Mechanism Design for Multi-Agent Meeting Scheduling

Elisabeth Crawford and Manuela Veloso
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, 15213
{ehc,mmv}@cs.cmu.edu

Abstract

In this paper we examine the benefits and limitations of mechanism design as it applies to multi-agent meeting scheduling. We look at the problem of scheduling multiple meetings between various groups of agents that arise over time. All the agents have private information regarding their time preferences for meetings. Our aim is to elicit this information and assign the meetings to times in a way that maximizes social welfare.

We discuss problems with previous attempts to design incentive compatible (IC) and individually rational (IR) mechanisms for meeting scheduling. We show how requesting agent preferences for entire schedules helps to eliminate IC problems. We focus, in particular, on the problem of determining when agents are available for meetings. We show that our choice of IC and IR mechanisms is quite restricted when we allow agents to declare their availability.

Keywords: Multi-Agent Systems, Mechanism Design, Meeting Scheduling

1. Introduction

For many people, scheduling meetings is a daily and time consuming task. Finding satisfactory times for new meetings can be very difficult, particularly when the meetings have many participants or existing meetings need to be moved. As such, a number of researchers e.g [23, 14, 7] and more recently [1, 17] have looked into designing intelligent software agents that can automate the process. There are two major challenges that are yet to be fully overcome: (i) the design of accurate algorithms to learn a user' scheduling preferences and (ii) the design of effective methods for agents to agree upon meeting times. In this paper we focus on the second problem and assume we are dealing with software agents that have an accurate model of their users' preferences. In particular, we investigate the feasibility of using mechanism design to solve (ii).

The multi-agent meeting scheduling problem (MAMS) consists of a set of agents A, a set of sequential time slots T, and a set of meetings M that are arise over time. Each agent in A has preferences about how the meetings in M are scheduled. Given a set of meetings M_i let τ_{M_i} denote all the possible ways in which these meetings can be arranged over T. We call agent $i's$ true preference function for a set of meetings M_i, $\theta_{i\tau}$, where the domain of $\theta_{i\tau}$ is τ_{M_i} and the range is a value representing the utility of this meeting arrangement to the agent.

To solve MAMS we need to find an assignment of meetings to times. An assignment t for a meeting m is valid if each agent participating in the meeting has no other meeting scheduled for time t. A solution to the MAMS problem consists of a valid assignment of a subset of the requested meetings M to time slots in T. In general we would like to find a valid assignment for all $m \in M$. If we consider T to be infinite, this is always possible, however, in this paper we will only consider finite T. Clearly there can be good and bad solutions to MAMS. When designing a system to solve the problem it is necessary to decide what metric, if any, is to be maximized. For instance, we may wish to maximize the number of meetings assigned or the sum of the agents' utilities.

In this paper, we focus on mechanism design approaches to MAMS. We investigate whether mechanism design can be applied to MAMS to find the schedules with the highest sum of agent utilities by eliciting each agent's true preferences $\theta_{i\tau}$.

This paper is organized into 6 sections. In Section 2, we introduce some concepts from mechanism design, including Incentive Compatibility (IC) and Individual Rationality (IR). We also discuss the main alternative approaches to solving MAMS namely multi-agent negotiation and distributed constraint satisfaction. In Section 3 we outline IC problems with previous mechanism design approaches to MAMS. In Section 4 we concentrate on the problem of providing agents with an incentive to truthfully reveal their availability. We show that when agents are asked to reveal their availability we are quite restricted in our choice of mechanisms if we desire IC and IR. In Section 5, we show that the effect availability declarations have on the meeting times chosen can be reduced by having agents quantify the value of other agents to the meetings. Finally in Section 6, we discuss our findings and suggest some avenues for future research.

2. Background

2.1. Mechanism Design

Mechanism design is concerned with defining games. The participants of the games are assumed to act with *self-interest* according to their preferences about the outcome [20]. Given this assumption it is the aim of the mechanism designer to choose actions and pay-offs such that the game will have the outcome he/she desires.

Mechanism design is commonly used in auctions. For instance, suppose we want to sell a good to the person that values it the most. If we collect private bids and simply sell the good to the highest bidder (for the price of his bid) it will not be in the best interests of the bidders to bid their true valuations. Rather, they will have an incentive to bid lower and we won't be able to guarantee that the good will go to the participant with the highest valuation. However, by using a *second price auction*, where the highest bidder pays the second highest bid for the good, we can ensure that it is in every bidder's best interest to bid his true valuation. The second price auction is a classic example of how mechanism design can be used to achieve a desired outcome.

The outcome desired by a mechanism designer depends largely on the specific application. However, the *social welfare maximizing outcome* is a common choice. The social welfare maximizing outcome is the outcome that maximizes the sum of the agents' utilities. In general, to compute the social welfare maximizing outcome we need to know each agent's utility for every possible outcome. These values are known only to each agent. Suppose an agent reveals incorrect values, then the outcome we compute is not guaranteed to maximize social welfare. As such, it is the job of the mechanism to get the agents to state these values truthfully, as in the second price auction.

Game Theory tells us that whenever a self-interested, rational agent has a strategy that dominates all others, he will play that strategy [8]. Thus, we want to make it every agent's dominant strategy to reveal his true utilities. We can think of a strategy as a rule for deciding which action to take. A *dominant strategy* is a strategy that is a best response to all possible strategies of the other agents. In other words, an agent has a dominant strategy if he is best off playing a particular strategy no matter what strategy the other agents play.

Consider the matrix game in Table 1. There are two players, the row player and the column player. The row player chooses either the top row or bottom row of the matrix and the column player chooses either the left or right column. Each square contains a pair of numbers, the first number is the pay-off to the row player if that combination of strategies is chosen, and the second is the pay-off to the column player. In our example if the row player plays U and the column player plays L the row player receives a pay-off of 5 and the column player a pay-off of 2. Notice that the row player is always better off playing U than D, since for both of the column player's strategies U is better than D. We say that U is the dominant strategy for the row player.

	L	R
U	5,2	6,5
D	1,4	3,5

Table 1. Example Game

Two common desiderata for mechanisms are incentive compatibility and individual rationality. A mechanism is *incentive compatible* (IC) if it is every agent's dominant strategy to reveal his utility values (i.e his valuations) truthfully. A mechanism is *individually rational* (IR) if agents cannot receive a negative pay-off from the mechanism i.e., they are never worse off for having participated. IC mechanisms normally require some agents to make payments to the system and the system sometimes also makes payments to the agents.

The Clarke Tax mechanism [3] is a well known mechanism that generally achieves both IC and IR. As the name suggests this mechanism requires the agents to make payments to the system. Below we give an example of how one might consider scheduling a meeting using the Clarke Tax mechanism.

Suppose we have some centralized mechanism that requests agents to bid for meeting times by stating the dollar value they place on the meeting being assigned to that time. The *Clarke Tax* mechanism [3] works by charging the agents according to the amount they influence the chosen meeting time. Under this scheme, each agent i states his money value for each possible outcome. The social welfare maximizing outcome x is then selected. Agent i pays an amount equal to the sum of the other agents' utilities for the outcome that would have been chosen by the mechanism had he not participated, minus the sum of the other agents' utilities for x .

Consider the utilities as reported in Table 2. The outcome chosen by the Clarke Tax mechanism is 9am since the total bids for this time (16) are greater than the total for 10am (14). Given this outcome we look at the tax for each agent. To calculate Agent 1's tax, we observe that the outcome that would have been chosen had Agent 1 not been involved is 10am. 10am has a value of 10 to the other two agents. The value these two agents have for the time chosen when Agent 1 is present (9am) is only 7. Thus, we require Agent 1 to

	9am	10am	Tax
Agent1	9	4	3
Agent2	4	6	0
Agent3	3	4	0
Total	16	14	

Table 2. Example of Clarke Tax calculation

pay the difference i.e., 3. The tax for Agent 2 and Agent 3 is computed similarly. Note that the removal of Agent 2 or Agent 3 will see 9am remain the chosen time. As such, their presence does not pivot the outcome and hence they pay no tax.

To see that the Clarke Tax Mechanism is IC, consider what happens if Agent 1 overbids for 9am and gives it a value of 10 instead of 9. 9am will still be chosen for the meeting. Since the tax Agent 1 pays does not depend on the value of his bid he still pays the same tax of 3. Now consider what happens if Agent 1 underbids for 9am. Suppose he bids 8 instead of 9. The time chosen by the mechanism will still be 9am and Agent 1 will pay the same tax as before. Hence underbidding has not gained Agent 1 anything. Suppose Agent 1 underbids further with a bid of 6 instead of 9. The time chosen by the mechanism will now be 10am and Agent 1 will receive a utility of 4 and pay no tax. However if Agent 1 had bid truthfully, he would have received utility 9 and paid a tax of 3 leading to a pay-off of 6. So Agent 1 would have been better off revealing his true preferences. For a formal introduction to the Clarke Tax mechanism see [20].

In real world MAMS mechanisms we cannot use money to decide on meeting times. Instead, we need to replace money with points as done by Ephrati et al. [7]. The system can distribute points to the agents periodically which they can use to bid for meeting times. These points take on some of the qualities of money as they are needed repeatedly by users to influence the scheduling of new meetings. Virtual money has also been used in other multi-agent and multi-robot systems e.g. [6].

2.2. Related work

Much of the previous work on the MAMS problem has looked at using negotiation between agents to find a solution e.g. [14, 23]. The following, is a simple example of the kind of negotiation protocol looked at in previous work:

- The host announces the meeting
- While the host has not found an intersection in times:
    - The host proposes some times to the attendees
    - The attendees propose some times to the host

These negotiation approaches assume that agents are to some extent truthful about their availability and utilities for times and/or are willing to compromise their preferences.

We can say an agent is truthful in a protocol, like the one described, if he offers all the times he is available for eventually, and offers times in the order of his preference. Truthfulness however, is certainly not a dominant strategy. Suppose all the agents but Agent i are truthful. Then Agent i may be able to gain by just proposing his favorite times and hoping the other agents eventually agree. This lack of IC could result in problems. For instance, an intersection of available times may not be found, despite one actually existing or the process may iterate for a very long time.

Different approaches have been taken to handling user preferences but one method that has been combined with negotiation is that described in [24]. Sen et al. [24] show how voting methods can be used (within a software agent) to evaluate different options for scheduling a meeting when users have multi-faceted and possibly competing preferences. This preference model is used to rank proposed meeting times and to decide which times are acceptable.

Garrido and Sycara [9] and Jennings and Jackson [14] take the approach of allowing agents to not only propose meeting times but also to quantify their preferences for proposals. The agent that is collecting the proposals then makes decisions about meeting times based on the reported utilities of all the meeting participants. This style of approach involves a lot of trust, since for the procedure to work well all the agents must report their preferences truthfully, and truthfulness is not a dominant strategy.

In [25] Shintani et al. propose a persuasion based negotiation approach for MAMS. The persuasion mechanism involves compromising agents adjusting their preferences so that their most preferred times are the persuading agent's most preferred times. Similarly to the other negotiation approaches this method relies on the cooperation of the agents and cannot guarantee outcomes that maximize the sum of the agents' utilities.

Distributed Constraint Reasoning (DCR) approaches have also been applied to MAMS. For example Modi and Veloso [16] model the meeting scheduling problem according to the DCR paradigm and evaluate strategies for making *bumping decisions*. The way in which agent's decide when to *bump* (i.e., move an existing meeting to accommodate a new meeting) can have implications for the efficiency of the meeting scheduling process. A meeting that is bumped must be rescheduled and it is possible that in rescheduling a bumped meeting more meetings will be bumped. Intuitively, if the agents want the scheduling process to finish quickly, they should try to bump meetings that will be easy to reschedule. Similarly to the negotiation approaches discussed, DCR approaches assume the agents are cooperative.

Recently, we have been looking at how agents can negotiate strategically in order to better satisfy their users' preferences [5]. In [5] we propose an approach based on the idea of adapting a playbook of strategies [2]. We show how agents can learn to choose strategies based on who they are negotiating with in order to achieve a better outcome for their users. This kind of approach could be used to negotiate about a variety of matters in which agents have self-interest.

The key issue with most of the DCR and negotiation-based approaches discussed is the degree to which they rely on the willingness of agents to compromise their preferences and cooperate. Certainly in a MAMS system it seems sensible to assume agents are in part cooperative. But most of these approaches largely ignore the possibility of agents behaving strategically and may not optimize global utility. Work on mechanism design for MAMS is motivated by the possibility of being able to optimize global utility even if some, or all, of the agents are strategic.

3. Issues with existing mechanism design approaches to MAMS

A mechanism design approach to MAMS was first taken by Ephrati, Zlotkin and Rosenschein [7]. Ephrati et al., used a *Clarke Tax* mechanism to try and ensure IC. Three different approaches were taken to implementing the mechanism: meeting-oriented; calendar-oriented and schedule-oriented. The authors state that each of these approaches is IC. However, their proof did not consider the *repeated* application of the Clarke Tax mechanism, rather it looked at a single step. We show, by example, that these approaches are not IC when we consider the repeated application of the mechanism and we observe that achieving IC is not a trivial problem in MAMS.

3.1. Meeting-Oriented Approach

In the meeting-oriented approach proposed by [7] when each new meeting enters the system all the agents are given more points with which to bid. The agents attending the meeting then bid over all time slots in the time horizon. The social choice is calculated by the mechanism, the tax applied and the meeting recorded in the system's copy of each agent's calendar. The time the meeting is scheduled at, is not considered available when scheduling future meetings for the agents concerned. When the agents have combinatorial preferences over meetings and times this approach is not IC. Consider the following example. Agent 1 has a low utility for meetings on Tuesday, but wants to complete some work with some other agents by Wednesday that will require two meetings over two days. Now, if Agent 1 thinks the other agents want one of the meetings scheduled on Wednesday, it may be in Agent 1's best interest to overbid for time slots on Tuesday. This is because Agent 1 gets a high utility from both meetings happening by Wednesday.

	Tue	Wed	Thu	Fri
Agent1	2	4	3	2
Agent2	3	5	4	4
Agent3	3	6	3	3
Total Welfare	8	15	10	9

Table 3. The Agents' true preferences

Suppose the agents' true preferences over the days without thinking ahead are those in Table 3. Let Agent 1's utility from the first meeting being scheduled on Tuesday *and* the second on Wednesday be 20 — which is more than his preference for those days on their own. Now, if we assume that the other two agents will not change their preferences when the new meeting enters the system, the first meeting will be scheduled on Wednesday and the second on Thursday. This will have a utility of 7 for Agent 1, and he will be charged 0 for the first meeting and 0 for the second. Now suppose Agent 1 changes his bid for Tuesday to 10. The first meeting will be scheduled on Tuesday and Agent 1 will be charged 5. The second meeting, will be scheduled on Wednesday, and Agent 1 will again pay 0. So in total, Agent 1 will get a pay-off of 15 which is higher than the 7 points for bidding truthfully. Hence we have shown by example that the meeting-oriented approach proposed in [7] is not IC when agents have combinatorial preferences over meetings and times.

Since new meetings arrive over time there may be benefits to viewing the meeting scheduling problem as an online mechanism design problem. There has been some progress recently on designing online mechanisms e.g.[21, 18, 15, 13], but we are aware of no existing research in this area that solves the meeting scheduling problem. The meeting scheduling problem is more complex than the models looked at so far since there are multiple distinct objects (times) that bids are placed on and bidders have combinatorial preferences over these objects.

3.2. Calendar-Oriented Approach

In the *calendar-oriented* approach proposed in [7] the agents express their preferences over a *complete time period once only*. As time passes, new times are added to the period under consideration and the agents are asked for their preferences. This approach does not allow agents to express preferences that are *meeting dependent* or preferences that depend on the calendar's current configuration. As such, this approach is not IC.

Consider the following scenario. Agent 1 has a high utility for back-to-back meetings — he gets 5 extra utility points when any two meetings are scheduled back-to-back. Suppose Agent 1 is considering how to allocate his preferences for 1,2,3 and 4pm on Monday. Agent 1 knows from experience that 2pm is a very popular time with many of the people he regularly needs to meet with, and as such he expects a meeting will be scheduled then. Agent 1 has a low utility for meeting at 1pm, but since he expects a meeting to be scheduled at 2pm perhaps he should overate his preference for 1pm? Table 4 contains Agents 1's and Agent 2's true preferences over meeting times on Monday afternoon. If we schedule two meetings with these preferences the first will be scheduled at 2pm and the second at 4pm and Agent 1 will have a pay-off of 5 (6 minus a tax of 1 for the first meeting and a tax of 0 for the second). But, Agent 1's back-to-back preference means the declaration in Table 5 will lead to a higher pay-off.

	1pm	2pm	3pm	4pm
Agent1	3	4	2	2
Agent2	2	3	2	4
Total	5	7	4	6

Table 4. True time preferences for Monday afternoon

	1pm	2pm	3pm	4pm
Agent1	5	4	2	2
Agent2	2	3	2	4
Total	7	7	4	6

Table 5. Strategic declarations for Monday afternoon

If Agent 1 reveals the preferences in Table 5 he will receive 12 points ($3+4+5$) in utility and pay a tax of 2 for the first meeting and a tax of 1 for the second resulting in a final pay-off of 9 points. Hence we have shown by example that the calendar-oriented approach proposed in [7] is not IC when agents have preferences that depend on the configuration of their calendar (e.g. the preference for having meetings back-to-back).

3.3. Schedule-Oriented Approach

The problems that have arisen in the approaches previously described have been caused by a lack of expressiveness in the bidding process. The lack of expressiveness has resulted in truthfulness not being a dominant strategy and hence the mechanisms are not IC. This is similar to the need for combinatorial auctions when the goods are complimentary. The final approach proposed in [7] allows the agents to bid over all possible schedules. The schedules are generated by taking every possible combination of meetings and time slots up to some time horizon. As each new meeting enters the system the agents are allowed to bid over all feasible schedule configurations. This approach has more promise in terms of IC, but some issues remain.

Suppose the time horizon in this method is two weeks. Further, suppose that the time a particular meeting is scheduled next month affects some agent's preference for when a meeting is scheduled this month. This causes the same IC problem we saw for the meeting-oriented example. Thus, this mechanism is not IC either. The longer the time horizon the less likely this problem is to occur. As the time horizon increases the number of possible schedules grows exponentially leading to a trade-off between IC and complexity. We note that the schedule-oriented approach leads to many possibilities for which the agents must submit preferences. Clearly this will require agents that have knowledge of their users' preferences, since users cannot be asked to assess large numbers of options. Further we cannot allow there to be so many options that the problem becomes intractable. The mechanism would no longer be IC if the agents or the mechanism had to make approximations [19].

4. Towards an effective solution for MAMS

In this section we consider in more detail a variant of the most promising approach proposed in [7] — the schedule-oriented approach. We show that even if we ignore time horizon issues, this mechanism has IC problems. We discuss how this can be solved. We then consider the problem of incorporating the availability of agents for particular times into a mechanism.

4.1. A schedule-oriented mechanism

The mechanism shown in Table 6 is a variation of the schedule-oriented mechanism described in [7]. The mechanism in [7] has a time frame parameter that specifies the number of time slots that are to be frozen as time progresses. By frozen we mean that any meeting scheduled cannot be changed. Clearly, if we allow time frames longer than one time slot, the agents cannot fully express their preferences and we will have IC problems. Thus, in our variation we do

Input: length of time slots, time horizon

procedure MAMS1:

1. Divide calendar into time slots $t_1, t_2, ...t_h$ of length m
2. *current-time* $= t_1$
3. *horizon-time* $= t_h$
4. $\forall$ agents i, $budget_i = x$
5. When agent requests meeting, do
6. $\forall$ agents i, $budget_i = budget_i + x_{new}$
7. notify all agents of meeting
8. notify agents if their presence is requested
9. generate all possible combinations of meetings and times from current time to horizon time, $schedule_1, schedule_2, ...schedule_k$
10. $\forall$ agents i, request valuation for each schedule
11. check $\forall$ agents i and $\forall$ schedules j:
12. $value_i(schedule_j) <= budget_i$
13. select schedule that maximizes social welfare
14. report schedule to agents
15. $\forall$ agents i, $budget_i = budget_i - clarketax(i)$
16. when m units of time have passed
17. current time = next time slot
18. create new time slot at horizon, and reassign horizon time

Table 6. Variant of the schedule-oriented approach

not refer to a time frame. In fact, a meeting could be moved right up until the *current-time* if this would maximize social welfare. Unfortunately, despite the fact that allowing agents to fully express their preferences within some time horizon has greatly reduced IC problems, at least one issue remains.

4.1.1. Further IC issues with the schedule-oriented approach The following example exposes a further IC problem with the schedule-oriented approach. Suppose our time horizon in Table 6 is a week in advance. It is Monday, there is some meeting that will most likely be scheduled on Friday that Agent 1 wants to attend. Agent 1 has a strong preference for the meeting to be at 2pm. Now suppose Agent 1 expects 5 new meetings will enter the system between now and Friday. When each meeting enters, Agent 1 will have to bid over the possible schedules. The question, is whether or not Agent 1 should truthfully reveal, that he greatly prefers schedules where the meeting is held at 2pm. If Agent 1 reveals this preference when each new meeting needs to be scheduled, his preferences could strongly influence the outcome (if they are not also the preferences of the others). Thus, it is possible he may have to pay a high tax each time. Instead of paying the tax each time, Agent 1 would be better off pretending he does not have a strong preference for the time of the meeting until the 5th meeting enters the system. At this point he can declare his preference and thus only risk having to pay a high tax once. As such, it is not a dominant strategy for Agent 1 to reveal his true preferences at each iteration and hence the mechanism is not IC. The complicated nature of this example highlights that designing an IC mechanism for this problem is non-trivial.

4.1.2. A fix We show a way the problem described in the previous section can be removed. Suppose we fix a schedule for some time period in one round of bidding. This eliminates the ability of meetings to be moved, and thus the problem described in the previous section. However, since we also need agents to be able to fully express their preferences over the time period, once the schedule has been fixed no new meeting can be added into that time period. Consider the algorithm (MAMS2) in Table 7.

The difficulty with the mechanism in Table 7 is that the agents would have to think ahead about the meetings they want scheduled. The shorter the time period chosen, e.g 3 days, a week, a fortnight etc the greater the time horizon IC problem but the smaller the problem of having to think ahead. Clearly however any very last minute meetings are likely to be a problem.

4.2. Availability

An important missing element from the approaches discussed thus far, is a way of handling the unavailability of agents. The systems proposed in [7], did not allow agents

Input: period-length, slot-length

procedure MAMS2:
1. Divide the calendar up into time periods $p_1, p_2, ...p_n$
2. Divide the time periods up into slots $t_1, t_2, ...t_k$
3. for current-period in $p_1, p_2, ...p_n$
4. while current-period not expired:
5. append requests for meetings in $p_{current+x}$ $x > 0$ to lists
6. $\forall$ agents i, $budget_i += $ *no-meetings*$*x_{new}$
7. notify agents of all meetings
8. notify agents of which meetings their presence is requested at
9. steps 9-15 of MAMS1

Table 7. Fixed schedule-oriented approach to MAMS

to indicate they are unavailable for certain times. In the calendar-oriented and meeting-oriented approaches from [7], if an agent had a meeting scheduled by the system for a particular time, the agent was considered unavailable at that time. This was the only way agents could be considered unavailable. However, for a meeting scheduling system to achieve efficient results we need agents to be able to declare their unavailability. Users may have important commitments such as appointments with medical specialists that cannot be moved. Furthermore, there can also be implications for the efficiency of the organization the user works for. Suppose for instance, that a user has a very important client presentation. It may be in both the user's and the company's interest if the user has no meeting immediately prior to the presentation so he can complete his preparations. Any practically implementable MAMS system must have a way of handling availability — preferably by a means that does not involve human intervention. As such, we would like a mechanism for MAMS that gets agents to reveal their availability as well as their time preferences.

We say that when an agent declares he is unavailable for a time, that the agent *vetoes* or *blocks* that time. In other words, the agent's declaration means that no meeting of which he is a participant can be held at that time. Simply asking agents for the times they are unavailable is not an IC approach. If we used this scheme there would be a clear incentive for the agents to say they are only available at the times for which they have the highest utility. Even more so than with preferences, we need agents to report their availability truthfully.

4.3. IC Mechanisms for MAMS with Availability Declarations

In this section we look at how the power of veto restricts our choice of IC mechanisms, and hence how it restricts the mechanisms we can use to solve MAMS.

4.3.1. Some Definitions Recall that a mechanism is *individually rational* (IR) if every agent is never worse off from participating in the mechanism. IR is a very desirable property in mechanism design since agents do not have to reason about whether or not they should take part.

When applying mechanism design to MAMS we are assuming agents have *quasi-linear* preferences in order to circumvent the *Gibbard-Satterthwaite Impossibility Theorem* [10] [22]. Let $x \in X$ be the possible outcomes of the mechanism. Let p_i be the payment agent i is required to make to the system. Now we let $u_i(x)$ be agent $i's$ utility from the whole process if the mechanism selects x, and let $v_i(x)$ be $i's$ valuation for that outcome.

We say that agent i has *quasi-linear* preferences if $u_i(x) = v_i(x) - p_i$, i.e., $i's$ utility is simply his valuation for the chosen outcome, minus the payment he must make. The assumption of quasi-linear preferences is considered reasonable in markets and it is no less reasonable in the context of meeting scheduling. Despite the restriction of quasi-linear preferences, we allow the agents to have general valuation functions.

The Groves family of mechanisms were developed in [12], [3] and [26] for the domain of quasi-linear preferences. Let $\theta_i^r \in \theta^r$ be the reported valuation function of each agent over X. The Groves mechanism selects the outcome, x^* as,

$$x^*(\theta^r) = argmax_{x \in X} \sum_{\forall i} v_i(x)$$

In other words, Groves mechanisms select the outcome that maximizes social welfare.

The payment rule of the groves mechanism is,

$$p_i(\theta^r) = h_i(\theta_{-i}^r) - \sum_{j \neq i} v_j(x^*)$$

where $h_i : \theta^r_{-i} \rightarrow \Re$ is an arbitrary function on the reported valuations of every agent except i.

The Groves mechanism are unique in the domain of quasi-linear preferences in that they are the only direct mechanisms that are allocatively efficient, IC and implementable in dominant strategies for general valuation functions [11]. By the Revelation Principle this result extends to general mechanisms[10].

Lemma 4.1 [1] *There exists no Groves mechanism for MAMS with availability declarations that maximizes social welfare and simultaneously satisfies IC and IR without making payments to agents when agents have general valuation functions and the structure of $h_i(\theta^r_{-i})$ is restricted as follows: $h_i(\theta^r_{-i}) = \sum_{i \neq j} v_j(?)$ where $v_j(?)$ is the reported value of agent j for one of the possible outcomes.*

Proof Consider a function $h_i(\theta^r_{-i})$ that potentially satisfies these properties. Since the mechanism must make no payments we have the following requirement:

$$h_i(\theta^r_{-i}) - \sum_{i \neq j} v_j(x^*) \geq 0$$

This implies $h_i(\theta^r_{-i})$ must satisfy the following inequality:

$$\sum_{i \neq j} v_j(x^*) \leq h_i(\theta^r_{-i}) \tag{1}$$

Now since the mechanism must also be IR we also require that:

$$h_i(\theta^r_{-i}) - \sum_{i \neq j} v_j(x^*) \leq v_i(x^*)$$

This implies the following further restriction on $h_i(\theta^r_{-i})$:

$$h_i(\theta^r_{-i}) \leq \sum_{i \neq j} v_j(x^*) + v_i(x^*) \tag{2}$$

Now to ensure requirement 1 we must base $h_i(\theta^r_{-i})$ on the preferences of the other agents for outcomes which they find preferable either individually (i.e., every agent's contribution to h is higher than to $\sum_{i \neq j} v_j(x^*)$) or in total (i.e., the sum of the agent's contributions to h is higher than $\sum_{i \neq j} v_j(x^*)$ even though some individual contributions may be lower). Note that we cannot simply let $h_i(\theta^r_{-i})$ be the function $\sum_{i \neq j} v_j(x^*)$ since no one would ever pay tax or receive payments and hence the mechanism could not possibly be IC.

We now consider requirement 2. When agents cannot veto options we normally ensure this property holds by reference to the fact that x^* is the social choice. Since without the power of veto, for any possible outcome y not chosen by the mechanism we know:

$$\sum_{i \neq j} v_j(y) \leq \sum_{i \neq j} v_j(x^*) + v_i(x^*)$$

or y would have been the social choice. However this does not necessarily hold when i has the ability to veto outcomes. We do know that it holds for any y *not* vetoed by i, but note that: (i) i may veto all but the outcome chosen and (ii) if we exclude vetoed outcomes we are letting i influence h and we loose the IC property. Furthermore there is no way to choose contributions to h from different outcomes for different agents such that we can both guarantee that 2 holds and that $h_i(\theta^r_{-i}) \geq \sum_{i \neq j} v_j(x^*)$ without reference to the type of i. The remaining possibility to ensure 1 and 2 without direct reference to the type of i is to set $h_i(\theta^r_{-i})$ to $\sum_{i \neq j} v_j(x^*) + 1$, since 1 is the smallest possible value i could have for x^*. However, this amounts to always applying a tax of 1 which clearly cannot satisfy IC.

As such there is not way to choose a function $h_i(\theta^r_{-i})$ of the form described such that $h_i(\theta^r_{-i})$ satisfies all the properties.

Theorem 4.2 *There is no mechanism for MAMS with the properties stated in the lemma for agents with general valuation functions.*

Proof This result follows directly from the uniqueness of Groves mechanisms and from the revelation principle.

4.3.2. The restriction on $h_i(\theta^r_{-i})$. In the lemma we restricted $h_i(\theta^r_{-i})$ to being $\sum_{i \neq j} v_j(?)$ where $v_j(?)$ is the reported value of agent j for one of the possible outcomes. There is no restriction placed on the form of $h_i(\theta^r_{-i})$ in the definition of the Groves mechanism. As such $h_i(\theta^r_{-i})$ could be a very general equation involving linear and non-linear terms and various constants. The Lemma would be strengthened if we could prove it for general h. We think this is probably possible, but finding a proof or alternatively a counter example remains open.

We note that the result described here is not only applicable to the MAMS problem. It applies more generally to situations where it is necessary to allow agents to veto some outcomes.

4.4. Mechanism for MAMS that Makes Payments

If we drop the requirement that the mechanism can only receive payments we can achieve an IC and IR mechanism when agents have the power to veto outcomes. By setting $h_i(\theta^r_{-i})$ to the value the other agents place on the worst possible outcome without Agent i we can ensure IR. The payment rule becomes:

$$p_i(\theta^r) = \sum_{j \neq i} v_j(x^w_{-i}) - \sum_{j \neq i} v_j(x^*)$$

1 The lemma in [4] was misstated. It should have included the condition that the mechanism does not make payments as well as the restriction on h

where $v_j(x^w_{-i})$ is the value of Agent j for the outcome with the lowest sum when agent i's values are excluded. Note that $\sum_{j \neq i} v_j(x^w_{-i}) \leq \sum_{j \neq i} v_j(x^*)$ — either the lowest valued outcome of the other agents is x^* or it is some other outcome with even lower value. As such, this payment rule is always IR since the mechanism will never actually tax. Furthermore since $h_i(\theta^r_{-i})$ depends only on the reported types of the other agents, the Groves Mechanism with this payment rule is IC.

In situations where money is being used such a payment rule would often not be feasible - since the mechanism would pay out money but never receive any. However, because we are using points as opposed to real money in MAMS, a mechanism that just makes payments is more realistic than if money were involved. However, if we have a mechanism that only ever makes payments, then the budget of every agent is almost certain to grow over time. Clearly inflation is likely to be very severe and agents are going to have to adjust their bidding as time progresses to take this into account. Inflationary problems could be reduced by using schemes that reduce the points supply but maintain relative purchasing power amongst the agents. For instance, a scheme of periodically halving all the agents' budgets could be used to stop the amount of points in the system becoming very large.

4.5. Clarke Tax Mechanism for Meeting Scheduling Without IR

In this section we look at what can happen if we ignore the IR requirement and simply use a Clarke Tax mechanism to encourage truthful availability and preference declarations. Table 8 shows example tax calculations for three agents under this mechanism.

	sch_1	sch_2	sch_3	sch_4	sch_5	Tax
Agent1	1	4	0	0	0	11
Agent2	5	3	4	9	4	0
Agent3	2	4	9	9	3	1
Total	8	11	13	18	7	

Table 8. True preferences, including availability

In the table a value of 0 indicates that the schedule is infeasible for that agent. When choosing the social welfare maximizing outcomes these schedules are blocked off. Thus, sch_2 with value 11 will be chosen. Now consider the payment Agent 1 has to make. If Agent 1 was not involved in the process, sch_4 would have been the social choice with value to the other players of 18. Without Agent 1, the value for sch_2 is only 7. To compute Agent 1's tax we minus 7 from 18 giving a tax of 11 as displayed in the table. Agent 1 is paying a high amount because his unavailability is greatly effecting the outcome. Notice that the amount Agent 1 pays is significantly higher than his value for the chosen schedule. Thus, it was clearly not in Agent 1's interest to participate in the mechanism in this round. If we continue to schedule meetings and Agent 1's unavailability continues to cause him to pay an amount of tax that is not IR he may end up with no points to express positive preferences with. We cannot limit the amount an agent spends to stop his points from becoming negative, because even if he only places bids of 1 on times he is available for, his unavailability on its own can cause a large tax. This example demonstrates how the mechanism's failure to ensure IR can reduce its effectiveness.

5. Reducing Veto Power

In many instances when there are more than two meeting participants, one person's unavailability at a particular time does not completely block off that time. If the time is very particularly favored by the other participants, or if the unavailable participant is not very important for the meeting, the meeting may still be scheduled. In this section we look at incorporating this weaker notion of veto power into a mechanism for MAMS. In particular we look at the affect it has on the IR problem in the Clarke Tax mechanism.

We propose a mechanism where not only do agents express preferences over schedules, but also over attendees. The agents specify their preferences for schedules and the amount their utility for a schedule is reduced by the absence of every combination of the other participants from each meeting in the schedule. Thus, a schedule may be picked where some agents are not available for all meetings.

Recall the preferences in Table 8. Now suppose that Agent 1 loses 1 utility point each for the absence of the other two agents . Agent 2 loses 3 utility points from Agent 1's absence and 2 from Agent 3's. Agent 3 loses 2 from Agent 1's absence and 1 from Agent 2's. Suppose also, that when two agents are unavailable the utility for the remaining agent is always reduced by 10. When a combination of the agents is unavailable we reduce the utility of that schedule to the other agents by the amount they value the participation of the absent combination. Table 9 shows the reduction in utilities that occurs. To show how this works, lets consider Agent 2. Now Agent 2's original preference for sch_3 was 4, as shown in Table 8. But, Agent 1 cannot attend a meeting specified in this schedule, so Agent 2's valuation for sch_3 is reduced by 3, which is the amount that Agent 2 values Agent 1's presence at the meeting. The other reductions are computed similarly.

	sch_1	sch_2	sch_3	sch_4	sch_5	Tax
Agent1	1	4	0	0	0	0
Agent2	5	3	1	6	1	1
Agent3	2	4	7	7	1	1
Total	8	11	8	13	2	

Table 9. True preferences, including value for others

We now choose the social welfare maximizing schedule. Notice that the schedule chosen will be a schedule for which Agent 1 is unavailable - sch_4 since the absence of Agent 1 did not outweigh the others' values for that time. The payments can then be computed using the normal Clarke Tax formula. To show that this scheme maintains IC, if used once, we first need a formal expression for the tax that is imposed on the agents. We will define the tax as follows:

$$tax_i(\theta_i) = \sum_{j \neq i} v_j(x^*_{-i}) - v_j{}'(absent_{-i}(x^*_{-i})) \quad (3)$$

$$- \sum_{j \neq i} v_j(x^*) - v_j{}'(absent(x^*)) \quad (4)$$

where $v_j{}'$ computes the loss in utility to agent j of the absence of unavailable agents for each meeting and $x^*{}_{-i}$ is the outcome that would have been chosen had i not participated. The tax is thus, the value the other agents would have had for the social welfare maximizing outcome had i not been a participant in the process at all, minus the other agents' valuations for the outcome chosen.

Notice that agent i can have no effect on the first summand. Thus, i wants to report θ_i and his value for the presence of others such that the system choses x^* that maximizes the following equation:

$$v_i(x^*) - v_i{}'(absent(x^*)) + \sum_{j \neq i} v_j(x^*) - v_j{}'(absent(x^*))$$

But this is exactly what the mechanism is trying to maximize since this will give the social welfare maximizing outcome. Thus, agent i is best off reporting his true valuations.

So this system maintains IC if used once. Thus, we can use it in either MAMS1 or MAMS2 or some other mechanism to handle availability. Note that we could also use a similar method for handling availability in the mechanism that makes payments. This scheme can still result in outcomes that are not IR when combined with the Clarke Tax approach, however it will happen less often as demonstrated by the example.

6. Conclusions

The benefit of taking a mechanism design approach to MAMS is that it gives the possibility of maximizing social welfare with no strategic thinking required by the agents (they can just report their true valuations). However, to achieve this goal, we need to design a mechanism that is IC and *practical*. We have made significant progress towards developing such a mechanism in this paper.

We have demonstrated IC problems with previous approaches and have shown ways to fix or reduce these difficulties. Furthermore, we have addressed the practical issue of incorporating availability declarations into a mechanism. We have shown that it reduces the sorts of mechanisms available to us. Nonetheless, we have discussed two mechanisms that incorporate availability and have shown how IR problems can be reduced in the Clarke Tax version. Our results demonstrate that making mechanism design work in real-world multi-agent systems is a theoretically challenging problem.

The problem of designing online IC mechanisms or mechanisms that are IC when used repeatedly is an important problem for further study. Such mechanisms would have applications to multiple domains including meeting scheduling. Finally, the feasibility of any mechanism for MAMS relies on the existence of software agents that can learn people's scheduling preferences. As such, exploring this learning task is also an important direction for future research.

Acknowledgments

Thanks to Tuomas Sandholm and the Fall'03 CS 15-892 class for helpful feedback. Thanks also to the anonymous reviewers for their detailed comments. This research is sponsored by the Department of the Interior (DOI) - National Business Center (NBC) and the Defense Advanced Research Projects Agency (DARPA) under contract no. NBCHC030029. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the DOI, NBC, DARPA or the U.S. government.

References

[1] P. Berry, M. Gervasio, T. Uribe, K. Myers, and K. Nitz. A personalized calendar assistant. In *Working Notes from the AAAI Spring Symposium on Interaction between Humans and Autonomous Systems over Extended Operation, March*, 2004.

[2] M. Bowling, B. Browning, and M. Veloso. Plays as effective multiagent plans enabling opponent-adaptive play selec-

tion. In *Proceedings of International Conference on Automated Planning and Scheduling (ICAPS'04)*, 2004.

[3] H. E. Clarke. Multi-part pricing of public goods. *Public Choice*, 11:17–33, 1971.

[4] E. Crawford and M. Veloso. Mechanism design for multi-agent meeting scheduling, including time preferences, availability, and value of presence. In *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Beijing, China, September 20–24*, 2004.

[5] E. Crawford and M. Veloso. Learning to select negotiation strategies in multi-agent meeting scheduling. In *Working notes of the Multiagent Learning Workshop, AAAI, Pittsburgh USA*, 2005.

[6] M. B. Dias and A. T. Stentz. A free market architecture for distributed control of a multirobot system. In *the 6th International Conference on Intelligent Autonomous Systems (IAS-6)*, pages 115–122, July 2000.

[7] E. Ephrati, G. Zlotkin, and J. S. Rosenschein. A non–manipulable meeting scheduling system. In *the Proceedings of the International Workshop on Distributed Artificial Intelligence*, Seatle, WA, 1994.

[8] D. Fundenberg and J. Tirole. *Game Theory*. MIT Press, 1991.

[9] L. Garrido and K. Sycara. Multi-agent meeting scheduling: Preliminary experimental results. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS'95)*. The MIT Press: Cambridge, MA, USA, 1995.

[10] A. Gibbard. Manipulation of voting schemes. a general result. *Econometrica*, 41:587–602, 1973.

[11] J. R. Green and J.-J. Laffont. Characterization of satisfactory mechanisms for the revelation of preferences for public goods. *Econometrica*, 45:427–438, 1977.

[12] T. Groves. Incentives in teams. *Econometrica*, 41:617–631, 1973.

[13] M. Hajiaghayi, R. Kleinberg, M. Mahdian, and D. Parkes. Online auctions with re-usable goods. In *Proc. 6th ACM Conf. on Electronic Commerce (EC'05)*, 2005.

[14] N. R. Jennings and A. J. Jackson. Agent based meeting scheduling: A design and implementation. *IEE Electronics Letters*, 31(5):350–352, 1995.

[15] R. Lavi and N. Nisan. Online ascending auctions for gradually expiring items. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1146–1155, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics.

[16] P. J. Modi and M. Veloso. Bumping strategies for the private incremental multiagent agreement problem. In *the Working Notes of the AAAI Spring Symposium on Persistant Agents*, 2005.

[17] P. J. Modi, M. Veloso, S. F. Smith, and J. Oh. Cmradar: A personal assistant agent for calendar management. In *the 6th International Workshop on Agent-Oriented Information Systems*, 2004.

[18] C. Ng, D. C. Parkes, and M. Seltzer. Virtual worlds: fast and strategyproof auctions for dynamic resource allocation. In *EC '03: Proceedings of the 4th ACM conference on Electronic commerce*, pages 238–239, New York, NY, USA, 2003. ACM Press.

[19] N. Nisan and A. Ronen. Algorithmic mechanism design (extended abstract). In *STOC '99: Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 129–140. ACM Press, 1999.

[20] D. Parkes. *Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency*. PhD thesis, University of Pensylvania, 2001. Chapter 2, Classic Mechanism Design.

[21] R. Porter. Mechanism design for online real-time scheduling. In *EC '04: Proceedings of the 5th ACM conference on Electronic commerce*, pages 61–70, New York, NY, USA, 2004. ACM Press.

[22] M. Satterthwaite. Strategy-proofness and arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10:187–217, 1975.

[23] S. Sen and E. H. Durfee. A formal study of distributed meeting scheduling. *Group Decision and Negotiation*, 7:265–289, 1998.

[24] S. Sen, T. Haynes, and N. Arora. Satisfying user preferences while negotiating meetings. *the International Journal of Human-Computer Studies*, 47(3):407–427, 1997.

[25] T. Shintani, T. Ito, and K. Sycara. Multiple negotiations among agents for a distributed meeting scheduler. In *Proceedings of the Fourth International Conference on Multiagent Systems*, pages 435 – 436, July 2000.

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