Dynamic Incentive Mechanisms

Dynamic Incentive Mechanisms

David C. Parkes
SEAS
Harvard University
Cambridge, MA 02138
parkes@eecs.harvard.edu

Ruggiero Cavallo
CIS
University of Pennsylvania
Philadelphia, PA 19104
cavallo@post.harvard.edu

Florin Constantin
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
florin@cc.gatech.edu

Satinder Singh
CSE
University of Michigan
Ann Arbor, MI 48109
baveja@umich.edu

Abstract

Much of AI is concerned with the design of intelligent agents. A complementary challenge is to understand how to design "rules of encounter" (Rosenschein and Zlotkin 1994) by which to promote simple, robust and beneficial interactions between multiple intelligent agents. This is a natural development, as AI is increasingly used for automated decision making in real-world settings. As we extend the ideas of mechanism design from economic theory, the mechanisms (or rules) become algorithmic and many new challenges surface. Starting with a short background on mechanism design theory, the aim of this paper is to provide a non-technical exposition of recent results on *dynamic incentive mechanisms*, which provide rules for the coordination of agents in sequential decision problems. The framework of dynamic mechanism design embraces coordinated decision making both in the context of uncertainty about the world external to an agent and also in regard to the dynamics of agent preferences. In addition to tracing some recent developments, we point to ongoing research challenges.

Introduction

How can we design intelligent protocols to coordinate actions, allocate resources and make decisions in environments with multiple rational agents each seeking to maximize individual utility? This problem of "inverse game theory," in which we design a game form to provide structure to interactions between rational agents, has sparked interest within artificial intelligence since the early 1990s, when the seminal work of Rosenschein and Zlotkin (1994) and Ephrati and Rosenschein (1991) introduced the core themes of mechanism design to the AI community.

Consider this an "inverse artificial intelligence" perhaps, the problem of designing rules to govern the interaction between agents in order to promote desirable outcomes for multi-agent environments. The rules are themselves typically instantiated algorithmically, with the computational procedure by which the rules of interaction are enacted forming the coordination *mechanism*, and the object of design and analysis.

The design of a suitable mechanism can require the integration of multiple methods from AI, such as those of preference elicitation, optimization, and machine learning, in addition to an explicit consideration of agent incentives. An interesting theme that emerges is that by careful design it is possible to *simplify* the reasoning problem faced by agents. Rather than require agents to struggle with complex decision problems, we are in the unusual position of being able to design simple decision environments.

Mechanism design theory was developed within mathematical economics as a way to think about what "the market"—viewed somewhat like an abstract computer—can achieve, at least in principle, as a method for coordinating the activities of rational agents. During the debates of the 1960's and 1970's about centralized command-and-control versus market economies, Hurwicz (1973) developed the formal framework of mechanism design to address this fundamental question. The basic set-up considers a system of rational agents and an outcome space, with each agent holding private information about its *type*, this type defining the agent's preferences over different outcomes. Each agent makes a claim about its type, and the mechanism receives these claims and selects an outcome; e.g., an allocation of resources, an assignment of tasks, or decision about a public project. Being rational agents, the basic modeling assumption is that an agent will seek to make a claim so that the outcome selected maximizes its utility given its beliefs about the claims made by other agents, i.e., in a game-theoretic equilibrium. Jackson (2000) provides an accessible recent survey of economic mechanism design.

Mechanism design theory typically insists on designs that enjoy the special property of *incentive compatibility*: namely, it should be in every agent's own best interest to be truthful in reporting its type. Especially when achieved in a dominant-strategy equilibrium (so that truthfulness is best whatever the claims of other agents) this obviates the need for strategic reasoning and simplifies an agent's decision problem. If being able to achieve incentive compatibility sounds a bit too good to be true, it often is; in addition to positive results there are also impossibility results that identify properties on outcomes that cannot be achieved under any incentive mechanism. Just as computer science uses complexity considerations to divide problems into tractable and intractable, mechanism design theory classifies outcome rules into those that are possible and impossible, working under its own lens of *incentive* constraints. Where mechanism design gets really interesting within AI is when the incentive constraints are in tension with the computational constraints, but we're getting a bit ahead of ourselves.

Two Simple Examples

For a first example, we can think about an auction for a last-minute ticket for a seat at an ice hockey game at the Winter Olympics. Rather than the usual auction concept (the highest bidder wins and pays her bid price), we can consider instead a *second-price sealed-bid* auction (Vickrey 1961). See Figure 1. There are three agents (A1, A2 and A3), each with a simple type that is just a single number representing its value for the ticket. For example, A2 is willing to pay up to \$1000, with its utility interpreted as $v - p$ for value $v = 1000$ and payment p. In a second-price auction, A2 would win, but pay \$900 rather than its bid price of \$1000.

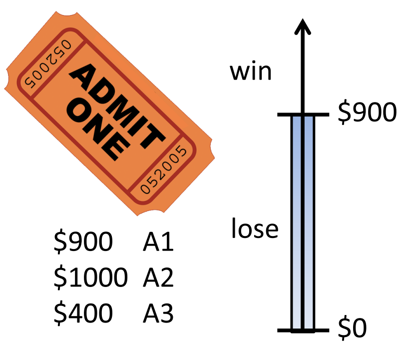


Figure 1: A single-item allocation problem. Agents A1, A2 and A3 have value \$900, \$1000 and \$400. A2 wins the hockey ticket and makes a payment of \$900 for any bid above \$900.

Collecting reports of a single number from each agent, allocating the item to the agent with the highest report (breaking ties at random) and collecting as payment the second-highest report defines the allocation mechanism. This mechanism is incentive compatible in the strong sense that truthful bidding is a dominant strategy equilibrium. A2 need not worry about bidding less than \$1000, its bid merely describes the most it might pay and its actually payment is the smallest amount that it could bid and still win.[1]

For a second example, consider a university campus with a central mall along which all students walk. The problem is to determine where to build a glass building that will house the skeleton of a 26-meter blue whale that washed up on Prince Edward Island in 1987.[2] Let us suppose that every student has a preferred location ℓ^* along the mall to locate the whale. Moreover, we assume *single peaked* preferences, such that for two locations ℓ and ℓ', both on the same side of a student's most preferred location, then ℓ' is less preferred than ℓ whenever ℓ' is further from ℓ^* than ℓ.

A truthful mechanism can locate the whale at the *median* of all the reported peaks (with a random tie breaking step if there is an even number of agents) (Moulin 1980). See Figure 2. For truthful reports of preference peaks, i.e., $(100, 200, 400)$, A2's report forms the median and the whale is located at 200-meters into the mall. No agent can improve the outcome in its favor. For example, A1 only changes the location when its report is greater than 200, but then the position moves away from its most preferred location of 100. It is a fun exercise to think about how the *mean* rule, in comparison, fails to be truthful.[3]

Mechanism Design and AI

These two examples are representative of problems for which the microeconomic theory of mechanism design has made great advances. In deriving clean, closed-form results, it is common to adopt a model in which the private information of an agent is a single number, perhaps a value, or a location. A typical goal is to characterize the family of incentive compatible mechanisms for a problem, in order to identify the mechanism that is optimized for a particular design criterion, perhaps social welfare, revenue, or some notion of fairness. Moreover, the examples are representative in that they are static problems: the set of agents is fixed and a decision is to be made in a single time period.

Contrast these examples, then, with the kinds of complex decision problems in which AI is typically interested: environments in which actions have uncertain effects, are taken across multiple time periods, and per-

[1] Another way to see this is that each agent faces the choice of winning or losing, and the price for winning is independent of his report. The report of an agent's value just determines which choice is made, with the choice that is made being optimal for the agent given the report. A2 faces a choice of losing for zero payment, or winning for \$900, and is happy to win. A3 faces a choice between winning for \$1000 or losing and is happy not to win.

[2] The University of British Columbia has just opened such an exhibit.

[3] Both of these examples illustrate *direct* mechanisms in which agents make a report about their type. In indirect mechanisms, a message sent by an agent is typically interpreted instead as providing partial information about an agent's type; e.g., a bid for an item at price p in an ascending price auction can be interpreted as a claim that $v \geq p$, for value v. Indirect mechanisms are important when preference elicitation is costly.

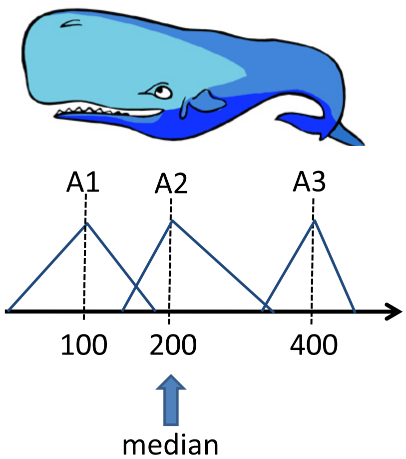


Figure 2: Single-peaked preferences and the median selection mechanism. Agents A1, A2 and A3 each have preferred locations at 100, 200 and 400. The whale exhibit is located at the median report of 200.

haps with the need to learn about the value and effect of different actions. But we are increasingly interested in multi-agent environments, and including those in which there is self-interest: e-commerce, crowdsourcing, multi-robot systems, sensor networks, smart grids, scheduling, and so forth.

The research challenge that we face is to adapt the methods of mechanism design to more complex problems, addressing computational challenges as they arise and extending characterization results as required, in order to develop a theory and practice of computational mechanism design for dynamic, complex, multi-agent environments. It seems to us that progress will require relaxing some aspects of mechanism design theory, for example substituting a kind of approximate incentive compatibility (or models of non-equilibrium behavior) and being willing to adopt heuristic notions of optimality in return for increasing mechanism scale and mechanism reach.

Outline

In what follows we will develop some intuitions for incentive mechanisms that are compatible with two different kinds of dynamics. There is *external* uncertainty, in which the dynamics are outside the boundary of individual agents and occur because of agent arrivals and departures and other uncertain events in the environment. In addition there is *internal* uncertainty, in which the dynamics are due to learning and information acquisition within the boundary of an individual agent, and where it is the private information of an agent that is inherently dynamic.

External Uncertainty

By external uncertainty, we intend to describe a sequential decision problem in which the uncertain events occur in the actual world rather than merely in an agent's view of the world. The dynamics may include the arrival and departure of agents with respect to the sphere of influence of a mechanism's outcome space, as well as changes to the outcomes that are available to a mechanism. Crucially, by insisting on external uncertainty we require that an agent that arrives has a fixed type, and is able to report this type, truthfully if it chooses, to the mechanism upon its arrival.[4]

A Dynamic Unit-Demand Auction

To illustrate this, we can return to the problem of allocating hockey tickets, and now ask what would happen if one ticket is sold on each of two days with agents arriving and departing in different periods.

There is one ticket available for sale on Monday and one available for sale on Tuesday. Both tickets are for the Wednesday game. The type of an agent is now associated with an arrival, departure and value. The arrival period is the first period in which it is able to report its type. The departure period is the final period in which an agent has value for winning a ticket. The arrival-departure interval denotes the set of periods in which an agent has value for an allocation. See Figure 3. In the example, the types are $((\$900, 1, 2), (\$1000, 1, 2), (\$400, 2, 2))$ so that A1 and A2 are "patient" with value for a ticket on either of days 1 or 2 but with A3 "impatient" and arriving on day 2 and requiring a ticket allocation on the day of its arrival. The setting is *unit-demand* because each agent requires only a single ticket.

We could run a sequence of second-price auctions, with A2 winning on Monday for \$900, dropping out, and A1 winning on Tuesday for \$400. But this would not be truthful, and we should expect that agents would try to misreport to improve the outcome in their favor. A2 can deviate and bid \$500 and win on Tuesday for \$400, or just wait until Tuesday to bid.

A simple variant provides a dynamic and truthful auction. We can adopt the same greedy decision policy, committing the item for sale in a given period to the unallocated agent with the highest value amongst those present. The item is held by the mechanism until the reported departure time of the agent. But rather than collect the second-price, the payment is set to the lowest amount the agent could have bid and still been allocated *in some period* (not necessarily the same period) within its arrival-departure interval (Hajiaghayi et al. 2005). In the example, A2 is allocated a ticket on

[4]Without a dynamic agent population, external uncertainty can be rolled into the standard framework of mechanism design because incentive constraints need bind only in the initial period, with a decision made and committed to (e.g., a decision policy) that will structure the subsequent realization of outcomes (Dolgov and Durfee 2005).

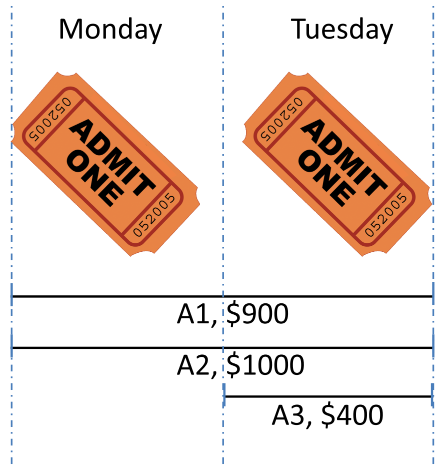


Figure 3: A dynamic auction for tickets to a Wednesday hockey game. Agents A1 and A2 have value \$900 and \$1000, respectively, for an allocation on Monday or Tuesday. A3 arrives on Tuesday, and has value \$400 for a ticket allocated that day.

Monday which is released to the agent on Tuesday. A1 is allocated on Tuesday. For payments, both agents pay \$400 because this is the smallest amount each could bid (leaving all other bids unchanged) and still be allocated a ticket.

[Could be formatted into a sidebar on "monotonicity".]

The key to understanding the truthfulness of the dynamic auction is that the allocation policy is *monotonic.* An agent that loses with a bid v, arrival a and departure d also loses for all bids with $v' \leq v, a' \geq a$ and $d' \leq d$. It is easy to see that the greedy allocation rule is monotonic, whatever the bids received from other agents. To see why monotonicity provides truthfulness, first consider any (a', d') report and some $v' \neq v$. The payment p for any winning bid is the "critical value" at which an agent first wins in some period. This is the same for all winning bids, and therefore if $v \geq p$ the agent wins and has no useful value misreport. If $v < p$ then the agent loses, and continues to lose for all v' except $v' \geq p$, but then its payment p is greater than its true value.

Knowing that value v will be truthfully reported, for all arrival-departure misreports, we can now consider temporal manipulations. For this, we assume that misreports $a' \leq a$ are not possible because the arrival models the time period when an agent first realizes his or her demand, or first discovers the mechanism. A misreport to a later departure with $d' > d$ is never useful because the mechanism does not release an allocated ticket to an agent until d' when the agent would have no value. Finally, deviations with $a' \geq a$ and $d' \leq d$ can only increase the payment made by a winning agent by monotonicity (since the critical value must increase) and will have no effect on the outcome for a losing agent.

[End of sidebar.]

But what about more general problems? Can we design incentive mechanisms for uncertain environments with a dynamic agent population, and where agents have general valuation functions on sequences of actions by the mechanism? In fact, this is possible, through a different generalization of the second-price auction to dynamic environments.

A Static Variation

We first review the *Vickrey-Clarke-Groves* (VCG) mechanism for static problems. For this, assume that the bids $((\$900, 1, 2), (\$1000, 1, 2), (\$400, 2, 2))$ can all be made on Monday (even for agent A3). The complete allocation and payments can now be determined in a single period. In the VCG mechanism, the tickets are allocated to maximize reported value, i.e., to A1 and A2. For the payment, we determine the externality imposed by an agent through its presence. This is $V_a - V_b$, where

V_a*: the total reported value to other agents from the optimal action that would be taken if the agent was absent, and*

V_b*: the total reported value to other agents from the action made when the agent is present.*

In the example, for A1 this is payment $p_1 = V_a - V_b = 1400 - 1000 = 400$ and for A2 this is $p_2 = V_a - V_b = 1300 - 900 = 400$. In both cases, this is the cost imposed on A3 by the presence of A1 or A2, which is \$400 because A3 would be allocated if either A1 or A2 was absent. For an incentive analysis, consider agent A1. The utility to A1 is $v \cdot \mathbb{I}(\text{A1 wins}) - p_1 = v \cdot \mathbb{I}(\text{A1 wins}) - (V_a - V_b) = v \cdot \mathbb{I}(\text{A1 wins}) + V_b - V_a$, where $\mathbb{I}(E)$ is 1 if event E is true and 0 otherwise. Now, V_a is independent of A1's bid and so can be ignored for an incentive analysis. We are left with A1's utility depending on

$$v \cdot \mathbb{I}(\text{A1 wins}) + V_b,$$

where v is its true value. A1 affects this quantity through the effect of its bid on the allocation, and thus whether or not it wins and also the allocation to other agents and thus V_b. But we see that to maximize its true value plus the total reported value of the other agents the agent can just be truthful. The VCG mechanism selects an allocation that maximizes total reported value and thus maximizes A1's true value plus the reported value of the other agents when A1 is truthful.[5]

A misreport is never useful for an agent and can provide less utility.

The Online VCG Mechanism

A generalization of the VCG mechanism to dynamic environments is provided by the *online VCG mechanism* (Parkes and Singh 2003). Payments are collected

[5] In the example, when A1 is truthful then $v + V_b = 900 + 1000 = 1900$ and this is unchanged for all reports $v' \geq 400$. For reports $v' < 400$, then $v + V_b = 0 + 1400 < 1900$ and A1's utility falls from 500 to 0.

so that each agent's expected payment is exactly the *expected externality* imposed by the agent on other agents upon its arrival. The expected externality is the difference between the total expected (discounted) value to the other agents under the optimal policy without agent i, and the total expected (discounted) value to the other agents under the optimal policy with agent i.[6] For this, the mechanism must have a correct probabilistic model of the environment, including a model of the agent arrival and departure process.

The online VCG mechanism aligns the incentives of agents with the social objective of following a decision policy that maximizes the expected total (discounted) value to all participants. The proof of incentive compatibility establishes that each agent's utility is aligned with the total expected (discounted) value of the entire system. The mechanism's incentive properties extend to any problem in which each agent's value, conditioned on a sequence of actions by the mechanism, is *independent* of the private type of other agents.

For a simple example, we can suppose that the types of A1 and A2 are unchanged while a probabilistic model states that A3 will arrive on Tuesday with a value that is uniform on $[\$300, \$600]$. Thus, the expected externality that A2 imposes on A3 is $450, which is the mean of this value distribution. In making this payment, A2 cannot do better *in expectation* by misreporting its type, as long as other agents in future periods play the truthful equilibrium and the probabilistic model of the center is correct.

The kind of incentive compatibility achieved by the online VCG mechanism is somewhat weaker than the dominant strategy equilibrium achieved in the static VCG mechanism. Notice, for example, that if A3 always bids $300 then A2 can reduce its payment by delaying its arrival until period 2. Rather, truthful reporting is an agent's best-response in expectation, just as long as the probabilistic model of the mechanism is correct and agents in the current and future periods are truthful.[7]

A Challenge Problem: Dynamic Combinatorial Auctions

In a combinatorial auction (CA), agent valuations can express arbitrary relationships between items, such as substitutes ("I want a ticket for one of the following hockey games") and complements ("I want a ticket for two games involving USA.") See Cramton et al. (2006) for a summary of recent advances for *static* CAs. In a dynamic CA, we can allow for an uncertain supply of distinct goods, agent arrival and departure, and agents with preferences on different sequences of allocations of goods.

The online VCG mechanism is well defined, and retains incentive compatibility for dynamic CAs. On the other hand, there remain significant computational challenges in getting these dynamic mechanisms to truly play out in realistic, multi-agent environments, and here there is plenty of opportunity for innovative computational advances to be made within AI. For example, the following issues loom large:

(a) *Bidding languages and preference elicitation.* This is an issue well-developed for static CAs but largely unexplored for dynamic CAs. One direction is to develop concise representations of valuation functions on sequences of allocation decisions. Another direction is to develop methods for preference elicitation, so that only as much information as is required to determine an optimal decision in the current period is elicited.

(b) *Winner determination and payment computation.* The winner-determination and payment computation problem has been extensively studied in static CAs, with tractable special cases identified, and approximation schemes and scalable, heuristic algorithms developed (Cramton, Shoham, and Steinberg 2006). Similarly, we require significant progress on winner determination in dynamic CAs, where the decision problem is online and one of stochastic optimization. Given approximations, then incentives will once again come into play.

(c) *Learning.* The incentive compatibility of the dynamic VCG mechanism relies, in part, on having an exact probabilistic model of the agent arrival and departure process. We must develop techniques to provide incentive compatibility (perhaps approximately) along the path of learning, in order to enable the deployment of dynamic mechanisms in unknown environments.

We are, in a sense, in the place that static CAs were around a decade ago, when there was a concerted effort to provide a solid computational footing for CAs, inspired in part by the push by the U.S. Federal Communications Commission to design CAs for the allocation of wireless spectrum. One domain that seems compelling, in working to develop a computational grounding for dynamic CAs is *crowdsourcing*, where human and computational resources are coordinated to solve challenging problems (Shahaf and Horvitz 2010; von Ahn and Dabbish 2008). A second domain of interest is *smart grids* in which renewable energy sources (necessarily more bursty than traditional power sources) are

[6] As is familiar from Markov Decision Processes, for infinite decision horizons a discount factor $\gamma \in (0, 1)$ is adopted, and the objective is to maximize the expected discounted sum, i.e., $V(0) + \gamma V(1) + \gamma^2 V(2) + \ldots$, where $V(t)$ is the total value to all agents for the action in period t.

[7] This is a refinement on a Bayesian-Nash equilibrium, referred to as a *within period ex post Nash* equilibrium because an agent's best strategy is to report its true type whatever the reports of other agents up to and including the current period, just as long as other agents follow the truthful equilibrium in future periods. It is equivalent to dominant-strategy equilibrium in the final period of a dynamic problem, when online VCG is equivalent to the static VCG mechanism.

dynamically matched through dynamic pricing against the shifting demand of users (Vytelingum et al. 2010).

Heuristic Mechanism Design

Within microeconomic theory, a high premium is placed on developing analytic results that exactly characterize the optimal mechanism rule within the space of all possible mechanism rules. But this is typically not possible in many of the complex environments in which AI researchers are interested because it would imply that we can derive an optimal *algorithm* in a domain. This is often out of reach. There is even debate about what optimality entails in the context of bounded computational resources, and still greater gaps in understanding about how to best use bounded resources (Russell and Wefald 1991; Russell, Subramanian, and Parr 1993). Rather, it is more typical to approach difficult computational problems in AI through a combination of inspiration and perspiration, with creativity and experimentation and the weaving together of different methods.

From this viewpoint, we can ask what it would mean to have a *heuristic* approach to mechanism design? One idea is to insist on provable incentive properties but punt on provably optimality properties. In place of optimality, we might adopt as a gold standard by which to evaluate a mechanism the performance of a state-of-the-art, but likely heuristic, algorithm for the relaxed version of the problem in which agents are cooperative rather than self-interested (Parkes 2009). In this sense, a heuristic approach to mechanism design is successful when the empirical performance of a designed mechanism is good in comparison with the performance of a gold standard algorithm that would be adopted in a cooperative system.

To make this concrete, we can return again to our running example and suppose now that our hockey enthusiasts have friends that also like hockey, and want to purchase multiple tickets. An agent's type is now a value for $q \geq 1$ tickets, and this is an "all-or-nothing" demand, with no value for receiving $q' < q$ tickets and the same value for more than q tickets. Optimal sequential policies are not available for this problem with current computational methods. The reason is a curse of dimensionality resulting from the need to include active agents in the state of the planning space.[8] A gold standard, but heuristic algorithm is provided by the methods of *online stochastic combinatorial optimization* (OSCO) (Hentenryck and Bent 2006). Crucially, we can model the future realization of demand as independent of past allocation decisions, conditioned on the past realization of demand. This *uncertainty independence* property permits high quality decision making through scalable, sample trajectory methods.

[8]In comparison, if agents are impatient and demand an allocation of q tickets in the period in which they arrive, the associated planning problem is tractable, with a closed-form characterization of the optimal policy and an analytic characterization of incentive compatible dynamic mechanisms (Dizdar, Gershkov, and Moldovanu 2009).

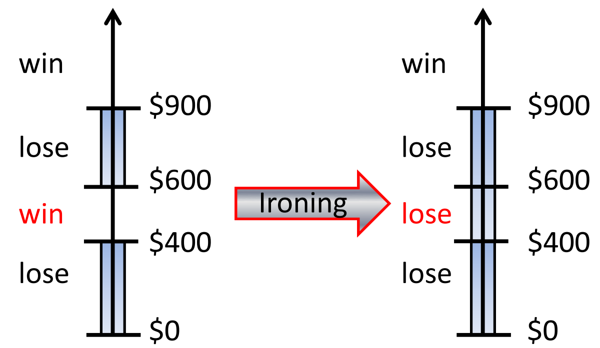


Figure 4: An illustration of output-ironing in a single-item allocation problem. The initial allocation rule is non-monotonic, with the agent winning for values [\$400, \$600] but losing for values [\$600, \$900]. Output-ironing would establish monotonicity by canceling the allocation decision in interval [\$400, \$600].

But, these OSCO algorithms cannot be directly combined with payments to achieve incentive compatibility. A dynamic-VCG approach fails because an agent may seek to misreport to "correct" for the approximation of the algorithm.[9] Moreover, the heuristic policies can fail to be monotonic, which would otherwise allow for suitable payments to provide incentive compatibility (Parkes and Duong 2007). For a simple example, suppose that three tickets for the Wednesday game are for sale, and that each can be sold on either Monday or Tuesday. There are three bidders, with types $(\$50, 1, 1, q = 2)$, $(\$90, 1, 2, q = 1)$ and $(\$200, 2, 2, q = 1)$ for A1, A2 and A3, where A3 arrives with probability $\epsilon \in (0, 50/200)$. The optimal policy (which we should expect from OSCO in this simple example) will allocate $\{A1\}$ on Monday and then $\{A2\}$ or $\{A3\}$ on Tuesday depending on whether or not A3 arrives. But, A2 could request $q = 2$ instead and report type $(\$90, 1, 2, q = 2)$. In this case, the optimal policy will allocate $\{\}$ on Monday, and then $\{A2\}$ or $\{A2, A3\}$ on Tuesday depending on whether or not A3 arrives. This fails an obvious generalization of monotonicity from the dynamic unit-demand auction, which insists on improving outcomes for higher value, earlier arrival, later departure, or smaller quantity. In this case, A2 asks for an additional ticket and in one scenario (when A3 arrives) goes from losing to winning.

However, it can be effective to perturb the decisions of an OSCO algorithm in order to achieve monotonicity. For this, a self-correcting process of *output-ironing*

[9]This effect is familiar from static problems (Lehmann, O'Callaghan, and Shoham 2002) but exacerbated here because we get a new unraveling, with agents no longer having a good basis for believing that other agents will be truthful, and thus no longer believing that the mechanism's model of the probabilistic process is correct, and so forth.

is adopted (Parkes and Duong 2007; Constantin and Parkes 2009).[10] A decision policy is automatically modified to provide monotonicity by identifying and canceling decisions for which every higher type would not provably be allocated by the algorithm (see Figure 4 for a simple one-dimensional example). For example, the allocation decision for A2 ($\$90, 1, 2, q = 2$) in the event A3 arrives would be canceled by output-ironing since it would be observed that a higher type ($\$90, 1, 2, q = 1$) would not be allocated. The computational challenge is to enable tractable sensitivity analysis, so that monotonicity can be verified. The procedure also requires that the original algorithm is almost monotonic, so that the performance of the heuristic mechanism remains close to that of the target algorithm.

Internal Uncertainty

By internal uncertainty, we describe a sequential decision problem in which the uncertain events occur within the scope of an individual agent's view of the world. The dynamics are those of information acquisition, learning, and updates to the local goals or preferences of an agent, all of which trigger changes to an agent's preferences.

To model this we adopt the idea of a *dynamic type*: the information, local to an agent and private, can change from period to period and in a way that depends on the actions of the mechanism. For this reason, we will need incentive compatibility constraints to hold for an agent in *every* period, so that we continually provide incentives to share private type information with the mechanism. In contrast, for external uncertainty in which an agent's type is static, it is sufficient to align incentives only until the period in which an agent makes a claim about its type.[11]

We will see an analog to the online VCG mechanism, in which payments are defined so that an agent's expected total payment forward from every period is the expected externality imposed by the agent on the other agents. Whereas with external uncertainty, this property on payments needs to hold only upon an agent's arrival, for internal uncertainty it must hold in *every* period.

[10]This use of "ironing" is evocative of removing the "non-monotonic rumples" from the decision policy, adopted here in the sense of Myerson's (1981) seminal work in optimal mechanism design. Whereas Myerson achieves monotonicity by ironing out non-monotonicity in the input into an optimization procedure, the approach adopted here is to achieve monotonicity by working on the output of an optimization procedure.

[11]We can also couple internal and external uncertainty, for example in enabling mechanism design for environments in which there is uncertainty about the availability of resources and in which individual agents are refining their own beliefs about their value for different outcomes (Cavallo, Parkes, and Singh 2009).

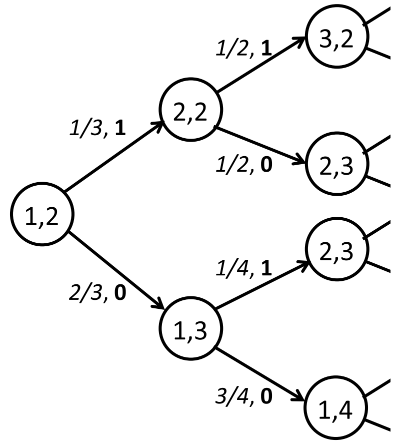


Figure 5: The Markov chain model for a Bayesian learning agent observing a sequence of $\{0, 1\}$ values received for attending a sequence of hockey games. Each state is labeled with counts (N_1^t, N_0^t) of the number of value 1 and value 0 observations current period t. Transitions are labeled with a probability (depending on the relative counts) and the value received.

Dynamic Auctions with Learning Agents

By way of illustration, suppose that our group of hockey enthusiasts are not sure exactly how much they enjoy the sport, and learn new information about their value whenever allocated a ticket. Each time someone receives a ticket to attend a game, they receive an independent sample of their value for watching hockey games. The mechanism design problem is to *coordinate the learning process*, so that a ticket is allocated in each period to maximize the expected discounted value, given that there is uncertainty about each person's true value. In particular, it might be optimal to allocate a ticket to someone other than the person with the highest *current* expected value to allow learning.

Consider Figure 5. This Markov chain illustrates the Bayesian learning process for a single agent. We assume that upon attending a game, an agent's response is extreme and the agent will either like the game (value=1) or dislike the game (value=0). An agent's value is sampled from a stationary distribution, with probability $\theta \in [0, 1]$ for liking a game. In any period t, an agent's current internal belief state is represented by the count (N_1^t, N_0^t) of games liked and disliked. This agent's initial belief state is (1,2) so that its subjective belief is 1/3 that it will like the next game, and 2/3 that it will dislike the next game. The Markov chain captures Bayesian learning. In the event that it likes the game, the observation is a '1' and it transitions to belief state (2,2).[12] From this state, its subjective belief that it will

[12]Formally, the agent's prior in period t on probability θ is a Beta distribution $\theta \sim Beta(\alpha^t, \beta^t)$, with parameters α^t and β^t. The Beta distribution satisfies the conjugate

like the next game is 1/2, and so forth. The dynamic type of an agent is its current belief state. This belief state also defines the probabilistic model for how its type will evolve based on sequences of actions.

For the multi-agent problem, the Markov chain of each agent advances upon an allocation of a ticket. In every period, the decision to make is which Markov chain to advance. This is a *multi-armed bandits problem* (MABP), with each "arm" corresponding to an agent, the state of an arm only changing when activated, and only one arm activated in every period. The optimal policy for the MABP solves the multi-agent learning problem, identifying an allocation policy to maximize expected discounted value and finding the right trade-off between exploration and exploitation. It may not be optimal to allocate an agent with the highest expected value when there is considerable uncertainty about another agent's value. With discounting, and an infinite time horizon, the optimal policy is characterized by an index policy and there is an algorithm that scales linearly in the number of arms (Gittins 1989).[13] What makes this a mechanism design problem is that each agent has private information about its current belief state, and will misreport this information if this improves the allocation policy in its favor.

The Dynamic VCG Mechanism

In the dynamic VCG mechanism (Bergemann and Välimäki 2008), each agent must report a probabilistic model for how its type will evolve based on sequences of actions, and also, in each period, its current type.

The mechanism selects an action in each period according to the optimal policy, i.e., the policy that maximizes the total expected discounted value. Let x^t denote the action selected in period t. As in the online VCG mechanism, the payments are defined so that the expected discounted payments made by an agent equals the expected externality it imposes on the other agents through its effect on the decision policy. But now the expected discounted payment must be aligned in *every* period and not just upon arrival.

For this, a payment equal to $V_a - V_b$ is collected in each period t, where

V_a*: the expected reported discounted value that the other agents would achieve forward from the current period under the optimal decision policy that would be followed if agent i was not present, and*

V_b*: the expected reported discounted value that the other agents would achieve forward from the current period under the optimal decision policy that would be followed if agent i was not present, but conditioned on taking action* x^t *in the current period.*

In so doing, the dynamic VCG mechanism aligns the incentives of an agent with the objective of maximizing the expected discounted value, summed across all agents. This makes it incentive compatible for an agent to report its true probabilistic model and, in each period, its current type.[14]

In the hockey ticket example, each agent will make a claim about its current belief state (N_1^t, N_0^t) in each period. The ticket will be allocated to the agent with the highest index, according to the method of Gittins (1989). The allocated agent will then receive value 1 or 0, make a payment, and advance to a new belief state, whereupon it will make a new report.[15]

For the special case of two agents, the expected externality imposed on A2 when A1 is allocated is $(1-\gamma)W_2$, where $W_2 = \frac{w_2}{1-\gamma}$ is the expected discounted value to agent 2 for receiving the item in every period including the current period, with $w_2 = \frac{N_1^t}{N_1^t + N_0^t}$ and discount factor $\gamma \in (0, 1)$. The amount $(1-\gamma)W_2 = W_2 - \gamma W_2$ represents the effect of pushing back the sequence of allocations to A2 by one period, and is therefore the expected externality imposed on A2 by the presence of A1 in the current period. In fact, we see that the payment collected is $(1-\gamma)W_2 = (1-\gamma)\frac{w_2}{1-\gamma} = w_2$ and exactly the expected value to A2 for the current ticket. For three or more agents, things get just a bit more complicated. For example, with three agents, A1's payment upon winning is not simply $\max(w_2, w_3)$, but greater than this. This is because we will usually have $V_a > \max(W_2, W_3)$, because there is an option value for being able to switch between A2 and A3 over time.[16]

Even with two agents the dynamic VCG mechanism is distinct from a sequence of second-price auctions. This is because the item need not be allocated to the agent with the highest expected value. Rather, A2 may be allocated over A1 if there is still useful information to be gained about A2's actual θ.

To think about a sequence of simple second-price auctions fails, suppose there are two agents, with belief states $(55, 45)$ and $(2, 2)$ respectively. A1 has expected value $55/100 = 0.55$ and A2 has expected value $2/4 = 0.5$. Suppose A1 is truthful in a second-price auction, and bids his current expected value. If A2 also bids truthfully, then with high probability (because $N_1^t + N_0^t$ is large for A1, and thus there is low variance on its

property, so that the posterior is in the same family. Given prior $Beta(\alpha^t, \beta^t)$, then the posterior is $Beta(\alpha^t+1, \beta^t)$ after a '1' and $Beta(\alpha^t, \beta^t + 1)$ after a '0.'

[13] An index policy is one in which an index (or "quality") is computed separately for each arm and the arm with the highest index is activated in every period.

[14] Truthful reporting is an equilibrium in the same refinement of the Bayesian-Nash equilibrium as for the online VCG mechanism, with truthful reporting optimal whatever the current types of agents, as long as the other agents report their true type forward from the current period.

[15] For technical reasons, every agent is entitled to make a report in every period, even if not activated. This is to allow an agent to correct an earlier misreport and be truthful going forward from the current state.

[16] The payment can be easily approximated through a sample-based computation, by sampling the trajectory of states reached under the optimal index policy to A2 and A3 alone.

estimate of θ) A2 will lose in *every* future period. But A2 has considerable uncertainty about its true type θ (which we may suppose is $\theta = 0.8$), and could instead bid more aggressively, for example bidding 0.6 despite having negative expected utility in the current period. This allows for exploration, and in the event of enjoying the hockey game then A2's revised posterior belief is parameterized $(3, 2)$, and the agent will win and have positive expected utility by bidding 0.55 or higher in the next period. In this way, the information value from winning and being allocated a ticket can outweigh the short-term loss in utility.

Dynamic Auctions with Deliberative Agents

For a second example of a problem with internal uncertainty, we go back to the auction for hockey tickets one last time. Suppose now that our sports enthusiasts are now competing for a block of 10 tickets, and that each bidder has uncertainty about how to use the tickets and a costly process to determine the best use and thus its ultimate value. Should the tickets go to a group of friends from college, to a charity auction, or be given as gifts to staff at work? In finding the best use, a bidder needs to take costly actions, such as calling old friends that want to chat forever, or developing and solving an optimization model to find the best use of tickets in rewarding performance and improving morale at work. Given a set of candidate uses, a bidder's value for the tickets is the maximum value across the possible uses.

The problem of deciding when, whether and for how long, to pursue this costly value-improvement process is a problem of *meta-deliberation*. What is the right tradeoff to make between identifying good uses and the cost of this process (Russell and Wefald 1991; Horvitz 1988; Larson and Sandholm 2004)? Here we also need to handle self-interest: one agent might seek to shirk deliberation by pretending to have a very costly process so that deliberation by other agents is prioritized first. The agent would then only need to deliberate about its own value if the other agents find that they have a low enough value. A socially optimal sequence of deliberation actions will try to identify high value uses from agents for which deliberation is quite cheap, in order to avoid costly deliberation by agents whose values are expected to be quite low.

Figure 6 provides a Markov Decision Process (MDP) to model the deliberation process of an individual agent. Later, we will show how to combine such a model into a multi-agent sequential decision problem. From any state, the agent has the choice of deliberating or stopping and putting the tickets to the best use identified so far. For example, if the agent deliberates once, with probability 0.33 its value for the tickets will increase from 0 to 3, and with probability 0.67 its value will increase from 0 to 1. Deliberation in this example is costly (with per-step cost of 1.1). Each agent has a model of its costly deliberation process and a discount factor $\gamma \in (0, 1)$. Let us assume that a deliberation action only revises an agent's value weakly *upwards* and that each agent has only a single deliberation action available in every state.

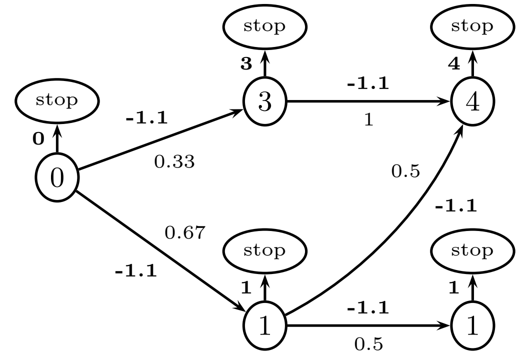


Figure 6: A Markov Decision Process to model the deliberation process of an agent. Each state is labeled with the value from the current best use of the block of tickets. This value is received from a *stop* action. The states labeled *stop* are terminal states. Upon a deliberation action (with cost 1.1) the agent undergoes a probabilistic transition to a new state with weakly higher value.

The dynamic VCG mechanism applies to this problem because each agent's value is independent of the type of other agents, conditioned on a sequence of actions. An optimal policy will pursue a sequence of deliberation actions, ultimately followed by an allocation action. The goal is to maximize the total expected discounted value of the allocation net of the cost of deliberation. Upon deliberation, the local state of an agent changes and thus we see the characteristic of internal uncertainty and dynamic type. The mechanism will coordinate which agent should deliberate in each period until deciding to allocate the item. Upon activating an agent, either by requesting a deliberation step or allocating the item, a payment may be demanded of the activated agent. The payment aligns incentives, and is such that the expected utility to an agent is always non-negative, even though it may be both engaged in deliberation *and* making payments. Upon deliberation, each agent will report its updated local type truthfully (e.g., its new value and revised belief about how future deliberation will improve its value).

By assuming sufficiently patient agents (with discount factor γ close enough to 1) and sufficiently costly deliberation, the problem has a structure reminiscent of the MABP because a single agent is "activated" in each period (either to deliberate or to receive the items) and an agent's state only changes when activated. One difference is that each agent has two actions (deliberate or stop) from each state. It is possible to convert an agent MDP into a Markov chain by a simple transformation in two steps. First, we prune away the actions that would not be optimal in a world in which this was the only agent. See Figure 7. In the example, the discount factor $\gamma = 0.95$. As long as the value of an agent only increases (as is the case in this domain) then this pruning step is sound (Cavallo and Parkes 2008). The second step is to convert the finite horizon Markov chain into an infinite horizon Markov chain by unrolling any terminal *stop* action with one-time value w into an absorbing state, with reward $(1-\gamma)w$ received in every period into perpetuity. See Figure 8. This step is valid because these states will remain absorbing states under an optimal policy for the MABP.

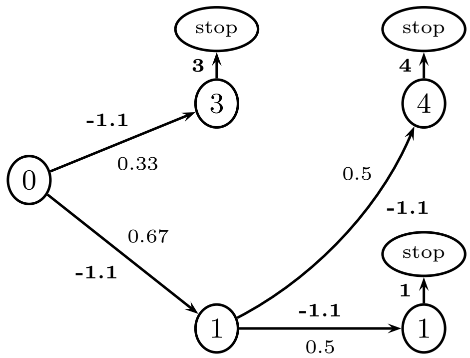


Figure 7: The single-agent deliberation model in which the suboptimal actions are removed. The result is a Markov chain where, for example, (i) the agent will deliberate in state 0, undergoing a probabilistic transition to a state with value 3 or 1, and (ii) the agent will stop deliberating and receive value 3 in state 3.

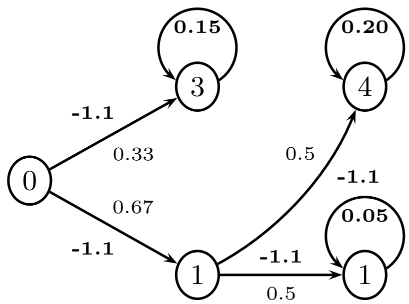
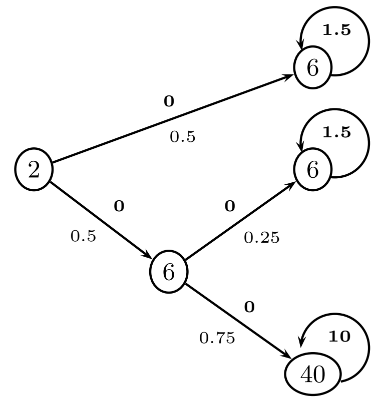
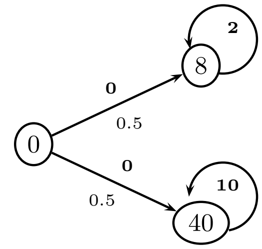


Figure 8: The single-agent deliberative model in which the terminal *stop* states in the Markov chain (the chain itself obtained by pruning suboptimal actions) are now transformed into absorbing states, obtaining an infinite-horizon Markov chain.

In the "metadeliberation auction" suggested in Cavallo and Parkes (2008), the dynamic VCG mechanism is implemented as follows: each agent first reports its local deliberation model to the mechanism along with its current deliberation state. The mechanism constructs a MABP by converting each agent's MDP into a Markov chain. The agent to activate in the current period is computed. If the state in the agent's pruned Markov chain is one from which a deliberation action is taken then this is suggested to the agent by the mechanism. Otherwise, the state is an absorbing state and the item is allocated to the agent. Payments are collected in either case, in order to align each agent's incentives.



(a) Agent 1



(b) Agent 2

Figure 9: An auction with deliberative agents. Agent 1's initial best-use value is 2, and may increase to as much as 40 through deliberation. Agent 2's initial best-use value is 0, and may increase to 8 or 40 through deliberation. The discount factor is $\gamma = 0.75$, while the cost of deliberation is zero.

Figure 9 illustrates a simple two agent example where the local MDPs have already been converted into Markov chains. Assume discount factor $\gamma = 0.75$ and deliberation costs are 0 for both agents. Initially A1 has value 2 and A2 has value 0. The optimal policy calls for A2 to deliberate, and make a payment of $1-\gamma$ times the expected value to A1 if decisions were optimized for him, i.e., $0.25 \cdot 11.1 = 2.775$. If A2 transitions to the value 40 state, in the second time period it is allocated the item and must pay 11.1. Alternatively, if it transitions to the value 8 state, the optimal policy calls for A1 to deliberate and make payment $0.25 \cdot 8 = 2$. If A1 then transitions to the absorbing state with value 6, the item is allocated to A2 who pays 6. If A1 instead transitions to the non-absorbing value 6 state, it deliberates again (again making payment 2). Finally if A2 then transitions to the value 40 state it is allocated the item and makes payment 8; otherwise A2 is allocated and makes payment 6.

A Challenge: Scaling Up

The incentive properties of the dynamic VCG mechanism extend to any problem in which each agent's private type evolves in a way that is independent of the type of other agents, when conditioned on the actions of the mechanism. But what is challenging in developing applications is the same problem that we identified in looking to apply the online VCG mechanism to CAs: the sequential decision problem becomes intractable in many domains, and substituting approximation algorithms does not sustain the incentive properties. Two important challenges include:

(a) *Tractable special cases and representation languages.* Can we identify additional models for multi-agent problems with dynamic private state for which the optimal decision policy is tractable? A related direction is to develop representation languages that allow agents to succinctly describe their local dynamics, e.g., models of learning and models of value refinement in the earlier examples.

(b) *Heuristic approaches.* Just as heuristic approaches seem essential for the design of practical dynamic mechanisms for environments with external uncertainty, so too will we need to develop methods to leverage heuristic algorithms for sequential decision making with agents that face internal uncertainty. For this, it seems likely that we will need to adopt approximate notions of incentive compatibility, and develop characterizations of coordination policies with "good enough" incentive properties.

The AI community seems especially well placed to be creative and flexible in creating effective dynamic incentive mechanisms. Indeed, Roth (2002) has written of the need for developing an "engineering" for economics and it seems that AI research has plenty to offer in this direction.

We need not be dogmatic. For example, it seems to us that incentive compatibility is nice to have if available, but we will need to adopt more relaxed criteria by which to judge the stability of mechanisms in the presence of self-interested agents. A good alternative will enable new approaches to the design and analysis of mechanisms (Lubin and Parkes 2009). Indeed, Internet ad auctions are inspired by, but not fully faithful to, the theories of incentive compatible mechanism design. On the other hand, folklore suggests that search engines initially adopted a second-price style auction over a first price auction because the first price auction created too much churn on the servers as bidders and bidding agents automatically chased each other around the bid space! So incentive compatibility, at least in some form of local stability, became an important and pragmatic criterion for designers.

Certainly, there are real-world problems of interest where insisting on truthfulness comes at a great cost to system welfare. Budish and Cantillon (2009) present a nice exposition of this in the context of course registration markets at Harvard Business School, where the essentially unique strategyproof mechanism—the "randomized serial dictatorship"—has bad welfare properties because of the callousness of relatively insignificant choices of early agents in a random priority ordering.

Conclusions

The promise of dynamic incentive mechanisms is that they can provide simplification, robustness and optimality in dynamic, multi-agent environments by engineering the right rules of encounter. Good success has been found in generalizing the canonical VCG mechanism to dynamic environments, and in adopting the property of monotonicity to enable the coupling of heuristic approaches to stochastic optimization with incentive compatibility in restricted domains. Still, many challenges remain, both in terms of developing useful characterizations of "good enough" incentive compatibility and in leveraging these characterizations within computational frameworks.

Dynamic mechanisms are fascinating in their ability to embrace both uncertainty that occurs outside of the scope of individual agents and also to "reach within" an agent and coordinate its own learning and deliberation processes.[17] But here we see the beginning of a problem of scope. Presumably we do not really believe that centralized decision making, with coordination even down to the details of an agent's deliberation process, is sensible in large scale, complex environments.

This is where it is also important to pivot away from direct revelation mechanisms– in which information is elicited by a center which makes and enforces decisions –to *indirect revelation* mechanisms. An indirect mechanism allows an agent to interact while revealing only the minimal information required to facilitate coordination. We wonder, then, whether dynamic mechanisms can be developed that economize on preference elicitation by allowing agents to send messages that convey approximate or incomplete information about their type in response to queries from a mechanism?[18]

A couple of other limitations about the mechanisms showcased in the preceding discussion should also be highlighted. First, we have been focused exclusively on goals of social welfare, often termed *economic efficiency.* Little is known about how to achieve alternative goals, such as revenue or various measures of fairness, in dynamic contexts. Secondly, we have assumed mechanisms in which there is money, that can be used for aligning agent incentives. But we saw from the median-choice rule mechanism and its application to the blue whale skeleton that there are interesting static

[17] For a more technical overview of dynamic mechanisms, see Parkes (2007). Lavi and Nisan (2000) first introduced the question of dynamic mechanisms to computer science, giving a focus to the design of *prior free* and incentive compatible online algorithms.

[18] Such mechanisms have been developed to good effect for settings of static mechanism design, but are still in their infancy for dynamic mechanism design (Said 2008).

mechanisms for contexts without money. Indeed, a recent line of research within computer science is developing around the notion of mechanism design without money (Procaccia and Tennenholtz 2009). But there is relatively little known about how to design dynamic mechanisms without money.[19] We might imagine that the curator of the exhibition on the university mall is interested in bringing through a progression of massive mammals. How should a dynamic mechanism be structured to facilitate a sequence of decisions about what, and where, to exhibit each year?

Acknowledgments

This work would not be possible without many great collaborators. The first author would like to warmly acknowledge Jonathan Bredin, Quang Duong, Eric Friedman, Mohammad T. Hajiaghayi, Takayuki Ito, Adam Juda, Mark Klein, Robert Kleinberg, Mohammad Mahdian, Gabriel Moreno, Chaki Ng, Margo Seltzer, Sven Seuken, and Dimah Yanovsky.

References

Adbulkadiroğlu, A., and Sönmez, T. 1999. House allocation with existing tenants. *Journal of Economic Theory* 88:233–260.

Bergemann, D., and Välimäki, J. 2008. The dynamic pivot mechanism. Technical Report Cowles Foundation Discussion Paper No. 1672, Yale University.

Budish, E., and Cantillon, E. 2009. The multi-unit assignment problem: Theory and evidence from course allocation at harvard. Technical report, University of Chicago Booth School of Business.

Cavallo, R., and Parkes, D. C. 2008. Efficient metadeliberation auctions. In *Proc. 23rd AAAI Conference on Artificial Intelligence (AAAI'08)*, 50–56.

Cavallo, R.; Parkes, D. C.; and Singh, S. 2009. Efficient mechanisms with dynamic populations and dynamic types. Technical report, Harvard University.

Constantin, F., and Parkes, D. C. 2009. Self-Correcting Sampling-Based Dynamic Multi-Unit Auctions. In *10th ACM Electronic Commerce Conference (EC'09)*, 89–98.

Cramton, P.; Shoham, Y.; and Steinberg, R., eds. 2006. *Combinatorial Auctions*. MIT Press.

Dizdar, D.; Gershkov, A.; and Moldovanu, B. 2009. Revenue maximization in the dynamic knapsack problem. Technical report, University of Bonn.

Dolgov, D. A., and Durfee, E. H. 2005. Computationally-efficient combinatorial auctions for resource allocation in weakly-coupled MDPs. In *Proc. 4th Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS'05)*.

Ephrati, E., and Rosenschein, J. S. 1991. The Clarke tax as a consensus mechanism among automated agents. In *Proc. 9th National Conference on Artificial Intelligence (AAAI-91)*, 173–178.

[19]But see Jackson and Sonenschein (2007), Zou et al. (2010), Abdulkadiroğlu and Sönmez (1999), and Lu and Boutilier (2010) for some possible inspiration.

Gittins, J. C. 1989. *Multi-armed Bandit Allocation Indices*. New York: Wiley.

Hajiaghayi, M. T.; Kleinberg, R.; Mahdian, M.; and Parkes, D. C. 2005. Online auctions with re-usable goods. In *Proc. ACM Conf. on Electronic Commerce*, 165–174.

Hentenryck, P. V., and Bent, R. 2006. *Online Stochastic Combinatorial Optimization*. MIT Press.

Horvitz, E. J. 1988. Reasoning under varying and uncertain resource constraints. In *Proc. 7th National Conference on Artificial Intelligence (AAAI-88)*, 139–144.

Hurwicz, L. 1973. The design of mechanisms for resource allocation. *American Economic Review Papers and Proceedings* 63:1–30.

Jackson, M. O., and Sonnenschein, H. F. 2007. Overcoming incentive constraints by linking decisions. *Econometrica* 75(1):241–258.

Jackson, M. O. 2000. Mechanism theory. In *The Encyclopedia of Life Support Systems*. EOLSS Publishers.

Larson, K., and Sandholm, T. 2004. Using performance profile trees to improve deliberation control. In *Proc. 19th Nat. Conf. on Art. Intell. (AAAI'04)*.

Lavi, R., and Nisan, N. 2000. Competitive analysis of incentive compatible on-line auctions. In *Proc. 2nd ACM Conf. on Electronic Commerce (EC-00)*.

Lehmann, D.; O'Callaghan, L. I.; and Shoham, Y. 2002. Truth revelation in approximately efficient combinatorial auctions. *Journal of the ACM* 49(5):577–602.

Lu, T., and Boutilier, C. 2010. The unavailable candidate model: A decision-theoretic view of social choice. In *Proc. 11th ACM Conference on Electronic Commerce (EC'10)*.

Lubin, B., and Parkes, D. C. 2009. Quantifying the Strategyproofness of Mechanisms via Metrics on Payoff Distributions. In *25th Conference on Uncertainty in Artificial Intelligence*, 349–358.

Moulin, H. 1980. On Strategy-Proofness and Single-Peakedness. *Public Choice* 35:437–455.

Myerson, R. B. 1981. Optimal auction design. *Mathematics of Operation Research* 6:58–73.

Parkes, D. C., and Duong, Q. 2007. An ironing-based approach to adaptive online mechanism design in single-valued domains. In *Proc. 22nd National Conference on Artificial Intelligence (AAAI'07)*, 94–101.

Parkes, D. C., and Singh, S. 2003. An MDP-based approach to Online Mechanism Design. In *Proc. 17th Annual Conf. on Neural Information Processing Systems (NIPS'03)*.

Parkes, D. C. 2007. Online mechanisms. In Nisan, N.; Roughgarden, T.; Tardos, E.; and Vazirani, V., eds., *Algorithmic Game Theory*. Cambridge University Press. chapter 16, 411–439.

Parkes, D. C. 2009. When analysis fails: Heuristic mechanism design via self-correcting procedures. In *Proc. 35th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM'09)*, 62–66.

Procaccia, A. D., and Tennenholtz, M. 2009. Approximate mechanism design without money. In *Proc. 10th ACM Conference on Electronic Commerce*, 177–186.

Rosenschein, J. S., and Zlotkin, G. 1994. Designing conventions for automated negotiation. *AI Magazine*. Fall.

Roth, A. E. 2002. The economist as engineer. *Econometrica* 70:1341–1378.

Russell, S., and Wefald, E. 1991. *Do the right thing*. MIT Press.

Russell, S. J.; Subramanian, D.; and Parr, R. 1993. Provably bounded optimal agents. In *Proc. 13th International Joint Conference on Artificial Intelligence (IJCAI-93)*, 338–344.

Said, M. 2008. Information revelation in sequential ascending auctions. Technical report, Yale.

Shahaf, D., and Horvitz, E. 2010. Generalized task markets for human and machine computation. In *Proc. 24th AAAI Conference on Artificial Intelligence*.

Vickrey, W. 1961. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance* 16:8–37.

von Ahn, L., and Dabbish, L. 2008. Designing games with a purpose. *Communications of the ACM* 51:58–67.

Vytelingum, P.; Voice, T. D.; Ramchurn, S. D.; Rogers, A.; and Jennings, N. R. 2010. Agent-based micro-storage management for the smart grid. In *Proc. 9th Int. Conf. on Autonomous Agents and Multiagent Systems*.

Zou, J.; Gujar, S.; and Parkes, D. 2010. Tolerable manipulability in dynamic assignment without money. In *24th AAAI Conference on Artificial Intelligence (AAAI-10)*.