0065.*

Thus, we confirm that while not precisely individually rational, our mechanism is very nearly so, and with a small adjustment becomes individually rational with little cost to the designer. Furthermore, the designer is able to make a positive (albeit small) profit no matter what the joint type of the agents is.

### 4.3 Vicious Auctions

In this section we study a design problem motivated by the Vicious Vickrey auction [Brandt and Weiß, 2001, Brandt et al., 2005, Morgan et al., 2003, Reeves, 2005]. The essence of this auction is that while it is designed exactly like a regular Vickrey auction, the players get disutility from the utility of the other player, which is a function of parameter  $l$ , with the regular Vickrey auction the special case of  $l = 0$ .

We generalize the Vicious Vickrey auction design using the same parameters as in the previous section such that the Vicious Vickrey auction is a special case

with  $q = k_2 = 1$  and  $k_1 = k_2 = k_3 = k_4 = K_1 = K_2 = 0$ , and the utility function of agents presented in the previous section can be recovered when  $l = 0$ . We assume in this construction that payments, which will be the same (as functions of players' bids and design parameters) as in the Myerson auction setting, have a particular effect on players' utility parametrized by  $l$ . Hence, the utility function in (7).

$$u(t, a, t', a') = \begin{cases} U_1 & \text{if } a > a' \\ 0.5(U_1 + U_2) & \text{if } a = a' \\ U_2 & \text{if } a < a' \end{cases} \quad (7)$$

where  $U_1 = q(1-l)t - (k_1(q(1-l) + (1-q)) - (1-q)l)a - ((1-q)l)t' - k_2(q(1-l) + (1-q))a' - K_1$  and  $U_2 = (1-q)(1-l)t - (k_3((1-q)(1-l) + q) - ql)a - qlt' - k_4((1-q)(1-l) + q)a' - K_2$ . In all the results below, we fix  $l = 2/7$ . Reeves [2005] reports an equilibrium for Vicious Vickrey with this value of  $l$  to be  $s(t) = (7/9)t + 2/9$ . Thus, we can see that we are no longer assured incentive compatibility even in the second-price auction case. In general, it is unclear whether there exist incentive compatible mechanisms in this design space, particularly because we constrain all our parameters to lie in the interval  $[0, 1]$ .

Before proceeding, we would like to modify the definition of individual rationality in this setting to be as follows: every agent can earn non-negative expected value less expected payment (that is, expected surplus). To formalize this,

$$EU(t) = v(t) - m(t) \geq 0,$$

where  $v(t)$  is the expected value to agent with type  $t$  and  $m(t)$  is the expected payment to the auctioneer by the agent with type  $t$ . This is in contrast with assuring each agent that every type will obtain non-negative expected utility. However, we believe that the alternative definition is more sensible in this setting, since it assures the agent that it will receive no less than the opportunity cost of participation in the auction, which we take to be zero.

### 4.3.1 Bayesian Mechanism Design Problems

**Maximize Revenue** Our first objective is to (nearly) maximize revenue in this domain. The results of automated mechanism design in two distinct cases are presented in Table 11.

The top part of Table 11 presents the results of simulated annealing search that uses the previously studied Vicious Vickrey as a starting point. Our purpose for doing so is two-fold. First, we would like to see if we can easily (i.e., via an automated process) do better than the previously studied mechanism. Second, we want to suggest automated mechanism design as a framework not only for finding good mechanisms from scratch, but also for improving mechanisms that are initially designed by hand. The latter could become especially useful in practice when applications are extremely complex and we can use theory and intuition to give us a good starting mechanism.

Parameters	Initial Design	Approx. Optimal Design
$q$	1	1
$k_1$	0	0
$k_2$	1	0.98
$K_1$	0	0.09
$k_3$	0	0.33
$k_4$	0	0
$K_2$	0	0
objective	0.48	0.49
$q$	random	1
$k_1$	random	1
$k_2$	random	0.33
$K_1$	random	0.22
$k_3$	random	0.22
$k_4$	random	0.12
$K_2$	random	0
objective	N/A	0.44

Table 11: Design that approximately maximizes revenue.

First, we determine the expected revenue and individual rationality properties of the Vicious Vickrey auction in the following Proposition.

**Proposition 18.** *The expected revenue from Vicious Vickrey auction with  $l = 2/7$  is approximately 0.48. This auction is not ex-post individually rational, but can be adjusted by awarding each agent 0.021. The adjusted revenue would become 0.0438.*

We now give the individual rationality and revenue properties of the auction that AMD obtains with Vicious Vickrey as the starting point.

**Proposition 19.** *The expected revenue from the auction  $\{1, 0, 0.98, 0.09, 0.33, 0, 0\}$  in Table 11 is approximately 0.49. This auction is ex post individually rational, and will remain so if the designer charges a fixed entry fee of 0.0027, giving itself a total revenue of approximately 0.4932.*

Thus, we found a design which yields more revenue than the design previously studied in the literature (adjusted to be individually rational).

Now, we assume that we have never heard of Vicious Vickrey and need to find a good mechanism without any additional information. Consequently, we present results of search from a random starting point in the lower section of Table 11. Properties of the resulting auction are explored in Proposition 20.

**Proposition 20.** *The expected revenue from the auction  $\{1, 1, 0.33, 0.22, 0.22, 0.12, 0\}$  in Table 11 is approximately 0.44. This auction is ex post individually rational, and can remain so if the designer charges all agents an additional fixed participation fee of 0.0199. This design change would increase the expected revenue to 0.4798.*

Thus, the design we obtained from a completely random starting point yields revenue that is not far below that of Vicious Vickrey (or the design that we found using Vicious Vickrey as a starting point), and is better than Vicious Vickrey if the latter is adjusted to be individually rational. Furthermore, this design can be improved considerably via a participation tax without sacrificing individual rationality.

Parameters	Initial Design	Approx. Optimal Design
$q$	random	0.37
$k_1$	random	0.8
$k_2$	random	1
$K_1$	random	0.49
$k_3$	random	0.29
$k_4$	random	0.67
$K_2$	random	0.48
objective	N/A	0.54

Table 12: Design that approximately maximizes welfare.

**Maximize Welfare** In Table 12 we present an outcome of the automated mechanism design process with the goal of maximizing welfare. The procedure, as above, involves simulated annealing with five random restarts, and an additional run with the current optimum welfare as the starting point of the simulated annealing run. In the optimization, we utilized both the Ex-Interim-IR and minimum revenue constraints. In the following proposition we establish the welfare, revenue, and individual rationality properties of this mechanism.

**Proposition 21.** *The expected welfare of the mechanism in Table 12 is approximately 0.54 and expected revenue is approximately 0.225. It is ex post individually rational for all types in  $[0.17, 1]$  and can be made individually rational for every type at an additional loss of 0.13 in revenue.*

While individual rationality does not hold for almost 80% of types, this failure is easy to remedy at some additional loss in revenue (importantly, the adjusted expected revenue will be positive).

After a sequence of successful applications of AMD, we stand before an evident failure: the mechanism we found is quite a bit below the known optimum of  $2/3$ . Interestingly, recall that the optimal revenue mechanism in the vicious setting had a strictly increasing bid function and  $q = 1$ , and consequently was also welfare-optimal by Theorem 14.

Instead of plainly dismissing this application as a failure, we can perhaps derive some lessons as to why our results were so poor. We hypothesize that the most important reason is that we introduced minimum revenue of 0 as an additional hard constraint. From observing the optimization runs in general, we notice that the optimization problem both in the Myerson auctions and the vicious auctions design space seems to be rife with islands of local optima in the

sea of infeasibility. Thus, the problem was difficult for black-box optimization already, and we only made it considerably more difficult by adding additional infeasible regions. In general, we would expect such optimization techniques to work best when the objective function varies smoothly and most of the space is feasible. Hard constraints make it more difficult by introducing (at least in our implementation) spikes in the objective value.<sup>13</sup>

We have seen some evidence to the correctness of our hypothesis already, since our revenue-optimal design also happens to maximize social utility. To test our hypothesis directly, we remove minimum revenue as a hard constraint in the next section, and instead try to maximize the weighted sum of welfare and revenue.

**Maximize Weighted Sum of Revenue and Welfare** In this section, we present results of AMD with the goal of maximizing the weighted sum of revenue and welfare. For simplicity (and having no reason for doing otherwise), we set weights to be equal. A design that our framework found from a random starting

Parameters	Initial Design	Approx. Optimal Design
$q$	random	1
$k_1$	random	0.51
$k_2$	random	1
$K_1$	random	0.09
$k_3$	random	0.34
$k_4$	random	0.26
$K_2$	random	0
objective	N/A	0.6372

Table 13: Design that approximately maximizes the average of welfare and revenue.

point is presented in Table 13. We verified using RW that  $s(t) = 0.935t - 0.18$  is an (approximate) symmetric equilibrium bid function. Thus, by Theorem 14 this auction is welfare-optimal.

**Proposition 22.** *The expected revenue from the auction in Table 13 is 0.6078. However, it is not ex interim individually rational, and the least fortunate type loses nearly 0.044. However, by compensating the agents the designer can induce individual rationality without affecting incentives, at a revenue loss of 0.088. This would leave it with an adjusted expected revenue of 0.5198.*

### 4.3.2 Robust Mechanism Design Problems

**Maximize Nearly-Minimal Revenue** We now apply our framework to the problem of robustly maximizing revenue of the designer. First, we present the

<sup>13</sup>Recall that we implemented hard constraints as a very low value of the objective. Thus, adding hard constraints increases nonlinearity of the objective function, and the increase could be quite dramatic.

result for the previously studied Vicious Vickrey auction.

**Proposition 23.** *By running the Vicious Vickrey auction, the designer can obtain at least  $2/9$  (approximately 0.22) in revenue for any joint type profile. By adjusting to make the auction individually rational, minimum revenue falls to  $220/1089$  (approximately 0.2).*

The results from running our automated design framework from a random starting point are shown in Table 14. We now verify the revenue and individual

Parameters	Initial Design	Approx. Optimal Design
$q$	random	0.86
$k_1$	random	1
$k_2$	random	0.71
$K_1$	random	0.14
$k_3$	random	0
$k_4$	random	0.09
$K_2$	random	0
objective	N/A	0.059

Table 14: Design that approximately robustly maximizes revenue.

rationality properties of this mechanism.

**Proposition 24.** *The design in Table 14 yields revenue of at least 0.059 to the designer for any agent type profile, but is not ex-interim individually rational. It can be made such if the designer awards each agent 0.0135 for participation, yielding the adjusted revenue of 0.032.*

As we can see, our randomly generated design is considerably worse than the adjusted Vicious Vickrey. However, adjusted Vicious Vickrey requires negative settings of several of our design parameters. Since our parameters are initially constrained to be non-negative, it is unclear whether a better solution is indeed attainable in our constrained design space, even at a slight ( $< 0.02$ ) sacrifice in individual rationality.

## 5 Related Work

The mechanism design literature in economics has typically explored existence of a mechanism that implements a social choice function in equilibrium [Mas-Colell et al., 1995]. Additionally, there is an extensive literature on optimal auction design [Mas-Colell et al., 1995], of which the work by Myerson [1981] is perhaps the most relevant. In much of this work, analytic results are presented with respect to specific utility functions and accounting for constraints such as incentive compatibility and individual rationality.

Several related approaches to search for the best mechanism exist in the computer science literature. Conitzer and Sandholm [2004] developed a search

algorithm for designing incentive compatible mechanisms given finite mechanism spaces and finite one-player games. More recently, Conitzer and Sandholm [2007] proposed an incremental approach to designing strategyproof mechanisms, and Sandholm et al. [2007] present a framework for designing multistage mechanisms.

When payoff functions of players are unknown, a search using simulations has been explored as an alternative. One approach in that direction, taken by Cliff [2002] and Phelps et al. [2002], is to co-evolve the mechanism parameter and agent strategies, using some notion of social welfare and agent payoffs as fitness criteria. An alternative to co-evolution explored in another paper by Phelps et al. [2003] is to optimize a well-defined welfare function of the designer using genetic programming. In this work the authors used a common learning strategy for all agents and defined an outcome of a game induced by a mechanism parameter as the outcome of multi-agent learning. In another paper, Phelps et al. [2004] compared two mechanisms based on expected social welfare with expectation taken over an empirical distribution of equilibria in games defined by heuristic strategies, as in Walsh et al. [2002]. Recently, Pardoe et al. [2006] introduced a metalearning approach for adaptively designing auction mechanisms based on empirical bidder behavior. The mechanisms designed thereby are actually learning rules that adapt to observed bidder behavior, and the metalearning technique is used to find learning parameters.

## 6 Conclusion

We presented a framework for automated indirect mechanism design using the Bayes-Nash equilibrium solver for infinite games developed by Reeves and Wellman [2004], Reeves [2005]. Results from applying this framework to several design domains demonstrate the value of our approach for practical mechanism design. The mechanisms that we found were typically either close to the best known mechanisms, or better. Our lone failure illuminated the difficulty of the automated mechanism design problem when too many hard constraints are present. After modifying the problem by eliminating the hard minimum revenue constraint and using multiple weighted objectives instead, we were able to find a mechanism with the best values of *both* objectives yet seen.

While in principle it is not at all surprising that we can find mechanisms by searching the design space—as long as we have an equilibrium finding tool—it was not at all clear that any such system will have practical merit. We presented evidence that indirect mechanism design can indeed be effectively automated on somewhat realistic design problems. Undoubtedly, real design problems are vastly more complicated than any that we considered (or any that can be considered theoretically). In such cases, we believe that our approach could offer considerable benefit if used in conjunction with other techniques, either to provide a starting point for design, or to tune a mechanism produced via theoretical analysis and computational experiments.

## References

- Aharon Ben-Tal and Arkadi Nemirovski. Robust optimization – methodology and applications. *Mathematical Programming*, 92:453–480, 2002.
- Felix Brandt and Gerhard Weiß. Antisocial agents and Vickrey auctions. In *Eighth International Workshop on Agent Theories, Architectures, and Languages*, volume 2333 of *Lecture Notes in Computer Science*, pages 335–347, Seattle, 2001. Springer.
- Felix Brandt, Tuomas Sandholm, and Yoav Shoham. Spiteful bidding in sealed-bid auctions. Technical report, Stanford University, 2005.
- Dave Cliff. Evolution of market mechanism through a continuous space of auction-types. In *Congress on Evolutionary Computation*, 2002.
- Vincent Conitzer and Tuomas Sandholm. Applications of automated mechanism design. In *UAI-03 Bayesian Modeling Applications Workshop*, 2003a.
- Vincent Conitzer and Tuomas Sandholm. Computational criticisms of the revelation principle. In *Workshop on Agent Mediated Electronic Commerce-V*, 2003b.
- Vincent Conitzer and Tuomas Sandholm. An algorithm for automatically designing deterministic mechanisms without payments. In *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 128–135, 2004.
- Vincent Conitzer and Tuomas Sandholm. Incremental mechanism design. In *International Joint Conference on Artificial Intelligence*, 2007.
- A. Corana, M. Marchesi, C. Martini, and S. Ridella. Minimizing multimodal functions of continuous variables with simulated annealing algorithm. *ACM Transactions on Mathematical Software*, 13(3):262–280, 1987.
- Mark Fleischer. Simulated Annealing: Past, present, and future. In *Winter Simulation Conference*, pages 155–161, 1995.
- Paul Klemperer. *Auctions: Theory and Practice*. Princeton University Press, 2004.
- Vijay Krishna. *Auction Theory*. Academic Press, 1st edition, 2002.
- Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green. *Microeconomic Theory*. Oxford University Press, New York, 1995.
- Richard D. McKelvey, Andrew M. McLennan, and Theodore L. Turocy. Gambit: Software tools for game theory, version 0.2005.06.13, 2005. URL <http://econweb.tamu.edu/gambit>.

- John McMillan. Selling spectrum rights. *The Journal of Economic Perspectives*, 8(3):145–162, 1994.
- John Morgan, Ken Steiglitz, and George Reis. The spite motive and equilibrium behavior in auctions. *Contributions to Economic Analysis and Policy*, 2(1), 2003.
- Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, February 1981.
- Sigurdur Olafsson and Jumi Kim. Simulation optimization. In E. Yucesan, C.-H. Chen, J.L. Snowdon, and J.M. Charnes, editors, *2002 Winter Simulation Conference*, 2002.
- David Pardoe, Peter Stone, Mayal Saar-Tsechansky, and Kerem Tomak. Adaptive mechanism design: a metalearning approach. In *Eight International Conference on Electronic Commerce*, 2006.
- Steve Phelps, Simon Parsons, Peter McBurney, and Elizabeth Sklar. Co-evolution of auction mechanisms and trading strategies: towards a novel approach to microeconomic design. In *ECOMAS 2002 Workshop*, 2002.
- Steve Phelps, Simon Parsons, Elizabeth Sklar, and Peter McBurney. Using genetic programming to optimise pricing rules for a double-auction market. In *Workshop on Agents for Electronic Commerce*, 2003.
- Steve Phelps, Simon Parsons, and Peter McBurney. Automated agents versus virtual humans: an evolutionary game theoretic comparison of two double-auction market designs. In *Workshop on Agent Mediated Electronic Commerce VI*, 2004.
- Daniel M. Reeves. *Generating Trading Agent Strategies: Analytic and Empirical Methods for Infinite and Large Games*. PhD thesis, University of Michigan, 2005.
- Daniel M. Reeves and Michael P. Wellman. Computing best-response strategies in infinite games of incomplete information. In *Twentieth Conference on Uncertainty in Artificial Intelligence*, pages 470–478, 2004.
- Tuomas Sandholm, Vincent Conitzer, and Craig Boutilier. Automated design of multistage mechanisms. In *International Joint Conference on Artificial Intelligence*, 2007.
- Patrick Siarry, Gerard Berthiau, Francois Durbin, and Jacques Haussy. Enhanced simulated annealing for globally minimizing functions of many continuous variables. *ACM Transactions on Mathematical Software*, 23(2):209–228, 1997.
- Yevgeniy Vorobeychik, Christopher Kiekintveld, and Michael P. Wellman. Empirical mechanism design: Methods, with an application to a supply chain scenario. In *ACM E-Commerce*, pages 306–315, 2006.

William E. Walsh, Rajarshi Das, Gerald Tesauero, and Jeffrey O. Kephart. Analyzing complex strategic interactions in multi-agent systems. In *AAAI-02 Workshop on Game Theoretic and Decision Theoretic Agents*, 2002.

## Appendix

### A Proofs

#### A.1 Proof of Theorem 1

Suppose  $p$  is the probability measure of  $T_A$  and suppose we select the best  $\theta_i$  of  $\{\theta_1, \dots, \theta_L\}$ . Suppose further that we take  $n$  samples for each  $\theta_j$ , and let  $T^n$  be the set of  $n$  type realizations. We will also use the notation  $\theta \in G$  to indicate an event that for a particular  $\theta$ ,  $\min_{t \in T^n} W(r, t, \theta) > \inf_{t \in T \setminus T_A} W(r, t, \theta)$ .

We would like to compute the number of samples  $n$  for each of these samples such that  $P\{\theta_i \notin G\} \geq 1 - \alpha$ .

Note that

$$P\{\theta_i \notin G\} \geq P\{\theta_1 \notin G \& \dots \& \theta_L \notin G\} = P\{\theta_j \notin G\}^L.$$

Now,

$$P\{\theta_j \in G\} = P\{t_1 \notin T_A \& \dots \& t_n \notin T_A\} = P\{t_i \notin T_A\}^n = (1 - p)^n.$$

Thus,

$$P\{\theta_i \notin G\} \geq (1 - (1 - p)^n)^L = 1 - \alpha.$$

Solving for  $n$ , we obtain the desired answer.

#### A.2 Proof of Theorem 2

Note that  $\alpha$  is just the probability that the actual measure  $r$  of set  $B$  is below  $p$  if none of  $n$  i.i.d. samples  $X_i$  from the type distribution violated the constraint:

$$\alpha = \Pr\{r \leq p \mid \forall i = 1, \dots, n, X_i \notin B\} = \frac{\Pr\{\forall i = 1, \dots, n, X_i \notin B \& r \leq p\}}{\Pr\{\forall i = 1, \dots, n, X_i \notin B\}}.$$

Since the samples are i.i.d.,

$$\Pr\{\forall i = 1, \dots, n, X_i \notin B \mid r\} = r^n,$$

and since we assumed a uniform prior on  $r$ , we get

$$\Pr\{\forall i = 1, \dots, n, X_i \notin B\} = \int_0^1 r^n dr = \frac{1}{n+1}$$

and

$$\Pr\{\forall i = 1, \dots, n, X_i \notin B \& r \leq p\} = \int_0^p r^n dr = \frac{p^{n+1}}{n+1}.$$

Consequently, we obtain the following relationship between  $\alpha$ ,  $p$ , and  $n$ :

$$\alpha = p^{n+1}.$$

Solving for  $n$ , we get

$$n = \frac{\log \alpha}{\log p} - 1.$$

### A.3 Proof of Theorem 3

We show that for the two-player game with types  $U[A, B]$  and payoff function

$$u(t, a, t', a') = \begin{cases} t - ha - ka' & \text{if } a > a' \\ \frac{t - ha - ka' + ha' + ka}{2} & \text{if } a = a' \\ ha' + ka & \text{if } a < a', \end{cases}$$

with  $h, k \geq 0$  and  $B \geq A + 1$  that the following is a symmetric Bayes-Nash equilibrium strategy:

$$\frac{t}{3(h+k)} + \frac{hA+kB}{6(h+k)^2}. \quad (8)$$

Consider first the special case that  $h = k = 0$ . Equation 8 prescribes a strategy of bidding  $\infty$  and it is clear that this is a dominant strategy in a game where the winner is the high bidder with no payments required.<sup>14</sup> We will now assume that  $h + k > 0$ .

Define  $m \equiv \frac{1}{3(h+k)}$  and  $c \equiv \frac{hA+kB}{6(h+k)^2}$  and let  $T$  be a random  $U[A, B]$  variable giving the opponent's type. Noting that the tie-breaking case ( $a = a'$ ) happens with zero probability given that (8) is a continuous function of a uniform random variable, we write the expected utility for an agent of type  $t$  playing action  $a$  as

$$\begin{aligned} \text{EU}(t, a) &= E_T[u(t, a, T, mT + c)] \\ &= E[t - ha - k(mT + c) \mid a > mT + c] \Pr(a > mT + c) \\ &\quad + E[h(mT + c) + ka \mid a < mT + c] \Pr(a < mT + c) \\ &= E \left[ t - ha - kmT - kc \mid T < \frac{a-c}{m} \right] \Pr \left( T < \frac{a-c}{m} \right) \\ &\quad + E \left[ hmT + hc + ka \mid T > \frac{a-c}{m} \right] \Pr \left( T > \frac{a-c}{m} \right) \end{aligned} \quad (9)$$

We consider three cases on the range of  $a$  and find the optimal action  $a_i^*$  for each case  $i$ .

<sup>14</sup>This assumes that the space of possible bids includes  $\infty$ . More generally, the dominant strategy is the supremum of the bid space but if this is not itself a member of the bid space (as is the case if the bid space is  $\mathbb{R}$ ) then there is in fact no Nash equilibrium of the game.

*Case 1:*  $a \leq Am + c$ . ( $\implies \frac{a-c}{m} \leq A$ )

The probabilities in (9) are zero and one, respectively, and so the expected utility is:

$$\text{EU}(t, a) = hm \frac{A+B}{2} + hc + ka.$$

This is an increasing function in  $a$ , implying an optimal action at the right boundary:  $a_1^* = Am + c$ . Thus the best expected utility for case 1 is

$$\text{EU}(t, a_1^*) = \frac{2A+B}{6}.$$

*Case 2:*  $a \geq Bm + c$ . ( $\implies \frac{a-c}{m} \leq B$ )

The probabilities in (9) are one and zero, respectively, and so the expected utility is:

$$\text{EU}(t, a) = t - ha - km \frac{A+B}{2} - kc.$$

This is a decreasing function in  $a$ , implying an optimal action at the left boundary:  $a_2^* = Bm + c$ . Thus the best expected utility for case 2 is

$$\text{EU}(t, a_2^*) = t - \frac{A+2B}{6}.$$

*Case 3:*  $Am + c < a < Bm + c$ .

Knowing that  $\frac{a-c}{m}$  is between  $A$  and  $B$  it is straightforward to compute the probabilities in (9) and the conditional expectation of  $T$ . So we write  $\text{EU}(t, a)$  as:

$$\begin{aligned} & \left( t - ha - km \frac{A + \frac{a-c}{m}}{2} - kc \right) \left( \frac{a-c}{m} - A \right) \\ & + \left( hm \frac{B + \frac{a-c}{m}}{2} + hc + ka \right) \left( B - \frac{a-c}{m} \right) \\ = & (-108a^2h^4 - 432a^2kh^3 - 648a^2k^2h^2 - 432a^2k^3h - \\ & - 108a^2k^4 + 36aAh^3 + 72ath^3 + A^2h^2 + 4B^2h^2 + \\ & + 4ABh^2 + 72aAkh^2 + 36aBkh^2 - 36Ath^2 + 216akth^2 + \\ & + 36aAk^2h + 72aBk^2h + 8A^2kh + 8B^2kh + 2ABkh + \\ & + 216ak^2th - 60Akh - 12Bkh + 36aBk^3 + 4A^2k^2 + B^2k^2 \\ & + 4ABk^2 + 72ak^3t - 24Ak^2t - 12Bk^2t) / (24(h+k)^2). \end{aligned}$$

Since this is a concave function of  $a$  the maximum is where the derivative with respect to  $a$  is zero, that is (skipping the tedious algebra for which we used Mathematica):

$$\begin{aligned} & \frac{\partial \text{EU}(t, a)}{\partial a} = 0 \\ \implies & a_3^* = \frac{t}{3(h+k)} + \frac{hA+kB}{6(h+k)^2}. \end{aligned}$$

Since  $A \leq t \leq B \implies Am + c \leq a_3^* \leq Bm + c$ ,  $a_3^*$  is in fact in the allowable range for case 3. The expected utility for case 3 is then

$$\text{EU}(t, a_3^*) = \frac{3t^2 + A^2 + B^2 + A(B - 6t)}{6}.$$

It now remains to show that neither  $\text{EU}(t, a_1^*)$  nor  $\text{EU}(t, a_2^*)$  is greater than  $\text{EU}(t, a_3^*)$  for any  $t$ .

Since  $t \geq A$  there exists a  $\delta \geq 0$  such that  $t = A + \delta$ . And since  $B \geq A + 1$  there exists an  $\varepsilon \geq 0$  such that  $B = A + 1 + \varepsilon$ . First,  $\text{EU}(t, a_3^*) \geq \text{EU}(t, a_2^*)$  because

$$\begin{aligned} & (\delta - 1)^2 \geq 0 \\ \implies & \delta^2 - 2\delta + 1 \geq 0 \\ \implies & \delta^2 + 1 \geq 2\delta \\ \implies & (A + \delta - A)^2 + 2A + 1 \geq 2A + 2\delta \\ \implies & (t - A)^2 + 2A + 1 \geq 2t \\ \implies & t^2 + A^2 + 2A + 1 \geq 2At + 2t \\ \implies & 3t^2 + 3A^2 + 6A + 3 + (3A\varepsilon + \varepsilon^2 + 4\varepsilon) \geq 6At + 6t \\ \implies & 3t^2 + A^2 + (A^2 + 2A + 2A\varepsilon + \varepsilon^2 + 2\varepsilon + 1) + \\ & \quad + (A^2 + A + A\varepsilon) - 6At \geq 6t - A - 2A - 2 - 2\varepsilon \\ \implies & 3t^2 + A^2 + (A + 1 + \varepsilon)^2 + A(A + 1 + \varepsilon) - 6At \\ & \quad \geq 6t - A - 2(A + 1 + \varepsilon) \\ \implies & 3t^2 + A^2 + B^2 + AB - 6At \geq 6t - A - 2B. \end{aligned}$$

Finally,  $\text{EU}(t, a_3^*) \geq \text{EU}(t, a_1^*)$  because

$$\begin{aligned} & (t - A)^2 \geq 0 \\ \implies & t^2 - 2At + A^2 \geq 0 \\ \implies & t^2 + A^2 \geq 2At \\ \implies & 3t^2 + 3A^2 \geq 6At \\ \implies & 3t^2 + 3A^2 + (3A\varepsilon + \varepsilon^2 + \varepsilon) \geq 6At \\ \implies & 3t^2 + 3A^2 + 3A + 3A\varepsilon + \varepsilon^2 + \varepsilon - 6At \geq 3A \\ \implies & 3t^2 + (A^2 + A + \varepsilon) - 6At + \\ & \quad + (A^2 + 2A + 2A\varepsilon + \varepsilon^2 + 2\varepsilon + 1) + A^2 \geq 3A + \varepsilon + 1 \\ \implies & 3t^2 + A(A + 1 + \varepsilon) - 6At + A^2 + (A + 1 + \varepsilon)^2 \\ & \quad \geq 2A + (A + \varepsilon + 1) \\ \implies & 3t^2 + AB - 6At + A^2 + B^2 \geq 2A + B. \end{aligned}$$

#### A.4 Proof of Theorem 4

It is direct from Theorem 3 that setting  $h = 1/3$  and  $k = 0$  yields a symmetric Bayes-Nash equilibrium  $s(t) = t$  when  $A = 0$ . We now show that the best response to truthful bidding is only truthful under this parameter setting—i.e., that SGA(1/3, 0) is the only BNIC game in the SGA family, for  $U[0, B]$  types.

Suppose that the opponent bids truthfully (i.e.,  $s(t) = t$  for one of the agents). First, assume that  $a \in [0, B]$ . The expected utility of an agent with type  $t$  from bidding  $a$  is then

$$\begin{aligned} \text{EU}(t, a) &= \int_0^a (t - ha - kT) dT + \int_a^1 (hT + ka) dT = \\ &= \frac{1}{2} (-3(h+k)a^2 + 2(Bk+t)a + B^2h). \end{aligned}$$

Since this function is strictly concave in  $a$ , we can use the first-order condition to find the optimum bid:

$$\frac{\partial \text{EU}(t, a)}{\partial a} = t - 3(h+k)a + Bk = 0$$

yielding

$$a = \frac{t + Bk}{3(h+k)}, \quad (10)$$

which is truthful for every type  $t$  only when  $h = 1/3$  and  $k = 0$ .

Now, if  $a \leq 0$ , it will always lose, and the expected utility is

$$\text{EU}(t, a) = \int_0^B (hT + ka) dT = B^2h/2 + kB a,$$

which is maximized when  $a = 0$ . Consequently, there is no incentive to ever bid below 0. Similarly, if  $a \geq B$ , the agent will never lose, and

$$\text{EU}(t, a) = \int_0^B (t - ha - kT) dT = -\frac{1}{2}B(2ah + Bk - 2t),$$

which is maximized when  $a = B$ . Thus, there is no incentive to ever bid above  $B$ . All incentive compatible mechanisms will thus induce bidding according to (10). It follows, then, that SGA(1/3, 0) is the only truthful mechanism for  $U[0, B]$  ( $B > 0$ ) types.

#### A.5 Proof of Theorem 5

The objective function in terms of  $h$  and  $k$  is

$$\begin{aligned} \min_{h,k} |E[t_w - 2h(\frac{t_w}{3(h+k)} + \frac{k}{6(h+k)^2}) - 2k(\frac{t_l}{3(h+k)} + \\ \frac{k}{6(h+k)^2}) | t_w > t_l]|. \end{aligned}$$

Since  $E[t_w | t_w > t_l]$  is the expectation of the first order statistic of two  $U[0, 1]$  random variables, it is  $2/3$  (and  $1/3$  for  $t_l$ ). Thus, the objective function above reduces to

$$\min_{h,k} \left| \frac{2h+k}{9(h+k)} \right|.$$

We now show that this expression cannot be less than  $1/9$ :

$$\begin{aligned} h &\geq 0 \\ \implies 2h &\geq h \\ \implies 2h+k &\geq h+k \\ \implies \frac{2h+k}{h+k} &\geq 1 \\ \implies \frac{2h+k}{9(h+k)} &\geq \frac{1}{9}. \end{aligned}$$

Since setting  $h = 0$  yields the minimum of  $1/9$  for any  $k > 0$  we conclude that all mechanisms  $\text{SGA}(0, k)$  minimize the objective function.

## A.6 Proof of Theorem 6

Let  $R$  designate the expected revenue of the winner.

$$\begin{aligned} R &= E[t_w - h(t_w/3(h+k) + k/6(h+k)^2) - \\ &\quad - k(t_l/3(h+k) + k/6(h+k)^2)] = \\ &= E[t_w] - hE[t_w]/3(h+k) - hk/6(h+k)^2 - \\ &\quad - kE[t_l]/3(h+k) - k^2/6(h+k)^2 = \\ &= 2/3 - (4h+5k)/18(h+k) = 4/9 - k/18(h+k). \end{aligned}$$

## A.7 Proof of Theorem 7

Let  $R$  designate the expected revenue of the loser.

$$\begin{aligned} R &= E[h(t_w/3(h+k) + k/6(h+k)^2) + \\ &\quad + k(t_l/3(h+k) + k/6(h+k)^2)] = \\ &= hE[t_w]/3(h+k) + hk/6(h+k)^2 + \\ &\quad + kE[t_l]/3(h+k) + k^2/6(h+k)^2 = \\ &= (4h+5k)/18(h+k) = 2/9 + k/18(h+k). \end{aligned}$$

## A.8 Proof of Theorem 8

First, we obtain the expression to be minimized.

$$\begin{aligned}
& \sup_{t>t'} |t - 2h(\frac{t}{3(h+k)} + \frac{k}{6(h+k)^2}) - 2k(\frac{t'}{3(h+k)} + \frac{k}{6(h+k)^2})| = \\
& \sup_{t>t'} |t - \frac{2ht + 2kt'}{3(h+k)} - \frac{k}{3(h+k)}| = \\
& \sup_{t>t'} |\frac{ht + 3kt - 2kt' - k}{3(h+k)}|.
\end{aligned}$$

Clearly, this is minimized when  $t = 1$  and  $t' = 0$ , yielding

$$\frac{h + 3k - k}{3(h+k)} = \frac{h + 2k}{3(h+k)}.$$

Now, note that since  $h, k \geq 0$ ,

$$\frac{h + 2k}{3(h+k)} \geq \frac{h+k}{3(h+k)} = \frac{1}{3}.$$

Thus, the expression cannot be less than  $1/3$ . Consequently, since setting  $k = 0$  for any  $h > 0$  results in the objective function value of  $1/3$ , it describes a subset of optimal values.

### A.9 Proof of Theorem 9

$$\begin{aligned}
& \inf_{t>t'} [t - h(\frac{t}{3(h+k)} + \frac{k}{6(h+k)^2}) - k(\frac{t'}{3(h+k)} + \frac{k}{6(h+k)^2})] = \\
& \inf_{t>t'} [t - \frac{ht' + kt}{3(h+k)} - \frac{k}{6(h+k)}] = \inf_{t>t'} [\frac{h(t-t')}{3(h+k)} - \frac{k}{6(h+k)}].
\end{aligned}$$

The infimum is equivalent to setting  $t = 0$  and  $t' = 0$ , and thus the expression is maximized if

$$\frac{k}{6(h+k)}$$

is minimized, which is effected by setting  $k = 0$ . The resulting optimal value is 0.

### A.10 Proof of Theorem 10

$$\begin{aligned}
& \inf_{t>t'} [h(\frac{t}{3(h+k)} + \frac{k}{6(h+k)^2}) + k(\frac{t'}{3(h+k)} + \frac{k}{6(h+k)^2})] = \\
& \inf_{t>t'} [\frac{ht' + kt}{h+k} + \frac{k}{6(h+k)}] = \inf_{t>t'} \frac{ht' + kt}{h+k} + \frac{k}{6(h+k)}.
\end{aligned}$$

The infimum is equivalent to setting  $t = 0$  and  $t' = 0$ , and the expression is thus maximized when  $h = 0$  for any  $k > 0$ , with the optimum of  $1/6$ .

### A.11 Proof of Lemma 11

First, let us derive  $Q(q, t)$  and  $U(q, x, t)$ , where  $q$  is the probability that player with the higher type wins the good and  $x(t)$  is the expected payment by players [Myerson, 1981].

$$\begin{aligned}
 Q(q, t) &= \int_0^t q dT + \int_t^1 (1 - q) dT = t(2q - 1) - q + 1. \\
 U(q, x, t) &= \int_0^t (tq - k_1 t - k_2 T - K_1) dT + \\
 &\quad + \int_t^1 ((1 - q)t - k_3 t - k_4 T - K_2) dT = \\
 &= (2q - k_1 - 0.5k_2 + k_3 + 0.5k_4 - 1)t^2 + \\
 &\quad + (1 - q - K_1 - k_3 + K_2)t - (0.5k_4 + K_2).
 \end{aligned}$$

The first constraint that must be satisfied according to Myerson [1981] is if  $s \leq t$  then  $Q(q, s) \leq Q(q, t)$ . This constraint is always satisfied in our design space by inspection of the form of  $Q(q, t)$  above.

Individual rationality constraint requires that  $U(q, x, 0) \geq 0$ , implying in our setting that  $0.5k_4 + K_2 \leq 0$ . Since all design parameters are constrained to be non-negative, this implies that  $k_4 = K_2 = 0$ , and, consequently,  $U(q, x, 0) = 0$ .

The version of the final constraint in Myerson [1981] in our setting

$$U(q, x, t) = \int_0^1 Q(q, s) ds = (q - 0.5)t^2 + (1 - q)t$$

implies that  $K_1 = k_3 = 0$  and  $q - k_1 - 0.5k_2 - 0.5 = 0$ , completing the proof.

### A.12 Proof of Theorem 12

The expected revenue to the designer is

$$U_0(q, x) = \int_0^1 \int_0^1 (x_1(t, T) + x_2(t, T)) dt dT$$

which by symmetry and Lemma 11 is equivalent to

$$U_0(q, x) = 2 \int_0^1 \int_0^t (k_1 t + k_2 T) dT dt = \frac{2}{3}k_1 + \frac{1}{3}k_2.$$

Rewriting the constraint from Lemma 11 to be  $k_1 + 0.5k_2 = q - 0.5$ , it is clear that the revenue is maximal when  $q = 1$ . Now, if we let  $k = k_1$  and  $k_2 = 1 - 2k$ , the expected revenue becomes  $(2/3)k + (1/3)(1 - 2k) = 1/3$ . Thus, we can set any  $k_1 \in [0, 0.5]$  and  $k_2 \in [0, 1]$ , respecting the constraint, to achieve optimal revenue of  $1/3$ .

### A.13 Proof of Proposition 13

We will use the equilibrium bids of  $s(t) = 0.72t - 0.73$  in this proof. First, let us derive the expected payment of an agent with type  $t$ , which we designate by  $m(t)$ . We simplify our task by taking advantage of strict monotonicity of the equilibrium bid function in  $t$ .

$$\begin{aligned}
 m(t) &= \int_0^t (0.95s(t) + 0.84s(T) + 0.78)dT + \\
 &\quad + \int_t^1 (0.73s(t) + 0.53)dT = \\
 &= 0.95t(0.72t - 0.73) + 0.84(0.36t^2 - 0.73t) + \\
 &\quad + 0.78t + 0.73(0.72t - 0.73)(1 - t) + 0.53(1 - t) = \\
 &= 0.4604t^2 + 0.0018t - 0.0029.
 \end{aligned}$$

By symmetry, the expected revenue is twice the expectation of  $m(t)$ :

$$R = 2 \int_0^1 m(t)dt = 2 \int_0^1 (0.4604t^2 + 0.0018t - 0.0029)dt > 0.3.$$

To confirm individual rationality, we need to compute the expected value to an agent with type  $t$  from this auction, which we label  $v(t)$ :

$$v(t) = \int_0^t 0.96tdT + \int_t^1 0.04tdT = 0.92t^2 + 0.04t.$$

The expected utility to an agent with type  $t$  is its expected value less expected payment:

$$EU(t) = v(t) - m(t) = 0.4596t^2 + 0.0382t + 0.0029.$$

Clearly, this is always positive. Furthermore, the designer can charge each agent an additional participation fee of 0.0029 and maintain individual rationality. Since this uniform fee will not affect agents' incentives, the designer will gain an additional 0.0058 in revenue without compromising the individual rationality constraint.

### A.14 Proof of Theorem 14

The intuition for the proof is straightforward. Suppose that the equilibrium bid function is strictly increasing and  $q = 1$ . Then, since the high bidder always gets the good, and the higher type is always the high bidder, the good always goes to the agent that values it more. Consequently, this design yields optimal welfare. The reverse argument works in the other case.

Formally, expected welfare is

$$pE_{t,T}[t \mid t > T] + (1 - p)E_{t,T}[t \mid t < T] + 0.5E_{t,T}[t \mid t = T],$$

where  $p$  is the probability that the high type gets the good. Since the probability that types of both agents are equal is 0, the third term is 0. Furthermore,  $E_{t,T}[t \mid t > T] = 2/3$ , since this is just the first order statistic of the type distribution, and  $E_{t,T}[t \mid t < T] = 1/3$  since it is the second order statistic of the type distribution. Consequently, expected welfare is  $(2/3)p + (1/3)(1 - p)$ . This is maximized when  $p = 1$ , and the maximal value is  $2/3$ . Now, if bid function is increasing in  $t$ , then  $q = p = 1$  ensures optimality. If bid function is decreasing in  $t$ , on the other hand,  $q = (1 - p) = 0$  ensures optimality.

### A.15 Proof of Proposition 15

We will work with the symmetric equilibrium bid of  $s(t) = 0.645t - 0.44$ . Since we have already shown the optimality of this mechanism, we just need to confirm individual rationality and compute the revenue from this auction.

As before, we start with computing the payment of an agent with type  $t$ :

$$\begin{aligned}
m(t) &= \int_0^t (0.88s(t) + 0.23s(T) + 0.28)dT + \\
&\quad + \int_t^1 (0.06s(t) + 0.32s(T))dT = \\
&= 0.88t(0.645t - 0.44) + 0.23(0.3225t^2 - 0.44t) + \\
&\quad + 0.28t + 0.06(0.645t - 0.44)(1 - t) + \\
&\quad + 0.32(-0.3225t^2 + 0.44t - 0.1175) = \\
&= 0.499875t^2 - 0.0025t - 0.064.
\end{aligned}$$

By symmetry, the expected revenue is twice the expectation of  $m(t)$ :

$$R = 2 \int_0^1 (0.499875t^2 - 0.0025t - 0.064)dt = 0.20275.$$

The expected value of an agent,  $v(t)$  is just  $t^2$ , since the high type always gets the good. Consequently, expected utility to an agent is

$$EU(t) = v(t) - m(t) = 0.50012^2 + 0.0025t + 0.064.$$

Since this is always nonnegative when  $t \in [0, 1]$ , ex post individual rationality constraint holds. Note also that it will hold weakly if we charge each participant 0.064 for entering the auction. Thus, the designer could gain an additional 0.128 in revenue without affecting incentives, welfare optimality, and individual rationality.

### A.16 Proof of Theorem 16

Since we are assuming symmetry and the equilibrium bid function is increasing in  $t$ , the objective is equivalent to

$$\begin{aligned} \inf_{t>T} [k_1 s(t) + k_2 s(T) + k_3 s(T) + k_4 s(t)] &= \\ \inf_{t>T} [k_1 m t + k_2 m T + k_3 m T + k_4 m t] &= \\ m \inf_{t>T} [(k_1 + k_4)t + (k_2 + k_3)T] &= 0. \end{aligned}$$

### A.17 Proof of Proposition 17

We will use the symmetric equilibrium bid of (approximately)  $s(t) = 0.43t - 0.51$ .

First we establish the robust revenue properties of the design. By symmetry, the robust objective is equivalent to

$$\inf_{t>T} (s(t) + 0.34s(T) + 0.69) = \inf_{t>T} (0.43t + 0.1462T + 0.0066) = 0.0066.$$

The expected utility of type  $t$  is

$$\int_0^t (t - s(t) - 0.34s(T) - 0.69)dT = 0.4969t^2 - 0.0066t,$$

which attains a minimum at  $t = 0.0066412$ , with the minimum value of just above  $-0.000022$ .

### A.18 Proof of Proposition 18

We will use the symmetric equilibrium bid of  $s(t) = (7/9)t + 2/9$ . The expected payment of type  $t$  is

$$m(t) = \int_0^t \left(\frac{7}{9}T + \frac{2}{9}\right)dT = \frac{7}{18}t^2 + \frac{2}{9}t.$$

The expected revenue is then

$$R = 2 \int_0^1 \left(\frac{7}{18}t^2 + \frac{2}{9}t\right)dt = \frac{13}{27}$$

which is approximately 0.48.

Since the high bidder always gets the good,  $v(t) = t^2$ . The expected utility of an agent with type  $t$  is then

$$eu = \frac{11}{18}t^2 - \frac{2}{9}t,$$

which attains its minimum when  $t = 2/11$ , with the minimum value of  $-44/2178$  (just under  $-0.02$ ). Thus, it is not individually rational. To fix the mechanism, the designer could afford each agent 0.021 for participation, reducing his revenue to 0.438.

### A.19 Proof of Proposition 19

We will use the symmetric equilibrium bid of  $s(t) = 1.613t - 0.234$ . First, we compute expected payment of type  $t$ :

$$\begin{aligned} m(t) &= \int_0^t (0.98s(T) + 0.09)dT + \int_t^1 0.33s(t)dT \\ &= 0.98(0.8065t^2 - 0.234t) + 0.09t + \\ &\quad + 0.33(1.613t - 0.234)(1 - t) \\ &= 0.25808t^2 + 0.47019t - 0.07722. \end{aligned}$$

The expected revenue is then

$$R = 2 \int_0^1 (0.25808t^2 + 0.47019t - 0.07722)dt = 0.4878.$$

Since the high bidder always gets the good,  $v(t) = t^2$ , and the expected utility of type  $t$  is then

$$EU(t) = 0.74192t^2 - 0.47019t + 0.07722.$$

The function  $EU(t)$  is always positive, and the minimum gain for any agent type is 0.00273. Thus, the designer could charge an entry fee of 0.0027 and gain an additional 0.0054 in revenue, for a total of 0.4932.

### A.20 Proof of Proposition 20

In this case, we will use the symmetric equilibrium bid of  $s(t) = 0.595t - 0.2$ . The expected payment of type  $t$  is

$$\begin{aligned} m(t) &= \int_0^t (s(t) + 0.33s(T) + 0.22)dT + \\ &\quad + \int_t^1 (0.22s(t) + 0.12s(T))dT \\ &= 0.595t^2 - 0.2t + 0.33(0.2975t^2 - 0.2t) + 0.22t + \\ &\quad + 0.22(0.595t - 0.2)(1 - t) + \\ &\quad + 0.12(-0.2975t^2 + 0.2t + 0.0975) \\ &= 0.526575t^2 + 0.1529t - 0.0323. \end{aligned}$$

The expected revenue is then

$$R = 2 \int_0^1 (0.526575t^2 + 0.1529t - 0.0323) \approx 0.44.$$

Since  $q = 1$ ,  $v(t) = t^2$ , and, therefore

$$EU(t) = 0.473425t^2 - 0.1529t + 0.0323,$$

which we can verify is always positive. Thus, this design is ex post individually rational. Since its minimum value is slightly above 0.0199, we can bill this amount to each agent for participating in the auction without affecting incentives or ex post individual rationality. This adjustment will give the designer 0.0398 of additional revenue, for a total of about 0.4798.

## A.21 Proof of Proposition 21

We use the symmetric equilibrium bid function  $s(t) = -0.22t - 0.175$  here.

Since the bids are strictly decreasing in types, the expected value of type  $t$  is

$$v(t) = \int_0^t 0.63t \, dT + \int_t^1 0.37t \, dT = 0.26t^2 + 0.37t.$$

By symmetry, the expected welfare is then

$$W = 2 \int_0^1 v(t) \, dt = 0.543.$$

The expected payment of type  $t$  is

$$\begin{aligned} m(t) &= \int_0^t (0.29s(t) + 0.67s(T) + 0.48) \, dT + \\ &\quad + \int_t^1 (0.8s(t) + s(T) + 0.49) \, dT \\ &= -0.29(0.22t + 0.175)t - 0.67(0.11t^2 + 0.175t) + \\ &\quad + 0.48t + 0.8(-0.22t - 0.175)(1 - t) - 0.11t^2 + \\ &\quad + 0.175t - 0.285 + 0.49(1 - t) \\ &= 0.1485t^2 - 0.004t + 0.065. \end{aligned}$$

Thus, we can compute the expected revenue:

$$R = \int_0^1 (0.1485t^2 - 0.004t + 0.065) \, dt = 0.225.$$

The expected utility of type  $t$  is

$$EU(t) = v(t) - m(t) = 0.1115t^2 + 0.374t - 0.065,$$

which attains its minimum at the lower type boundary of 0, with the minimum value of -0.065, and is negative over the range of types  $[0, 0.17]$ . Thus, the designer could make the mechanism completely ex interim individually rational at a loss of an additional 0.013 in revenue by offering each agent a participation gift of 0.065. With this gift, the revenue would fall to 0.095.

## A.22 Proof of Proposition 22

We use the symmetric equilibrium bid function  $s(t) = 0.935t - 0.18$  here.

The expected payment of an agent with type  $t$  is

$$\begin{aligned}
 m(t) &= \int_0^t (0.51s(t) + s(T) + 0.09)dT + \\
 &+ \int_t^1 (0.34s(t) + 0.26s(T))dT \\
 &= 0.51(0.935t^2 - 0.18t) + 0.4675t^2 - 0.18t + 0.09t + \\
 &+ 0.34(0.935t - 0.18)(1 - t) + \\
 &+ 0.26(-0.4675t^2 + 0.18t + 0.2875) \\
 &= 0.5049t^2 + 0.2441t + 0.01355.
 \end{aligned}$$

The expected revenue is thus

$$R = 2 \int_0^1 (0.5049t^2 + 0.2441t + 0.01355)dt = 0.6078.$$

The expected utility of an agent with type  $t$  is

$$EU(t) = v(t) - m(t) = 0.4951t^2 - 0.2441t - 0.01355,$$

which is negative for a fairly broad range of types (although always above the tolerance level that we set). Type  $t^* = 0.24652$  fairs the worst, incurring a loss of nearly 0.044. However, by compensating both agents this amount, we ensure ex interim individual rationality without affecting incentives. As a result, the designer will lose 0.088 in expected revenue, which will fall to 0.5198.

## A.23 Proof of Proposition 23

By symmetry, the objective value is equivalent to

$$\inf_{t>T} s(T) = \inf_{t>T} \left(\frac{7}{9}T + \frac{2}{9}\right) = 2/9.$$

The rest follows by Proposition 18.

## A.24 Proof of Proposition 24

The objective value is equivalent to

$$\begin{aligned}
 \inf_{t>T} (s(t) + 0.71s(T) + 0.14 + 0.09s(t)) &= \\
 = \inf_{t>T} (0.3t - 0.045 + 0.71(0.3T - 0.045) + 0.14 + 0.09(0.3t - 0.045)) &= 0.059.
 \end{aligned}$$

The expected utility of an agent is

$$\begin{aligned}
eu(t) &= \int_0^t (0.86t - 0.3t + 0.045 - 0.71(0.3T - 0.045) - 0.14)dT + \\
&\quad + \int_t^1 (0.14t - 0.09(0.3T - 0.045))dT = \\
&= 0.56t^2 + 0.07695t - 0.1065t^2 - 0.14t + 0.14t - 0.14t^2 + 0.00405 - \\
&\quad - 0.00405t - 0.0135(1 - t^2) = \\
&= 0.327t^2 + 0.0729t - 0.0135.
\end{aligned}$$

which attains a minimum value of -0.0135. Thus, the participation award of 0.0135 to each agent is necessary to make this design individually rational, with the resulting robust revenue of 0.032.