

Scheduling Meetings by Agents

A. Grubshtein⁽¹⁾, A. Gershman⁽¹⁾, A. Meisels⁽¹⁾, L. Rokach⁽²⁾ and Roie Zivan⁽³⁾

⁽¹⁾Department of Computer Science

⁽²⁾Department of Information System Engineering

⁽³⁾Department of Industrial Engineering and Management
Ben-Gurion University of the Negev
Beer-Sheva, 84105, Israel

Abstract

The Scheduling of Meetings of multiple users is a real world problem that was studied intensively in recent years. Most former studies used a simplified version of the problem as a benchmark for evaluating constraint satisfaction and optimization algorithms. The present paper investigates the variety of aspects that need to be taken into consideration in order to design a realistic model for representing and solving meetings scheduling problems (MSPs). The proposed model represents the multiple components of the real-world problem in terms of their utilities and costs and enables the use of constraints optimization algorithms to solve MSPs.

A central component of the proposed model of MSPs is a mechanism to balance the trade-off between competitive and cooperative environments. Agents solve the problem by balancing the global (e.g., cooperative) optimum against typical self-interests of users. These are represented in the model by the quality of the resulting personal schedule.

The experimental evaluation of the features of the proposed model uses a complete optimization algorithm and an alternative Local Search Algorithm which produces a high quality (but not necessarily optimal) solution in a reasonable time.

1 Introduction

In the modern world the use of electronic calendars is becoming more and more popular. Simple tools such as *Microsoft Outlook*, for example, allow users to maintain their scheduled appointments electronically. The fact that the information regarding people's schedules is held mainly electronically makes the next step towards a calendar that will not only hold the information but will schedule it as well, natural and desired. Indeed, many entrepreneurs, corporations and researchers [BCG⁺06, MVSO04, SK06, Jen95, WL06, eOp] sought means to devise an intelligent scheduling mechanism. Despite the above attempts, most electronic calendars are little more than passive information repositories. The meeting scheduling problem is also a well studied application with regard to distributed computation [WF02, ML04, MV04]. However, most of these studies investigate a refined problem which can be easily represented by the standard models of constraint satisfaction or constraint optimization and do not capture many of the aspects of the problem and therefore are not applicable.

The goal of this paper is to study the aspects and challenges of this relevant application in order to produce a model and solving algorithms that can be used for online meeting scheduling by agents. We propose a model in which the special characteristics and preferences of people and organizations are represented as utilities and costs. In the proposed model utilities represent *gains* of agents, from meetings that are scheduled and are going to take place. The underlying assumption is that all agents participating in some meeting gain something, otherwise they would not participate. Costs, which are *negative*, represent disadvantages of scheduled meetings for the participating agent. Costs arise from the fact that some personal schedules are worse than others. The scheduling of a meeting can result in a less preferable weekly (or daily) schedule for the agent and this fact is represented by an incurred cost.

The relation between utilities (advantages) and costs (disadvantages) of the problem's components is very important. It enables a cooperation of agents in scheduling their meetings (thereby gaining their individual utilities) and at the same time a representation of the conflicting interests of agents, each trying to minimize the cost of its personal schedule. The tuning of these costs can also be used by the system administrator for implementing a policy within an organization. The resulting problem that represents our model is a maximization constraint optimization problem which can be solved either by a central agent or by cooperating agents in a distributed way. We discuss in Section ?? why a distributed solution is preferred.

One of the most important questions deals with the relations between the agents in the problem. One can expect a different outcome for a meeting scheduling problem in which agents are all members of the same organization than for a problem with the same inputs in which the agents belong to different, possibly competing organizations. These two extremes are dealt in previous studies by very different mechanisms. In standard *DisCSP* and *DisCOP* studies, the agents are expected to be fully cooperative and follow the protocol regardless of their private good [YH00, MSTY05, BMBM05, ZM06]. Thus, the outcome of such a protocol is a solution whose global utility is optimal, yet does not necessarily account for the personal schedule of participants. An alternative approach is offered by researchers in the field of *Game Theory*, which deals with a competitive environment where agents are self-interested and their utilities are private (and often conflicting) [BDS06, BFHS07, NRTV07, EZR94, CV06, PFP06]. The latter three, in particular, employed a monetary oriented approach to this problem. Their models point out many of the difficulties involved in solving the MSP.

The present paper proposes a model that gives a balanced solution to this trade-off by using a global objective function which limits the tolerance of agents towards the cost they are expected to pay in order to achieve a higher global utility. In other words, it enables agents to limit the damage to the quality of their personal schedules that results from a better (e.g., higher utility) *global* solution. The amount of this tolerance can be tuned by the system's administrator in order to achieve a desirable balance which is required between these two extremes.

In order to evaluate the proposed model for the Meetings Scheduling Problem a set of preliminary experiments that determines the behavior of the problems and its dependency on important parameters is performed. In computing the behavior of *MSPs* under changes of parameter values a complete constraints optimization algorithm and a local search algorithm based on [ZXWW05] are used. The results of both approaches are quite similar. The local search algorithm produced high quality solutions in a reasonable time. Such an incomplete method is required for solving large problems.

The rest of the paper is organized as follow. Section 2 presents a detailed problem definition for the Meeting Scheduling problem. It models multiple parts of agents' pri-

orities and preferences and incorporates an intuitive version of individual rationality. The other proposed part for individual rationality incorporates self-interested agents in an optimization mechanism for a global optimal schedule. This part of the model balances a fully cooperative strategy, which maximizes a global sum of personal utilities, and the self-interested approach that focuses on the *quality* of personal schedules.

A preliminary experimental evaluation of the behavior of *MSPs*, their difficulty and their solving algorithms is presented in Section 3. A discussion of the proposed approach to Scheduling Meetings by Agents is in Section 3.1.

2 Meetings Scheduling - Problem Definition

Consider a large set of users whose personal calendars are connected by constraints that arise from the fact that multiple groups of the overall set of users need to participate in meetings. Meetings involve groups of users and include at least two participants. Meetings can take place in different places and carry a different significance for their participants. In addition, all participating users have their own personal calendars and need to coordinate all their meetings. It is realistic to assume that users can move some of their already scheduled meetings, in order to enable the scheduling of additional meetings. In order to automate the search for a solution to the Meeting Scheduling Problem (MSP), each user is represented by an agent and the set of agents coordinate and schedule all meetings of all users. The final result is a set of updated personal calendars for all agents/users.

Each agent participates in one or more meetings and each meeting has 2 or more participants. A general MSP uses utilities to represent the significance that users attach to meetings and to certain features of the resulting schedule, as it appears in their personal schedules. There are two main goals to the general definition of the MSP.

1. Capture as many features of the preferences and considerations that people use, when constructing real world schedules, as possible.
2. Try to express the trade-off between the (possibly conflicting) personal preferences of users about their personal schedules and the overall goal of scheduling all meetings of all participating users.

The proposed model captures these two goals by using two parts in the user's assessment of quality of schedules. One part is the utility arising from meetings that take place. These are essentially positive. They represent gains for the participating agents. The other part is termed *cost* and represents the degradation of a user's personal schedule that is caused by the scheduling of the meeting at hand. Naturally, the size of utilities (gains) depend on the nature of the meeting, on the time it is scheduled to take place, and on other preferences of the participating agents. These utilities have a generic structure and components (that apply to all agents) but are evaluated individually by each agent. The general structure of the utilities and preferences for agents is described below. The features listed below are parametrized by a set of values that represent a wide range of realistic and useful MSPs.

We start with the definitions of a calendar, of meetings, and of priorities of meetings and agents. These serve to define the preferences of agents with respect to their meetings and with respect to the importance of the agents. Importance is both intrinsic (e.g., the initiator of a meeting) and meeting-dependent.

Definition 1 Calendar - a collection of days, each day is divided into time-slots.

Definition 2 PC - Priority Class of an agent, from a predefined range of integers.

Definition 3 PF - Priority Factor - determines the difference between priority classes.

Definition 4 AMU - Agent Meeting Urgency - a number from a predefined range. All participants of a meeting have an AMU for it. A higher AMU value means that the meeting is more important for the agent (e.g., of increased priority for the agent).

Definition 5 DCF - Deadline Cost Function. Each meeting has a start time (ST), a soft deadline (SD) and a hard deadline (HD). A meeting can only be scheduled after ST and before HD . The Deadline Cost Function (DCF) assigns a cost to scheduling the meeting in every time-slot in the allowed range. In the interval between ST and SD the DCF cost is zero. For the remaining interval, between SD and HD , the DCF is a monotonically none-decreasing function (which can be a constant k).

Definition 6 MU - the Meeting Utility sums up the importance of a meeting for an agent. We define $MU = PF * PC + AMU - DCF$, for all agents.

For every personal schedule (calender) of an agent additional parameters can define its costs that represent degradation in quality of the schedule. Such costs represent the preferences of users about their personal schedules and therefore can be specified by each user differently.

Definition 7 FTC - Free Time-slot Cost is a function that defines for each agent and each time-slot, how much utility is lost by not keeping it free.

Based on the above definitions of the building blocks of schedules and users' preferences, one can define the computed utilities for meetings and for their scheduled time-slot. These are the utilities that are gained by agents. A fully cooperative scheduling algorithm will try to maximize the overall utility which is the sum of the utilities of all meetings over all agents.

Consider the following example scenario. Professor Phil and his two research assistants Alice and Anna would like to schedule some meetings between them. In addition, each has to schedule some meetings with their own students. For simplicity we assume the following:

- The calender of each individual contains a single day composed of 4 time-slots.
- Students have no other meetings, and no preference of when should the meetings be scheduled.
- Prof. Phil was assigned a priority factor (PC) of 3, his two assistants were assigned a PC of 2 and all students have a PC of 1.
- The Priority Factor is set to 5 ($PF = 5$).
- For all meetings, the ST is set to 10am and the SD and HD are set to 6pm.
- The DCF for all time-slots is zero.
- Each individual may adjust the utility for meetings she participates in by at most one unit ($-1 \leq AMU \leq 1$).

There are a total of six meetings to schedule:

Calendar of Prof. Phil (A_1) PC = 3	Timeslot	Meeting ID	MU_1	$\sum MU_i$	MC
	10:00	M_1	14	24	5
	12:00	M_2	16	38	7
	14:00	Free			
	16:00	M_3	15	20	5

Calendar of Alice (A_2) PC = 2	Timeslot	Meeting ID	MU_2	$\sum MU_i$	MC
	10:00	Free			
	12:00	M_2	11	38	7
	14:00	M_4	11	16	3
	16:00	M_5	10	20	4

Calendar of Anna (A_3) PC = 2	Timeslot	Meeting ID	MU_3	$\sum MU_i$	MC
	10:00	M_6	10	25	5
	12:00	M_2	11	38	7
	14:00	Free			
	16:00	M_5	10	20	4

Figure 1: Example schedule of 3 agents that participate in 6 meetings.

1. M_1 : a meeting between Prof. Phil and two of his students.
2. M_2 : a meeting between Prof. Phil, Alice and Anna.
3. M_3 : a meeting between Prof. Phil and one of his students.
4. M_4 : a meeting between Alice and one of her students.
5. M_5 : a meeting between Alice and Anna.
6. M_6 : a meeting between Anna and three of her students.

The Meeting Utility (MU) assigned by Prof Phil for meeting M_2 is 16 since his $PC = 3$ and $PF = 5$, which when multiplied equals 15. This meeting is important for the Prof, so he chose to increase its utility by 1 (which is within the bounds of the AMU). Alice and Anna also think that this meeting is important and chose to increase their utilities by 1, but since their $PC = 2$ their Meeting Utility is only $MU = 11$. The sum of utilities of all participants is therefore $\sum MU_i = 16 + 11 + 11 = 38$.

Figure 1 presents an optimal schedule for these six meetings. Luckily, all meetings can be scheduled, and there is even a free time-slot in each personal calendar. For clarity, the calendars of the students are not shown.

In order to demonstrate the addition of another meeting a few more definitions are required. Since the MSP accommodates existing personal calendars (schedules), utilities need to be attached also to operations of *re-scheduling of existing meetings*. This enables a complete evaluation of the utility (or gain) from the scheduling of a meeting.

Definition 8 MC - *The Moving Cost describes the loss of utility generated from assigning an already scheduled meeting to a different time-slot.*

This cost is necessary since we would like to have stable schedules, which do not change too much. Therefore, rescheduling an existing meeting would only be done in case that the gain of utility from rescheduling is greater than the cost of moving it.

Incorporating the above definition of the building blocks of a meeting scheduling problem, one arrives at the following complex model. A meeting is represented by a tuple $\langle A, MP_s, MC_s, ST, SD, HD \rangle$ where:

- A is a set of agents that need to participate in the meeting
- MU_s is a list of MU , the i -th element corresponds to the i -th agent in A
- MC_s is a list of MC , the i -th element corresponds to the i -th agent in A
- ST is the earliest possible Start Time to schedule the meeting, SD and HD are the Soft and Hard Deadlines for this meeting.

By using the above definitions one can compute the meeting utility (MU) for all meetings of all agents. This is the gain that is associated with scheduled meetings for all participating agents. Each according to its priorities and personal utilities.

Definition 9 A schedule s_i of some agent A_i assigns either a time-slot or a special value null to each meeting that the agent participates in. A schedule is valid if no two meetings in it occur at the same time.

Definition 10 A global schedule S contains a schedule for each agent. A global schedule is consistent if each schedule in it is valid and for each meeting, every schedule of every agent participating in the meeting assigns it the same value.

All of the parameters defined above attempt to capture the components of the gain to agents from the scheduling of meetings. The gain to each agent, from the fact that some meeting is scheduled and will take place, arises from its personal priorities with respect to the given meeting. The meeting utility (MU) includes also negative parts (such as the DCF), but the overall value is positive. This is a manifestation of a well known principle from games theory - *Individual Rationality* (IR) [Par08, SLB08]. *Individual Rationality* is a term used to describe the willingness of agents to participate in the process. In classic game theory in order to achieve individual rationality it is enough that the outcome of the process for agents will be non-negative [Par08].

The present model for MSPs incorporates IR by assuming that only a meeting for which $MU > 0$ is going to be scheduled. Otherwise, the agent would simply not participate in the meeting. In a sense, the model assumes realistically that agents must agree to any given schedule of a meeting in which they participate. This means that they gain something from the meeting (e.g., $MU > 0$) and hence Individual Rationality holds. All of the above parts that compose the utility of a scheduled meeting relate to the fact that the meeting takes place within its legal span of time.

Returning to the example in Figure 1, let us assume that Prof. Phil (A_1) wants to add a new meeting (M_7) with Alice (A_2). The moving costs (MC) for the already scheduled meetings are listed on the right-most column in Figure 1. The new Meeting Utilities for M_7 are:

- for Prof. Phil (A_1): $AMU = 1$, and therefore $MU_1 = 16$.
- for Alice (A_2): $AMU = 1$, and therefore $MU_2 = 11$.

Calendar of Prof. Phil (A_1) PC = 3	Timeslot	Meeting ID	MU_1	$\sum MU_i$	MC_1
	10:00	M_1	14	24	5
	12:00	M_7	16	27	2
	14:00	M_2	16	38	3
	16:00	M_3	15	20	5

Calendar of Alice (A_2) PC = 2	Timeslot	Meeting ID	MU_2	$\sum MU_i$	MC_2
	10:00	M_4	11	16	3
	12:00	M_7	11	27	3
	14:00	M_2	11	38	2
	16:00	M_5	10	20	3

Calendar of Anna (A_3) PC = 2	Timeslot	Meeting ID	MU_3	$\sum MU_i$	MC_3
	10:00	M_6	10	25	5
	12:00				
	14:00	M_2	11	38	2
	16:00	M_5	10	20	1

Figure 2: Resulting schedule after adding meeting M_7 .

Prof. Phil and Alice want to add meeting M_7 and a new optimal global schedule will be achieved if we move the meeting M_4 of Alice (A_2) and her student from 14:00 to 10:00. Then the meeting M_2 can be moved from 12:00 to 14:00, and the new meeting M_7 can be scheduled at 12:00. The resulting state is presented in Figure 2. By rescheduling M_4 to 10:00 Alice loses 3 utility units (MC_2 for this meeting is 3). By rescheduling M_2 to 14:00, the Prof. loses 3 units, Alice loses 2 units and Anna loses 2 units. The combined utility lost from moving meetings is therefore $3 + 3 + 2 + 2 = 10$. The new meeting M_7 adds a utility of $MU_1 + MU_2 = 16 + 11 = 27$. Thus, the total combined utility is increased by $27 - 10 = 17$.

Note that in this example Anna (A_3) lost 2 units from her schedule utility due to the moving cost associated with the rescheduling of M_2 . Even though this action does not improve Anna's personal schedule, if we take a closer look, it does not harm it either. Anna's personal schedule before and after the rescheduling, remained equal in total gain. The moving cost Anna attached to M_2 would prevent rescheduling for no global gain or for a very small global gain, and thus keep Anna's schedule fairly stable. Only at the chance of gaining more than this cost, would a rescheduling occur. The rational behind this model's behavior (as opposed to refusing to accept any rescheduling that would not directly increase Anna's personal schedule) is that Anna, Alice and the Prof. are all part of the same organization. Each individual maintains Individual Rationality, by ensuring there is a personal gain in accepting the proposed schedule. At the same time, each individual would also like to help others improve the quality of their schedules, as long as it does not harm its personal gain.

Let us now further extend our example, by trying to add another meeting (M_8) between Prof. Phil and Anna (A_3). This new meeting is less urgent for the Prof. than the meeting with Alice and therefore he assigns it a lower priority. The new Meeting Utilities are:

- For the Prof. : $AMU = 0$ and $MU_1 = 15$.

- For Anna (A_3): $AMU = 1$ and $MU_3 = 11$.
- The combined Meeting Utilities are $MU_1 + MU_3 = 15 + 11 = 26$.

The system makes the following attempts to schedule the new meeting:

1. At 10:00, the Prof. is meeting two students (M_1) and Anna is meeting 3 students (M_6). Canceling them would cost both of their utilities 49 units. Canceling meetings worth 49 units of utility in order to add a meeting worth 26 units is illogical. If Anna will move her meeting to the empty slot she has at 12:00, the cost would be 24 for canceling the meeting of the Prof. and 5 for moving the meeting of Anna. All together, 29, which is still larger than the potential gain of 26.
2. At 12:00, Anna has an empty slot but the Prof. has a meeting worth a total of 27, which again is greater than the potential gain of 26.
3. At 14:00, both of them are participating in a meeting with utility 38.
4. At 16:00, canceling the meeting of the Prof. costs 24 and canceling the meeting of Anna costs 24 (48 together). Moving the meeting of Anna to 12:00 will cause the cancellation of the meeting between the Prof. and Alice at 12:00 which is too expensive.

Final Outcome: the meeting is not scheduled.

2.1 The cost of scheduling

The fact that the utility of a scheduled meeting is positive is a necessary condition for agents to accept it (e.g., IR). However, there must be another part to individual rationality, that relates to the whole personal schedule of an agent. Think of the process of scheduling meetings as a mechanism that agents are aware of and which automatically schedules meetings for all agents. The process of scheduling meetings involves a hard combinatorial search. Solving the constrained optimization problem that involves all meetings, priorities that agents assign to these meetings, and any additional constraints. Agents need to accept the result of the combinatorial optimization. The acceptance condition relates to the quality of the resulting personal schedule for every agent. The present paper proposes a model for incorporating the quality of the personal schedules into the scheduling mechanism. Two parts are needed in order to achieve this goal - a universal measure for schedule quality, and a form of using these qualities in the global optimization procedure.

Let us dwell upon the idea of establishing individual rationality through the optimization mechanism. The process of scheduling meetings includes two parts. One part is completely individual and can be performed by each agent separately and privately. This part was described above in the form of utilities and costs that are assigned individually by every agent. The other part is global. It can be thought of as a cooperative process of all agents. This is the part that searches for the best global schedule, performing a hard optimization computation. However, the problem remains - *what global objective to optimize?* A trivial solution is to optimize the global sum of all agents' utilities. This is a fully cooperative scenario that has been assumed by all experimental evaluations of distributed constraints optimization algorithms (cf. [MV04, WF02]). The

present paper incorporates self-interested agents in the underlying cooperative scheduling problem. This fact rules out the simple mechanism of optimizing the global sum of utilities (or gains). A mechanism that optimizes such a simple sum does not guarantee a reasonable personal schedule to all participating agents. A large sum of utilities can include very small utilities for some of the agents and this would be unacceptable.

To evaluate the quality of a personal schedule one needs to take into account complex relations among multiple scheduled meetings. Are they too dense ? too sparse ? do they leave enough free time ? etc. The general evaluation approach proposed by the present paper attempts to do this in a very naive but general way. It defines an ideal schedule for each agent, one that incorporates all of the agent's requests, and then evaluates every schedule by its relation to the ideal one. The ideal schedule is defined to have the maximal total utility for the agent. In other words, a total utility that is the sum of all the utilities of the agent's meetings without any deduction of costs, such as free-time costs, etc. The total utility of the ideal schedule for any given agent is composed of meetings that are each scheduled to an ideal time-slot, not interfering with any other of the agent's concerns. Such a personal schedule may not be even potentially possible, because of unavailability of enough non-conflicting slots. Nonetheless, it can form for each agent a scale against which it can evaluate its current schedule. The evaluation of any personal schedule can be the ratio of the utility of the schedule to the utility of the ideal schedule for that agent. The advantage of this simplistic measure of quality is that it takes into account the differences among different agents' personal calendars. An agent with many meetings that are scheduled well, with no additional costs, can have the same quality as an agent with fewer meetings. On the other hand, the quality of the schedule of an agent that participates in more meetings and that has meetings that are not ideally scheduled and incur less than optimal utilities is lower because the resulting ratio is lower.

The ratio:

$$quality(s) = \frac{U_i(s)}{\sum_{\text{all meetings of } A_i} MU_{s_i}}$$

is the ratio between the gain in the current schedule and an estimate of the maximal utility that can be gained from a personal schedule of the agent A_i . The denominator is fixed for each agent. This ratio can serve as a universal criterion for the quality of the personal schedule of agents. The trade-off between *quality* and the maximal utility that is found by the optimization algorithm can serve to guarantee the acceptance of the proposed scheduling mechanism, by the participating self-interested agents. The goal is to express the trade-off between finding a global maximal utility and minimizing the deterioration of personal schedules. The global utility represents the success of the scheduling procedure in finding a schedule that accommodates all meetings. The minimization of the sum of $(1 - quality)$ for all personal schedules represents a wish to minimize the degradation of personal schedules, that results from the global optimum.

The above definitions provide all of the needed ingredients for the general (multi-agent) meetings scheduling problem and can be summarized as follows:

Definition 11 *The Meeting Scheduling Problem - Input: a tuple $\langle A, S', M, f \rangle$ in which A is a set of agents, S' is a consistent schedule for all agents in A , M is the collection of all scheduled and unscheduled meetings, f is a function that takes a tuple of "quality" measures for all agents as a parameter and returns a global utility.*

Output: A new consistent scheduling S , that maximizes an objective that combines cooperation and self-interest. One simple example is $G = \text{global utility} - f(\text{quality})$.

The above definitions were designed to provide the resulting personal and global schedules with the following properties:

1. A complete, legal and consistent global schedule - all meetings are scheduled or labeled as unscheduled. No two meetings with a common participant(s) are scheduled at the same time. All participants of a meeting scheduled it to the same time-slot.
2. The global schedule is stable. Meetings cannot be re-scheduled without degrading the global utility. The Moving Cost can be adjusted to express the degree of stability one is willing to sacrifice in order to achieve a better schedule.
3. The priority of a meeting increases with the number of participants.
4. The global schedule is fair. An optimal global schedule has the maximal overall utility, subject to an acceptable quality of all personal schedules. The cost of quality is subject to a factor (QF) that can express how important quality is for the system. below).

3 Experimental Evaluation of MSPs

Our empirical studies include four sets of experiments which evaluate a variety of aspects of our proposed model.

The experimental setup uses a 3 day week, each day with 3 time-slots (overall 9 possible time-slots for each meeting). There are 4 agents of varying priority classes which scheduled a total of 10 meetings. The participants of each meeting were selected randomly. The values of the quality of schedules were between 80% and 60%. An agent that reaches a utility smaller than 80% of its utopian utility loses 15% of its utility as a quality-related penalty and an agent that reaches a utility smaller than 60% of its utopian utility loses 50% of its utility as a penalty. The moving cost factor was set to 1 which means that the cost of moving a meeting was determined by the priorities of the participants. The time-slot cost is 1 for the mid day time-slots and 2 for the time-slots at the beginning and at the end of the day.

In order to produce the results we used a complete *COP (B&B)* algorithm and a *local search algorithm*. Our local search algorithm is based on the *DSA [ZXWW05]*, and employs *anytime mechanism* proposed by [?]. In each iteration, an agent first locates all of her meetings which are still not properly scheduled. She then proceeds with assigning a new time for these meetings - either the time proposed by the initiator of the meeting, or, in case she is the initiator, by randomly selecting one of her free time slot. Using the anytime mechanism we may further explore the solutions space and thus the agent may decide to unassign a meeting with some probability P_E . Finally, a decision to keep the above mentioned changes is made, again - according to some probability. If the agent decides to keep the changes, she marks them in her schedule, and notifies relevant agents of the update. The local search algorithm was executed with 1000 iterations.

In the first experiment the varying parameter was the length of the interval between the *start deadline* and the *soft deadline*. The hard deadline is randomly picked to be any day past the soft deadline. Intuitively, the larger the interval, the easier the scheduling is expected to be.

Three different utilities were measured in each experiment.

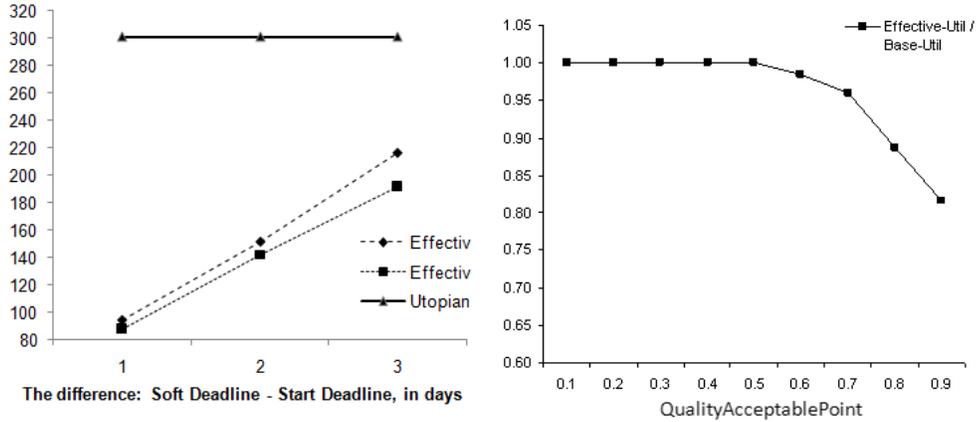


Figure 3: On the left - The relation between the length of the interval between the start and soft deadline and the optimal utility. On the right - The ratio of the effective utility to the base utility against the quality factor.

1. *Utopian Utility*: is a fixed number for a given problem. It is simply the sum of all meeting priorities.
2. *Effective Optimal Utility*: is the utility of the optimal feasible solution found by a complete algorithm.
3. *Effective Utility LS*: is the best utility found by our *local search algorithm*.

The left hand side of Figure 3 presents these utilities for different intervals (in days) between the start and the soft deadlines. For each interval 10 different problem instances were solved and the results presented are an average of the results of these runs. As expected, the optimal effective utility increases when the interval is larger. The *local search* algorithm produced solutions with lower utilities, but still close to the optimal solution.

In the second experiment we examined the effect of quality-related penalty on the global utility results. The setup of the problem was the same as in the first experiment. An agent that reached a utility smaller than the *QualityAcceptablePoint* of its Utopian utility lost 15% of its utility as a quality-related penalty and an agent that reached a utility smaller than the *QualityAcceptablePoint* - 0.2 of its Utopian utility, lost 50% of its utility as a quality-related penalty.

The right hand side of Figure 3 presents the effect of the quality-induced cost on the effective utility. This was measured by the ratio of the Effective utility to the *Base* utility. The *Base* utility is the same as the effective utility, but with a *zero quality cost*. For values of the *QualityAcceptablePoint* parameter, of up to 0.5 the ratio between base and effective utility is 1, since there are no quality penalties. As the parameter increases, we can see that the effective utility decreases. This is true up to a point where roughly 20% of the utility is lost due to quality penalties (for a parameter of 0.9).

In our third experiment the effect of adding meetings to the utility is measured. The setup of the problem is similar to the previous experiments. An agent that reached a utility smaller than 0.6 of its Utopian utility lost 15% of its utility as a quality penalty

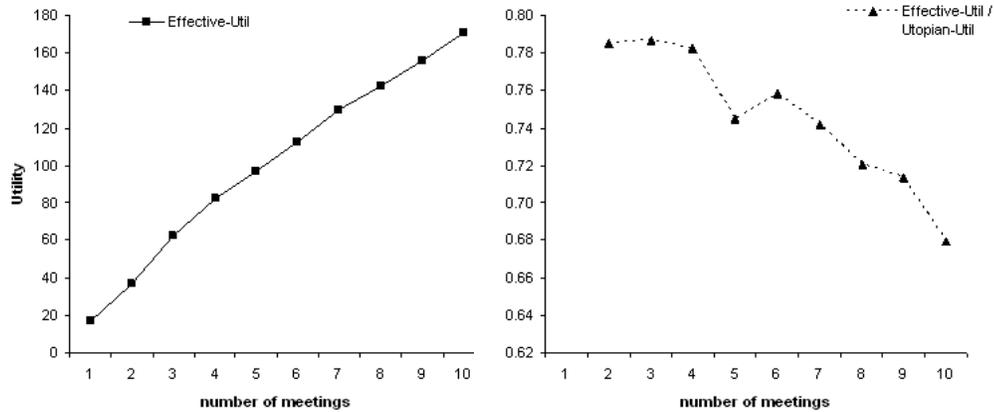


Figure 4: On the left - The effective utility after the addition of meetings to the schedule. On the right - The ratio of the Utopian and effective utilities, after the addition of meetings to the schedule.

and an agent that reached a utility smaller than 0.4 of its Utopian utility lost 50% of its utility.

In our experiments, the meetings were not all scheduled at once. A realistic incremental process was used, which started with an empty schedule, and at each iteration optimally solved the scheduling problem with a single new additional meeting. The solution to the last iteration provided the base schedule for the next iteration. The rescheduling (moving) of a meeting that was scheduled in a previous iteration suffers a moving cost penalty.

The left hand side of Figure 4 presents the increase in the effective utility, as more meetings are added. This is intuitively clear, since the new schedule can either:

1. Leave the new meeting unscheduled and not changing the utility.
2. Increase the utility by including the new meeting.

The optimal utility does not decrease because any previous schedule (with the new meeting left unscheduled) is still a valid schedule, with the same utility it previously had. The new meeting added only provides more opportunities for possibly better schedules.

The right hand side of Figure 4 depicts the ratio between the effective and Utopian utility. The ratio decreases as more meetings are added. When the number of meetings is low, the ratio is high. It appears that most of the meetings can be scheduled, and without much penalties (such as deadline cost). This is the result of the fact that the schedule is fairly empty. As the number of meetings increases, meetings are scheduled in less "optimal" time-slots, which means paying more costs (e.g., lowering their overall utilities), and thus this ratio decreases.

A real world MSP encompasses anywhere between several dozens to thousands of users. Each user has multiple meetings, and usually, an even larger number of possible time slots to place them in. Solving such a problem with a complete *COP* solver is not practical. In the fourth and last set of experiments the local search algorithm was evaluated solving large MSPs. To this end a much larger problem was generated which included 20 agents of varying priority classes attempted to schedule 30 meetings, over

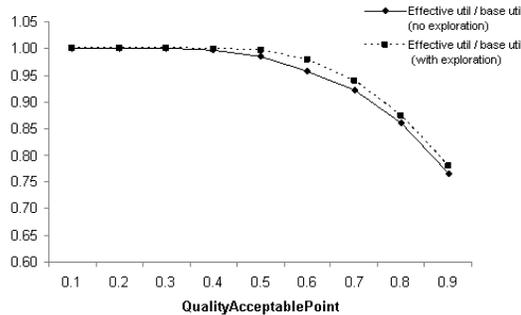


Figure 5: The ratio of the effective utility to the base utility when the quality-related penalty increases while running a local search, with and without exploration.

a 4 days week with a 3 time-slots day. Participants of each meeting were completely random. As before, the moving cost factor was set to 1 and the time-slot cost was 1 for mid day time-slots and 2 for the time-slots at the beginning and at the end of each day. An agent that reached a utility smaller than the *QualityAcceptablePoint* of its Utopian utility lost 15% of its utility as a quality-related penalty and an agent that reached a utility smaller than $QualityAcceptablePoint - 0.2$ of its Utopian utility, lost 50% of its utility as a quality-related penalty. We examined the effect of the quality-related penalty on the global utility results as was done with the complete *COP* solver. The experiment was conducted twice: with no exploration ($P_E = 0$) and with a small amount of exploration ($P_E = 0.001$). Both runs were limited to 1000 iterations, and solutions were produced in less than a minute.

Figure 5 presents the effect of the quality-related penalty on the effective utility. Again, we measured the effect by presenting the ratio of the Effective utility to the *Base* utility. The results of our *MSP-DSA* act roughly the same as the *complete search* - a ratio of nearly 1.0 is maintained up to a quality acceptable point limit of 0.5, after which it decreases (slightly faster than the *complete search*) due to quality-related penalties.

3.1 Discussion

The results presented in the previous section reveal a number of properties and aspects of the meetings scheduling problem. First, our set of experiments shows that as deadlines are further apart the common utility is higher. In a cooperative environment the result would be beneficial. However, in a competitive environment where some agents always choose the most tight deadlines the common utility would decrease.

The experimental results indicate that local search works well when compared to the optimal solution. This fuels our hopes that the proposed model can be used for very large search spaces and reduce the time needed to find solutions that are close to optimal. In real-world meetings scheduling it is many times the case that waiting too long for the optimal solution is not an acceptable option.

Our second experiment presents the effect of our proposed quality-based function on the common utility. The slow decrease in the utility as the quality parameter is incremented enables a system administrator to apply the policy in the organization.

The extreme trade-off between fully cooperative and fully competitive environments can be balanced at any point on the scale.

Our third experiment demonstrates the realistic incremental aspect of generating a schedule. Previous studies treated the meetings scheduling problem as a given combinatorial problem that needs to be solved [WF02, ?, ML04]. However, real life calendars are incremental by nature and the requirement of individual rationality (i.e., that gains must be positive) constraints the schedule of additional meetings.

4 Conclusions

The Meetings Scheduling problem is a challenging distributed constraints optimization problem. Beside being a hard problem which requires a search algorithm in order to be solved, it combines aspects of personal good of agents (which expect a high quality personal schedule) and of a global good (i.e., a feasible solution with as many scheduled meetings as possible). The present paper proposes a detailed model for MSPs which incorporates two forms of individual rationality. On the one hand, the model guarantees non-negative gains (e.g., utilities) for all agents, so that all agents choose to participate. The proposed model captures a wider range of aspects of the problem than previous papers by emphasizing the trade-off between cooperative and competitive search by agents. It investigates a combinatorial aspect of individual rationality, incorporating the combinatorial search for a globally feasible schedule with an acceptable quality for personal schedules or calendars. The preliminary experimental results show that a balanced point between global and personal good can be found.

References

- [BCG⁺06] P. Berry, K. Conley, M. Gervasio, B. Peintner, T. Uribe, and N. Yorke-Smith. Deploying a personalized time management agent. In *Proc. 5th Intern. Joint Conf. Auton. Agents and Multi Agent Systems (AAMAS-06)*, pages 1564 – 1571, Hakodate, Japan, 2006.
- [BDS06] M. Benisch, G. B. Davis, and T. Sandholm. Algorithms for rationalizability and curb sets. In *AAAI-06*, pages 598–604, Boston, MA, USA, July 2006.
- [BFHS07] F. Brandt, F. Fischer, P. Harrenstein, and Y. Shoham. A game-theoretic analysis of strictly competitive multiagent scenarios. In *Proc. 20th Intern. Joint Conf. Artif. Intell. (IJCAI-2007)*, pages 1199–1206, Hyderabad, India, January, 2007.
- [BMBM05] C. Bessiere, A. Maestre, I. Brito, and P. Meseguer. Asynchronous backtracking without adding links: a new member in the abt family. *Artificial Intelligence*, 161:1-2:7–24, January 2005.
- [CV06] E. Crawford and M. Veloso. Mechanism design for multi-agent meeting scheduling. *Web Intelligence and Agent Systems*, 4(2):209–220, 2006.
- [eOp] eOptimize. Product: About:time for exchange server. <http://www.eoptimize.com/exchange.asp>.
- [EZR94] E. Ephrati, G. Zlotkin, and J.S. Rosenschein. A non manipulable meeting scheduling system. In *Proc. Intern. Workshop on Distributed Artificial Intelligence*, Seattle, WA, 1994.
- [Jen95] A.J. Jennings, N.R.; Jackson. Agent-based meeting scheduling: a design and implementation. *Electronics Letters*, 31(5):350–352, 2 Mar 1995.

- [ML04] A. Meisels and O. Lavee. Using additional information in discsp search. In *Proc. 5th workshop on distributed constraints reasoning, DCR-04*, Toronto, 2004.
- [MSTY05] P. J. Modi, W. Shen, M. Tambe, and M. Yokoo. Adopt: asynchronous distributed constraints optimization with quality guarantees. *Artificial Intelligence*, 161:1-2:149–180, January 2005.
- [MV04] J. Modi and M. Veloso. Multiagent meeting scheduling with rescheduling. In *Proc. 5th workshop on distributed constraints reasoning DCR-04*, Toronto, September 2004.
- [MVS004] P. Modi, M. Veloso, S. Smith, and J. Oh. Cmradar: A personal assistant agent for calendar management, 2004.
- [NRTV07] N. Nissan, T. Roughgarden, E. Tardos, and V. Vaziriani. *Algorithmic Game Theory*. Cambridge University Press, 2007.
- [Par08] D.C. Parkes. Computational mechanism design. In *Tutorials at 10th Conf. on Theoretical Aspects of Rationality and Knowledge (TARK-05)*. Institute of Mathematical Sciences, University of Singapore, 2008.
- [PFP06] A. Petcu, B. Faltings, and D.C. Parkes. Mdpop: Faithful distributed implementation of efficient social choice problems. In *Proc. 5th Intern. Joint Conf. Autonom. Agents and Multiagent Sys. (AAMAS-06)*, pages 1397–1404, 2006.
- [SK06] Elhadi Shakshuki and Hsiang-Hwa Koo. Software agents for meeting scheduler. *aina*, 02:252–256, 2006.
- [SLB08] Y. Shoham and K. Leyton-Brown. *Multiagent Systems: Algorithmic, Game Theoretic and Logical Foundations*. Cambridge University Press, 2008.
- [WF02] R. J. Wallace and E. Freuder. Constraint-based multi-agent meeting scheduling: effects of agent heterogeneity on performance and privacy loss. In *Proc. 3rd workshop on distributed constraint reasoning, DCR-02*, pages 176–182, Bologna, 2002.
- [WL06] Zhihui Wang and Danjun Liu. Patent (intel corp.): Method, apparatus, and system for distributed meeting scheduling based on autonomous multi-agent, 2006. <http://www.patentstorm.us/patents/71108173-description.html>.
- [YH00] M. Yokoo and K. Hirayama. *Distributed Constraint Satisfaction Problems*. Springer Verlag, 2000.
- [ZM06] R. Zivan and A. Meisels. Message delay and discsp search algorithms. *Annals of Mathematics and Artificial Intelligence (AMAI)*, 46:415–439, 2006.
- [ZXWW05] W. Zhang, Z. Xing, G. Wang, and L. Wittenburg. Distributed stochastic search and distributed breakout: properties, comparison and applications to constraints optimization problems in sensor networks. *Artificial Intelligence*, 161:1-2:55–88, January 2005.