Mechanism Design and Deliberative Agents

Kate Larson
School of Computer Science
University of Waterloo
Waterloo, ON N2L 3G1
Canada
klarson@cs.uwaterloo.ca

Tuomas Sandholm
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213
USA
sandholm@cs.cmu.edu

ABSTRACT

The central mechanism design problem is to develop incentives for agents to truthfully reveal their preferences over different outcomes, so that the system-wide outcome chosen by the mechanism appropriately reflects these preferences. However, in many settings, agents' do not know their actual preferences *a priori*. Instead, an agent may need to compute or gather information to determine whether they prefer one possible outcome over another. Due to time constraints or the cost of acquiring information, agents must be *deliberative* in that they need to carefully decide how to allocate their computational or information gathering resources when determining their preferences. In this paper we study the problem of designing mechanisms explicitly for deliberative agents. We propose a set of intuitive properties which we argue are desirable in deliberative-agent settings. We show that these properties are mutually incompatible, and that many approaches to mechanism design are not robust against undesirable behavior from deliberative agents.

Categories and Subject Descriptors: 1.2.11 Multiagent Systems
General Terms: Design, Theory
Keywords: Mechanism Design, Game Theory, Resource-bounded Agents

1. INTRODUCTION

Game theory and mechanism design have played an important role in the field of multiagent systems. They have provided a foundation for analyzing negotiation and resource allocation protocols, and have supplied a set of tools and techniques for engineering incentives so that agents behave in the way we, the system designers, want them to. However, game theory and mechanism design have traditionally ignored constraints on the computational resources and information gathering technology of agents in many real-world applications.

Recently, researchers have started studying how computational and informational constraints influence the behavior of both the mechanism and the agents in multiagent systems. One direction of research has focused on what happens if the mechanism does not

have infinite computational powers. In many interesting applications, such as combinatorial auctions, the mechanism is required to solve (possibly multiple) $\mathcal{NP}$-hard problems. Determining how to replace the mechanism with approximation algorithms, while still maintaining the desirable game-theoretic properties has been a vibrant research area [6, 10, 12].

Researchers have also started to study computational and informational constraints placed on the agents participating in a mechanism. Mechanism design has traditionally assumed that the agents know their preferences over different outcomes, and the mechanism problem is to simply get the agents to reveal this information. However, in many settings agents do not know their preferences and instead must learn them by computing or acquiring information (at some cost). One model which has been studied allows an agent the choice between participating in the mechanism without knowing its true preferences or paying a fee to learn them. Questions asked using this model include what sort of incentives are required for agents to acquire information about their own preferences [1], and how does information acquisition depend on the rules of the mechanism [2, 13, 14].

It has also been noted that an agent's decision as to whether to compute or gather information about it's own preferences can depend on the preferences of other agents [15]. We have proposed explicitly modeling the computing and information gathering actions of agents, along with the decisions they make when deciding how to use their computing or information gathering resources. In particular, our model allows agents to compute or gather information on other agents' preferences. We placed this *deliberative-agent* model into a game theoretic framework, and have analyzed classic bargaining and auction mechanisms in order to gain an understanding of the impact that computational and information gathering constraints have on agents' strategic behavior [7, 8].

In this paper, using the deliberative-agent model, we ask the question "Is it possible to design mechanisms explicitly for deliberative agents?" In particular, we wish to know whether it is possible to create mechanisms which reduce the strategic and deliberative burden of agents. We propose a set of three intuitive properties that we would like mechanisms for deliberative agents to have. We show that in general it is not possible to design *interesting* mechanisms which have all three of our desired properties. While our main result is negative, it opens up several fascinating research directions.

The rest of the paper is organized in the following way. In the next two sections we provide background material needed for the results. In particular we present the basic mechanism design concepts used in the paper, and describe the deliberative-agent model. In Section 4 we present our results. We describe the properties we believe mechanisms for deliberative agents should have, show

that any direct mechanism will be prone to a phenomena called strategic-deliberation, and show that, in general, interesting mechanisms can not satisfy our desired properties. We conclude the paper with a discussion of the implications of our results and describe some interesting future research directions.

2. MECHANISM DESIGN

Mechanism design studies the problem of how to provide appropriate incentives to agents so that they willingly reveal their private preferences. Once the preferences of the agents are known, the mechanism selects a system-wide outcome. However, agents are self-interested and may try to manipulate the mechanism in order to force an outcome which is desirable for themselves, as opposed to the system as a whole. In this section we provide a brief overview of important mechanism design concepts. A more thorough presentation of mechanism design can be found in many game theory and economic texts [11].

We assume that there is a set of agents denoted by N (with $|N| = n$). Each agent $i \in N$ is defined by a *type*, θ_i, which represents the private information and preferences of the agent. Each agent has a utility function, $u_i(o, \theta_i)$ which depends on the outcome o, and its type θ_i. If we knew the actual types of the agents, $\theta = (\theta_1, \ldots, \theta_n)$ then we would be able to apply a *social choice function*, $f(\theta)$, which selects the optimal outcome given the agents types. The problem, however, is that we do not know the true types of the agents. Instead, we use a mechanism to try to reach the correct outcome.

A mechanism, M, defines a game which, when agents play in equilibrium, results in the same outcome as the social choice function $f(\theta)$. In particular, a mechanism $M = (S, g(\cdot))$ defines the sets of allowable strategies, $S = S_1 \times \ldots \times S_n$ where S_i is the strategy space of agent i, and $g(s)$ is a function which specifies an outcome, o, for each strategy profile $s = (s_1, \ldots, s_n) \in S$. An agent i is free to select any strategy in S_i, and, in particular, will try to select a strategy which will lead to an outcome that maximizes its own utility. The mechanism *implements* social choice function $f(\theta)$ if there exists an equilibrium s^* such that $g(s^*) = f(\theta)$. A special subclass of mechanisms is the *direct mechanisms*. In a direct mechanism, each agent i announces a type, $\hat{\theta}_i$ to the mechanism. It is not necessary that $\hat{\theta}_i = \theta_i$. However, in *incentive-compatible* (direct) mechanisms there exists an equilibrium s^* where $s^* = (\theta_1, \ldots, \theta_n)$. The important *Revelation Principle* states that if a social choice function $f(\cdot)$ can be implemented, then it can be implemented by an incentive-compatible direct mechanism. Another feature of a mechanism is that agents can not be forced to participate. Thus, mechanisms are often designed so that it is in an agent's best interest to take part. That is, the utility of an agent who participates in the mechanism is guaranteed to be (in expectation) no less than the utility it could get from not participating. A mechanism with this property is called *individually-rational*.

The structure of agents' preferences influence the structure of the mechanism. In this paper we assume that the preferences of the agents take the form of *quasi-linear utilities*. That is, the preference of agent i can be captured by a function of the form

$$u_i(o, \theta_i) = v_i(x, \theta_i) - p_i$$

where $o = (x, p)$ is an outcome which specifies a choice x (for example an allocation of items or resources) and a transfer $p_i \in \mathbb{R}$ for each agent, where $p = (p_1, \ldots, p_n)$. An agent's utility depends on its *valuation function* $v_i(x, \theta_i)$ which specifies how much an agent values choice x given its type θ_i. A mechanism for a quasi-linear setting is defined by $M = (S, g(\cdot))$ where $g(\cdot) = (x(\cdot), p(\cdot))$.

3. DELIBERATIVE AGENTS

In order to effectively participate in a mechanism, an agent needs to know its preferences over the different outcomes the mechanism may produce. However, there are many applications where agents do not know their preferences *a priori* but instead must compute or gather information. An example of a situation where an agent may not know its preferences *a priori* is a vacation planning setting. An agent may represent a user who wishes to go on a holiday but is unsure as to how much it is willing to pay for the holiday. As more information is gathered about the cost of flights and hotel rooms, and as more is learned about the destination, the agent (and its user) refines its preferences about taking the trip. The realities of limited computational resources and time pressures caused by real-time environments mean that agents are not always able to optimally determine how much they really value different alternatives.

In this paper we assume that agents are *deliberative*. A deliberative agent is an agent who must compute or gather information in order to determine its preferences, has restrictions on its computing or information gathering capabilities, and who carefully considers how to use its available resources given its restrictions.

We assume that a deliberative agent has a set of deliberation resources. In the rest of the paper when we use the term "resources" we mean deliberation resources. We denote the resources of agent i by T_i. An agent is able to apply its resources to any preference-determination problem it wishes. If there are m possible problems, then we let $(t_1, \ldots, t_m) \in T_i^m$ denote the situation where the agent has devoted t_j resources to problem j. In particular, the agent is allowed to deliberate on its own preference problems, as well as the problems of any other agent. many environments agents

We model deliberation-resource limitations through cost functions. The cost function of agent i is $\text{cost}_i : T_i^m \mapsto \mathbb{R}^+$. The only restriction placed on the cost functions are that they must be additive and non-decreasing.

A deliberative agent i is endowed with a multi-set of algorithms $\mathcal{A}_i = \{A_i^j\}$ where A_i^j is the algorithm agent i can use on problem j. We use the term *algorithm* in its broadest sense; algorithms are step-by-step procedures for solving some problem. In particular, we include information gathering processes in our set of algorithms. The algorithms of a deliberative agent have the anytime property; they can be stopped at any point and are guaranteed to return a solution, but if additional resources are allocated to the problem then a better solution (or at least no worse a solution) will be returned.

A deliberative agent carefully decides how to allocate its resources on algorithms given its cost function. To help with this process, deliberative agents are equipped with a set of *performance profiles*, $\mathcal{PP}_i = \{P_i^j\}$, one performance profile for each algorithm. A performance profile has two components. First, it describes how allocating resources to a problem changes the output of the algorithm. In particular, it describes, for any resource allocation to a problem, what possible solutions the algorithm may return, conditional on any and all features of the algorithm and problem instance which are deemed to be of importance to the agent. This is coupled with a procedural component, which, given the descriptive part of the performance profile, returns a deliberation policy that describes how the agent should optimally allocate deliberation resources to the algorithm. We assume that agents have fully normative performance profiles [7]. To summarize, a deliberative agent is defined by

$$\langle T_i, \text{cost}_i(\cdot), \mathcal{A}_i, \mathcal{PP}_i \rangle$$

where T_i is the set of deliberation resources of agent i,

$$\text{cost}_i : T_i^m \mapsto \mathbb{R}$$

is a cost function which limits the amount of resources the agent can use, $\mathcal{A}_i$ is the multi-set of algorithms available to the agent, and $\mathcal{PP}_i$ is the associated set of performance profiles.

Agents use their deliberation resources to determine their preferences. As mentioned in the previous section, we assume that agents have private-value quasi-linear utilities. In practice, agents deliberate to determine their valuation functions. The utility for agent i upon determining valuation function $v_i(t_i)$ after allocating $t = (t_1, \ldots, t_m)$ resources when the outcome is (x, p) is

$$u_i((x, p), t) = v_i(x, t_i) - p_i - \text{cost}_i(t).$$

Note that the agent may allocate resources to problems which do not directly affect its own valuation function. In particular, an agent may decide to deliberate on the valuation or preference problem of another agent.

We differentiate between deliberative actions and non-deliberative actions. A deliberation action for agent i, $d_i^j(t_j)$, is the act of allocating t_j deliberation resources to problem j. The set of possible deliberation actions that an agent can take is denoted by D_i and includes the action of not deliberating on any problem ($\emptyset_D$). As an agent takes deliberation actions, it collects information about its current preferences as well as how future deliberation actions will likely change this information. This information is stored in a *state of deliberation*,

$$\phi_i(t) = \langle n_1(t_1), \ldots, n_m(t_m) \rangle,$$

where $n_j(t_j)$ denotes all information agent i has about its preferences with respect to problem or instance j given that it has allocated t_j resources. This information is derived from the performance profiles and the deliberation results of the agent.

We define the set X_i to be the set of non-deliberative actions that an agent i can take. This set is determined by the mechanism. For example, in a sealed-bid auction, the set X_i is simply the set of bids that the agent may submit to the auctioneer, while in an ascending auction the set X_i is the set of messages that an agent can send to the mechanism whenever the price increases (i.e. $X_i = \{\text{in}, \text{out}\}$).

A strategy for a deliberative agent is a policy which specifies which actions to execute (deliberative and other) at every point in the game. A *history* at stage st, $H(st) \in \mathcal{H}(st)$, defines the state of deliberation of the agent at stage st, the set of all non-deliberative actions the agent has taken before stage st, as well as all actions it has observed other agents taking. Using this definition of a history, it is possible to define a *deliberation strategy*.

Definition 1 (Deliberation strategy) *A deliberation strategy for a deliberative agent i is*

$$S_i = (\sigma_i^{st})_{st=0}^{\infty}$$

where

$$\sigma_i^{st} : \mathcal{H}_i(st) \mapsto D_i \times X_i.$$

To clarify the definition, we present a simple example.

A strategy $S_i = (\sigma_i^{st})_{st=0}^{\infty}$ for a deliberative agent in a sealed-bid auction, where bids are collected at stage st^* is defined as follows;

$$\sigma_i^{st}(H(st)) = \begin{cases} (d_i^j(t_j), \emptyset_A) & \text{when } st < st^* \\ (d_i^j(t_j), b_i) \; b_i \in \mathbb{R} & \text{when } st = st^* \\ (d_i^j(t_j), \emptyset_A) & \text{when } st > st^* \end{cases}$$

where j is any problem that the agent can deliberate on. That is, before the auction the agent can take any deliberation action it wishes, at the bid collection stage the agent submits a bid, and after the auction closes, the agent may (or may not) take further deliberation actions.

In this new, enlarged, strategy space we look for equilibria, which are named *deliberation equilibria* [7]. In studying how deliberative agents behave in classic auction mechanisms, we noted a new type of strategic behavior where, in equilibrium, one agent actively devotes deliberation resources on another agent's preference problem [8]. This behavior was called *strategic deliberation*.

Definition 2 (Strategic Deliberation) *If an agent i uses any of its deliberation resources on another agent's preference problem, then agent i is strategically deliberating.*

In the rest of the paper we make several assumptions. First, as mentioned earlier, we assume that agents have private-value quasi-linear utilities. We also assume that the agent descriptions are common knowledge. That is, we assume that the performance profiles and cost functions of the agents are common knowledge. We do not assume that agents are able to observe which computing actions other agents are taking during a game.

4. MECHANISM DESIGN FOR DELIBERATIVE AGENTS

In this section we study the problem of designing mechanisms explicitly for deliberative agents. Our goal is to achieve a thorough understanding of the impact that agents' limited resources have on mechanism properties, as well as how it influences the type of mechanisms we can design.

4.1 Agents' Types and a Revelation Principle

A fundamental tool in mechanism design is the Revelation Principle. It states that under very weak conditions mechanism designers need only focus on incentive-compatible direct mechanisms in order to discover which social-choice functions are implementable. That is, mechanism designers can restrict their attention to mechanisms where the agents reveal their *types* truthfully. The difficulty with applying the classic Revelation Principle to a deliberative-agent setting is that it is unclear what the type of an agent is, since the classic definition of type captures information about an agent's preferences over outcomes, which may be unknown to the agent itself.

We propose defining the type of a deliberative agent to be everything that it uses to determine its preferences. That is, the type of a deliberative agent i depends on it's algorithms, $\mathcal{A}_i$, it's performance profiles, $\mathcal{PP}_i$, and it's cost function $\text{cost}_i(\cdot)$. Furthermore, the type of an agent is defined by an *instance* which specifies the features of the problems the agent is computing or gathering information on. We denote the instance set by $\mathcal{I}$ and let $\{x_1, \ldots, x_m\} \in \mathcal{I}$ denote a specific instance where x_j is the instance of problem j. We define the type space of agent i to be

$$\Theta_i = \mathcal{A}_i \times \mathcal{PP}_i \times \{\text{cost}_i(\cdot)\} \times \mathcal{I}$$

and a type instance to be

$$\theta_i = (A_i^j, PP_i^j, \text{cost}_i(\cdot), \{x_1, \ldots, x_m\}) \in \Theta_i$$

Using this definition of an agent's type, it is possible to derive a Revelation Principle for deliberative agents.

Theorem 1 (Revelation Principle) *Suppose there exists a mechanism $M = (S_1, \ldots, S_n, g(\cdot))$ that implements the social choice*

function $f(\cdot)$ in dominant strategies. Then $f(\cdot)$ is truthfully implementable in dominant strategies.

Proof: The proof follows an argument similar to that of the original Revelation Principle. Assume that indirect mechanism M implements $f(\cdot)$ in dominant strategies. Then there exists a strategy profile $s^* = (s_1^*, \ldots, s_n^*)$ such that $u_i(g(s_i^*(\theta_i), s_{-i}(\theta_{-i})) - \text{cost}_i(s_i^*(\theta_i)) \geq u_i(g(s_i'(\theta_i), s_{-i}(\theta_{-i})) - \text{cost}_i(s_i'(\theta_i))$ for all s_i' and s_{-i}, and $f(\theta_i) = g((s_1^*(\theta_1), \ldots, s_n^*(\theta_n)))$. Create a new mechanism M' such that, when given type θ_i, mechanism M' executes strategy s_i^* and charges $\text{cost}_i(s_i^*(\theta_i))$. If $s_i^*(\theta_i)$ is the optimal strategy for agent i for each $\theta_i \in \Theta_i$ in mechanism M, given any set of strategies chosen by the other agents, then truth telling will be a dominant strategy in M'. Thus, there is a mechanism that truthfully implements $f(\cdot)$. Using a similar argument, it is possible to derive a Bayes-Nash Revelation Principle. $\square$

At first reading, Theorem 1 suggests that designing mechanisms for deliberative agents is the identical problem as designing mechanisms for fully rational agents. However, there are several unrealistic assumptions made in Theorem 1 which are above and beyond the criticisms of the Revelation Principle for classic settings [3].

First, Theorem 1 assumes that an agent is capable of revealing its type to the mechanism in a single step. While this can be a large burden for a classical agent, in a deliberative-agent setting it implies that an agent must communicate a full description of its cost function, its algorithms, its performance profiles, and all features of its current problem instances. It is unreasonable to assume that a deliberative agent would be capable of doing such a thing. Second, Theorem 1 assumes that the mechanism itself has enough computational and information gathering resources of its own to determine the preferences of all agents. This is in addition to determining the outcome once it knows the preferences of all agents, which may be a computationally difficult problem in and of itself.

4.2 Properties for Mechanisms

The problem with the approach proposed by Theorem 1 is that it ignores the fact that the agents participating in the mechanism are deliberative. We believe that good mechanisms for deliberative agents should have not only traditional desirable properties such as individual rationality and incentive-compatibility, but should also exhibit desirable deliberative properties. In this section we outline some intuitive deliberative properties we believe a mechanism should exhibit.

A concern with the approach taken in Theorem 1 was the assumption that the mechanism could take responsibility for solving deliberation problems for the agents. We believe that this assumption shifts too high a computational burden to the mechanism center, and that the main role of the mechanism center should be to determine an outcome, once it knows the preferences of the agents. That is, we believe that a mechanism should be *preference formation-independent* in that the mechanism should not be involved with the process by which agents form their preferences.

Property 1 (Preference formation-independent) *A mechanism should be preference formation-independent. That is, a mechanism should not be involved in the process by which agents form their preferences. In particular, the mechanism should not solve agents' individual deliberation problems.*

In Section 3 we noted that in several common auction mechanisms, agents have incentive to use some of their own deliberation resources to determine the preferences of others. This behavior was coined strategic-deliberation. We believe that a well-designed mechanism should reduce the strategic burden on the agents, and

thus try to remove the incentives for agents to strategically deliberate. We say that a mechanism where agents have no incentive to strategically deliberate is deliberation-proof.

Property 2 (Deliberation-proof) *A mechanism is deliberation-proof if, in equilibrium, no agent has incentive to strategically-deliberate.*

Finally, we believe that mechanisms should promote a certain level of consistency in the strategies of the agents. By this we mean that the mechanism should not provide incentives for an agent to follow a strategy which causes other agents to believe that, under the assumption the agent is a utility-maximizer, the true preferences of the agent are impossible. For example, in a single-item auction an agent needs to have a value, v, for the item being auctioned. A mechanism is misleading if it provides incentives so that an agent reports that it values the item strictly more than v and that it is potentially willing to pay more than v. A mechanism is non-misleading if it provides incentives to the agent to claim that its true value is less than or equal to v.

Property 3 (Non-misleading) *A mechanism is non-misleading if, in equilibrium, no agent has incentive to follow a strategy which, if observed by other agents, would lead them to believe that its true preferences are impossible. That is, if pref_i^* is the true preferences of agent i, and $\text{belief}_j(\text{pref}_i)$ is the probability agent j places on the event that the true preferences of agent i are pref_i, then agent i has no incentive to follow a strategy such that, upon agent j correctly updating its beliefs, $\text{belief}_j(\text{pref}_i^*) = 0$.*

Truthful mechanisms are a subset of non-misleading mechanisms.

4.3 Mechanisms for Deliberative Agents

In this section we investigate what mechanisms can be designed that satisfy the properties outlined in Section 4.2. We first note that it is trivially possible to design a mechanism which is preference formation-independent, deliberation-proof, and non-misleading. Any dictatorial mechanism which always selects the preferred outcome of one agent satisfies the proposed properties. Similarly any completely random mechanism where the outcome is independent of the agents' preferences also satisfies the properties. Therefore, in the rest of this paper we restrict ourselves to *strategy-dependent* mechanisms where the outcome selected by the mechanism depends on the strategies chosen by the agents.

Definition 3 (Strategy-dependent) *A mechanism $M = (S_1, \ldots, S_n, g(\cdot))$ is strategy dependent if for each agent i there exists strategies s_i', s_i'' such that for strategy profiles $s' = (s_1, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n)$ and $s'' = (s_1, \ldots, s_{i-1}, s_i'', s_{i+1}, \ldots, s_n)$, it is the case that*

$$g(s') \neq g(s'').$$

Most interesting mechanisms are strategy-dependent. For example, any mechanism which maximizes social welfare must be strategy-dependent. We note that mechanisms involving randomization can be strategy-dependent. For example, an auction which allocates an item randomly, based on a probability weighted by its bid, is strategy-dependent.

We now present our first theorem describing (non-) achievable properties of mechanisms for deliberative agents.

Theorem 2 *There exists no direct strategy-dependent mechanism which is preference formation-independent, deliberation-proof and non-misleading across all possible deliberative-agent types.*

Proof: Let $M = (S_1, \ldots, S_n, g(\cdot))$ be a direct mechanism. Since we are in a quasi-linear environment, $g(\cdot) = (x(\cdot), p(\cdot))$ where $x(\cdot)$ is a choice function and $p(\cdot)$ is the transfer function. If M is preference formation independent then agents can not reveal information about their algorithms or performance profiles to the mechanism. Instead agents are restricted to reporting their determined preferences (valuations) or expected determined preferences (expected valuations). If M is non-misleading then in a direct mechanism the agents have incentive to truthfully reveal their preferences (expected preferences).

Define performance profiles P_1 and P_2 as follows

$$P_1 = \begin{cases} 0 & \text{if } \sum t_i = 0 \\ v_1^h \text{ with prob. } q & \text{if } t_1 > 0 \\ v_1^l \text{ with prob. } 1-q & \text{if } t_1 > 0 \end{cases}$$

and

$$P_2 = \begin{cases} 0 & \text{if } \sum t_i = 0 \\ v_2^h \text{ with prob. } r & \text{if } t_2 > 0 \\ v_2^l \text{ with prob. } 1-r & \text{if } t_2 > 0 \end{cases}$$

where $v_1^l, v_1^h, v_2^l, v_2^h$ are chosen such that

$$x((v_1^h, \cdot)) = 1$$
$$x((\cdot, v_2^l)) = 1$$
$$x((v_1^l, v_2^h)) = 2$$

where $x(\cdot) = i$ means that an allocation of choice is made in favor of agent i. Also, since the mechanism is non-misleading, it must be the case that the transfer functions $p_i()$ can not depend on the declared preference of agent i. That is $p_i(v_j)$, $i \neq j$. Define the cost functions of the agents to be $\text{cost}(t_1, t_2) = \epsilon(t_1 + t_2)$ for small $\epsilon > 0$, and $\text{cost}_2(t_1, t_2) = t_1 + Kt_2$ for some constant $K > 1$. For small ϵ agent 1 has a dominant strategy which is to deliberate for one step on its own problem and then truthfully reveal what it has obtained. A straightforward study of the constraints placed on the utility functions of agent 1 shows that this is a dominant strategy.

Given that agent 1 has a dominant strategy, agent 2 must determine its best response. By studying constraints placed on the agent's utility we can show that agent 2 has incentive to deliberate on the problem of agent 1 whenever

$$\frac{q + (1-q)r}{1 - (1-q)r} \leq K \leq r(v_2^h - p_2(v_1^l)) - \frac{1+q}{1-q}$$

where we are free to define q, r, v_2^h, v_1^l and K. Strategic deliberation occurs because agent 2 has incentive to learn about agent 1's deliberation results, before deciding on its own deliberation policy. In particular, if it learns that agent 1 has valuation v_1^h, then agent 2 has no incentive to deliberate on its own problem. $\square$

In earlier work we showed that the Vickrey auction and generalized Vickrey auction are not deliberation-proof [8]. Theorem 2 shows that the problem is more widespread than just efficient mechanisms, and does not depend on the specific allocation and payment rules of the Vickrey auction.

While Theorem 2 is negative in that it states that direct mechanisms do not have the properties we desire, it does give us insight into why strategic-deliberation may occur. In particular, the proof shows that the deliberation policy of an agent may depend on the preferences of another. The problem with direct mechanisms is that they only allow an agent to obtain relevant preference information from other agents through strategic-deliberation.

Many indirect (multi-stage) mechanisms reveal information. For example, in some forms of ascending auctions, an agent may be aware of how many other agents remain in the auction at a specific price. The information provided by the mechanism may be useful to agents and help them formulate their optimal deliberation strategies. In particular, this information can be used to remove the incentives for agents to strategically-deliberate.

We propose explicitly modeling the information provided by an indirect mechanism, M, as a *feedback game*. A feedback game, $(M, \mathcal{F}_M)$, is the extensive form game induced by mechanism M coupled with a feedback function $\mathcal{F}_M$. This extensive form game includes the deliberation actions of the agents. At every step in the game, the feedback function $\mathcal{F}_M$ maps the actions of the agents into an information structure which describes what information the mechanism reveals to each agent. For example, in a sealed-bid auction, where agents must submit bids by stage st', (the agents can deliberate before and after stage st'), the corresponding feedback function would be

$$\mathcal{F}(st, a(st)) = \begin{cases} \emptyset & \text{if } st < st' \\ (x, p) & \text{if } st \geq st' \end{cases}$$

where $a(st)$ is the set of actions taken by the agents at stage st, $\emptyset$ is the situation where no information is revealed and (x, p) is the final allocation and transfers. In an ascending auction where agents are aware of which other agents are participating, a feedback function takes the form

$$\mathcal{F}(st, a(st)) = (p, \{i | \text{agent } i \text{ is in auction at price } p\})$$

Introducing an explicit feedback function does not modify the original mechanism since nothing in the game induced by the mechanism is changed.

Lemma 1 *Given any mechanism M it is possible to construct a feedback function $\mathcal{F}_M$ such that the equilibria in the feedback game $(M, \mathcal{F}_M)$ are the same as the equilibria in the original mechanism.*

The feedback function provides us with a tool for reasoning about what information is available to the agents and how agents are acting based upon the information they have. Using this tool we are able to prove our main theorem.

Theorem 3 *There does not exist any strategy-dependent mechanism which is preference formation-independent, deliberation-proof and non-misleading across all possible deliberative-agent types.*

Proof: The full proof is long and technical. In this paper we describe only the main points. First we restrict ourselves to the space of preference-formation independent mechanisms. That is, agents are only permitted to reveal information about their deliberation results. Second, we assume for technical reasons that if an agent i announces to the mechanism that it has not deliberated on its own preference problem, then $p_i = 0$. This assumption guarantees a weak form of individual rationality.

Let M be a mechanism which implements social choice function $f(\cdot)$ and define the appropriate feedback function $\mathcal{F}_M$. If M is a direct mechanism then from Theorem 2, it is not deliberation-proof.

Assume that M is an indirect mechanism. Using a Revelation Principle like argument, it is possible to find a direct non-misleading mechanism M' that implements $f(\cdot)$. Using the approach outlined in Theorem 2 define algorithms, performance, profiles, cost functions so that strategic deliberation occurs in equilibrium in M'.

Let v_i^* be the valuation that agent i would have achieved if it had deliberated on its own problem. At each stage in the mechanism let $\hat{V}_i$ be the partition that agent i makes of agent j's possible values for v_j^*. In particular, $\hat{V}_i = \{V_i(t), V_i'(t)\}$ where if $v_j^* \in V_i(t)$ then agent i has incentive to deliberate on its own problem and if $v_j^* \in V_i'(t)$ then agent i is best off not deliberating. Given the

construction of the instances, there exist stages where $V_i'(t) \neq \emptyset$. In particular, any stage t where agent 2 has done no deliberating has $V_2'(t) \neq \emptyset$.

Next we look at the information structures generated by feedback function $\mathcal{F}$ at stages where $V_2'(t) \neq \emptyset$. We say that $\mathcal{F}_M$ is *separating* at stage t if for any non-misleading actions taken when agent 1's true value lies in $V_2(t)$ and for any non-misleading actions taken by agent 1 when agent 1's true value lies in $V_2'(t)$, the outcomes of $\mathcal{F}_M$ are different. We say that $\mathcal{F}_M$ is *pooling* otherwise. This generates a *signaling game* with a Bayes-Nash equilibrium [11]. In this game, given the constructed agents, if $\mathcal{F}_M$ is pooling then agent 2 has incentive to strategically deliberate, but if $\mathcal{F}_M$ is separating, then agent 1 has incentive to mislead by always signaling that it has valuation v_1^h.

$\square$

5. IMPLICATIONS

Theorems 2 and 3 state that it is not possible to design a strategy-dependent mechanism for deliberative-agent settings which have the desirable properties we proposed. In particular, we can not have a strategy-dependent mechanism which is preference formation-independent, deliberation-proof and non-misleading. While this is a negative result, it also opens up some interesting questions. In particular, it may be possible to relax one of the proposed properties while maintaining the others.

We believe that *non-misleading* is a fundamental property which should not be weakened, since it is already a weak constraint on the equilibrium strategies of agents. One of our motivations for studying the problem of designing mechanisms for deliberative agents was that we wished to reduce the strategic burden placed on the agents. Allowing mechanisms to be misleading runs counter to this motivation.

It may be possible to weaken or remove the *deliberation-proof* property. From the analysis in this paper and previous experimental results [9], it appears as though strategic deliberation is prone to occur in environments where agents are asymmetric in their deliberating capabilities. One research direction is to try and design mechanisms which are guaranteed to be deliberation-proof as long as there is a limited amount of asymmetry amongst the agents, but which have no guarantees otherwise. Categorizing the amount and type of asymmetry required for strategic-deliberation to occur is an important £rst step.

A different approach is to embrace strategic deliberation. For example, if some agents have adequate computational or information gathering capabilities so that it is easy or inexpensive for them to determine valuations and preferences for other agents, it might be bene£cial, from a system-wide perspective, to allow these agents to deliberate for other agents, and serve as experts in the mechanism [5]. There are many incentive issues that need to be addressed before such an approach is feasible. In particular, the mechanism must provide adequate incentives so that the "right" agents solve the "right" preference problems, and share this information appropriately.

Finally, *preference formation-independence* is a strong requirement. By relaxing this property slightly, and allowing the mechanism center to have more information about the deliberative processes of the agents, it may be possible to design mechanisms which are deliberation-proof and non-misleading. An appealing idea is to allow the mechanism to have information about the performance pro£les and cost functions of the agents. From its global perspective, the mechanism could guide the agents in their deliberation-control policies. A promising approach is to combine a mecha-

nism with search techniques from the Operations Research literature (such as [16]) so that the mechanism can sequentially query agents to encourage them to deliberate on their own problems. This approach has been used in a simple information gathering setting where agents have the possibility of gathering information only about their own preferences in a single step [4]. Generalizing the approach so that strategic-deliberation is avoided raises some interesting issues since incentives need to be engineered so that agents follow the recommendations of the mechanism, and truthfully reveal their preferences. Another interesting problem is determining what information agents must reveal to the mechanism. We believe that agents should only be required to reveal some small amount of details about their deliberative capabilities and technologies. Determining the minimal amount of information agents must reveal to the mechanism in order to avoid strategic-deliberation and misleading behavior is another interesting problem.

6. CONCLUSION

Agents often do not know their preferences *a priori*. Instead, they must actively and deliberatively execute a computational or information gathering process in order to determine them. This places an added burden on the agents participating in mechanisms since they are faced with the problem of how to determine their preferences, and then how to reveal them to the mechanism. These two aspects are closely interrelated. An agent's preferences in¤uence what it will reveal to a mechanism. Similarly, what an agent plans to reveal will in¤uence how it decides to determine its preferences.

In this paper we proposed designing mechanisms explicitly for deliberative agents. We laid out a set of properties which we believe mechanisms for deliberative-agents should exhibit. We proposed that a mechanism should be *preference formation-independent* in that the mechanism should not be required to know how the agents are determining their preferences. We also proposed that a mechanism should be *deliberation-proof* in that no agent should have incentive to actively determine the (partial) preferences of other agents. Finally, we claimed that a mechanism should be *non-misleading* in that no agent should be given incentive to take actions that purposefully mislead others. We show that in interesting mechanisms, it is impossible to obtain all three properties together.

While the results in this paper are negative in that they show that we are unable to design mechanisms with the properties we desire, it does open up a range of interesting research questions. In particular, it may be possible to weaken one of the properties slightly in order to achieve the other two. Deciding which property should be relaxed, and by how much are problems we believe are worthwhile pursuing.

7. REFERENCES

[1] Dirk Bergemann and Juuso Välimäki. Information acquisition and ef£cient mechanism design. *Econometrica*, 70:1007–1034, 2002.

[2] Olivier Compte and Philippe Jehiel. Auctions and information acquisition: Sealed-bid or dynamic formats?, 2001. Working Paper.

[3] Vincent Conitzer and Tuomas Sandholm. Computational criticisms of the revelation principle. In *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*, New York, NY, 2004. Short paper. Full-length version appeared in the AAMAS-03 workshop on Agent-Mediated Electronic Commerce (AMEC).

[4] Jacques Cremer, Yossi Spiegel, and Charles Zheng. Optimal selling mechanisms with costly information acquisition. Working paper, August 2003.

[5] T. Ito, M. Yokoo, and S. Matsubara. Towards a combinatorial auction protocol among experts and amateurs: The case of single-skilled experts. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'03)*, pages 481–488, Melbourne, Australia, 2003.

[6] Noa K£r-Dahav, Dov Monderer, and Moshe Tennenholtz. Mechanism design for resource bounded agents. In *Proceedings of the Fourth International Conference on Multi-Agent Systems (ICMAS)*, pages 309–315, Boston, MA, 2000.

[7] Kate Larson and Tuomas Sandholm. Bargaining with limited computation: Deliberation equilibrium. *Arti£cial Intelligence*, 132(2):183–217, 2001. Short early version appeared in the Proceedings of the National Conference on Arti£cial Intelligence (AAAI), pp. 48–55, Austin, TX, 2000.

[8] Kate Larson and Tuomas Sandholm. Costly valuation computation in auctions. In *Theoretical Aspects of Rationality and Knowledge (TARK VIII)*, pages 169–182, Siena, Italy, July 2001.

[9] Kate Larson and Tuomas Sandholm. Experiments on deliberation equilibria in auctions. In *International Conference on Autonomous Agents and Multi-Agent Systems*, New York, NY, USA, July 2004.

[10] Daniel Lehmann, Lidian Ita O'Callaghan, and Yoav Shoham. Truth revelation in rapid, approximately ef£cient combinatorial auctions. *Journal of the ACM*, 49(5):577–602, 2002. Early version appeared in ACMEC-99.

[11] Andreu Mas-Colell, Michael Whinston, and Jerry R. Green. *Microeconomic Theory*. Oxford University Press, 1995.

[12] Noam Nisan and Amir Ronen. Computationally feasible VCG mechanisms. In *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*, pages 242–252, Minneapolis, MN, 2000.

[13] David C. Parkes. Optimal auction design for agents with hard valuation problems. In *Agent-Mediated Electronic Commerce Workshop at the International Joint Conference on Arti£cial Intelligence*, Stockholm, Sweden, 1999.

[14] Nicola Perisco. Information acquisition in auctions. *Econometrica*, 68(1):135–148, January 2000.

[15] Tuomas Sandholm. Issues in computational Vickrey auctions. *International Journal of Electronic Commerce*, 4(3):107–129, 2000. Special Issue on Applying Intelligent Agents for Electronic Commerce. A short, early version appeared at the Second International Conference on Multi–Agent Systems (ICMAS), pages 299–306, 1996.

[16] Martin Weitzman. Optimal search for the best alternative. *Econometrica*, 47(3):641–654, 1979.